A Comparison on Data Augmentation Methods Based on Deep Learning for Audio Classification

A Comparison on Data Augmentation Methods Based on Deep Learning for Audio Classification

Shengyun Wei[1*], Shun Zou[1] , Feifan Liao[1] and weimin lang[1]

[1] College of Information and Communication, National University of Defense Technology, Wuhan, 430019, China

[*]Corresponding author's e-mail: yumocloud@protonmail.ch

Abstract. Deep learning focuses on the representation of the input data and generalization of the model. It is well known that data augmentation can combat overfitting and improve the generalization ability of deep neural network. In this paper, we summarize and compare multiple data augmentation methods for audio classification. These strategies include traditional methods on raw audio signal, as well as the current popular augmentation of linear interpolation and nonlinear mixing on the spectrum. We explore the generation of new samples, the transformation of labels, and the combination patterns of samples and labels of each data augmentation method. Finally, inspired by SpecAugment and Mixup, we propose an effective and easy to implement data augmentation method, which we call Mixed frequency Masking data augmentation. This method adopts nonlinear combination method to construct new samples and linear method to construct labels. All methods are verified on the Freesound Dataset Kaggle2018 dataset, and ResNet is adopted as the classifier. The baseline system uses the log-mel spectrogram feature as the input. We use mean Average Precision @3 (mAP@3) as the evaluation metric to evaluate the performance of all data augmentation methods.

1. Introduction

Deep learning has achieved great success in computer vision tasks such as image classification, object location and detection, and image segmentation[1]. This is closely related to the continuous evolution of deep network models, the improvement of computing power and the rapid development of big data. Similarly, in audio-related tasks, such as speech recognition[2] and audio classification[3], deep learning based methods achieve state-of-the-art performance. Audio classification is a basic and most common problem in the field of audio signal processing. This task is essentially a mode discrimination problem that judges the category of audio clips according to the feature and context of the audio signal. Typical application scenarios include analysis of sound events, instrument and artist identification, environmental monitoring and smart assistants. Despite the broad application prospects, such methods still face many challenges in practical. Particularly, deep network models with large parameters are heavily dependent on large-scale training samples[4]. Unfortunately, many tasks lack large amounts of diverse, trustworthy data. The consequences of this is model overfitting and poor generalization. This problem is extremely challenging in both the visual and the audio fields. Data augmentation techniques have been proven to effectively alleviate the model overfitting[5-11]. However, there are many data augmentation methods for audio tasks in the existing literature, and as far as we know, there is no literature to make a fair comparison and analysis for them. This paper mainly includes: firstly, we summarize the existing methods of audio data augmentation, and compare them in detail. Secondly, we

conduct experiments on a representative large dataset, hoping to provide a more reliable benchmark for researchers. Thirdly, we propose a new effective data augmentation method and prove its effectiveness.

This paper focuses on the data augmentation methods in audio classification tasks. It should be noted that data augmentation is not the only way to reduce overfitting and improve the generalization ability of deep learning models. Other strategies, such as model structure optimization, transfer learning, One-shot and Zero-shot learning deal with overfitting from different aspects. However, since the essence of overfitting is the mismatch between deep network model and the training data. Different from other methods, data augmentation starts from the perspective of data. It assumes that it can extract more useful information from the original data set through data perturbation and combination.

Why does data augmentation technology improve the performance of the algorithm? In fact, data augmentation averages over the orbits of the group that keeps the data distribution invariant, which leads to variance reduction[12]. Slightly different from image data augmentation technology, audio data augmentation somehow depends on expert knowledge. Audio signals, for example, are time series, and have pitch and tempo. The spectrum has physical meaning both horizontally and vertically in the two-dimensional coordinate system. Therefore, when we discuss the data augmentation method, we pay more attention to the unique features of the audio signal, which is different from the image, so that we can better understand the audio signals and build a more efficient classification system.

The remainder of this paper is organized as follow： in section 2, we first briefly review the data augmentation methods in image domain and audio domain, and summarize the methods. In section 3, different kinds of data augmentation methods are introduced in detail. In section 4, we fairly compare the performance of these methods through adequate experimentation. Finally, in section 5, we get the conclusion and related discussion.

2. Related work

The main processing flow of audio classification includes: pre-processing the original audio data, feature extraction and feeding the feature into the deep convolutional neural network. It can be seen that, although audio signal classification has its uniqueness, once it is converted into a two-dimensional spectrum, its subsequent processing methods (including data augmentation) can refer to image. Therefore, we first summarize some data augmentation methods for images, and then introduce the data augmentation methods for audio only.

Common data augmentation methods for image are simple transformations, including random crops, horizontal flipping, random rotations and affine transformations. The recently proposed two methods call Cutout[5] and random Erasing[6] are dedicated to solve the computer vision problems with occlusion. The accuracy of image classification is improved by randomly cutting out the square area from the input image, which enables the network to focus on the whole image instead of a certain part of it. Although these methods differ greatly in processing, but what these methods have in common is that they do not change the label of the original sample.

SamplePairing[7], Mixup[8] and Between-Class learning[9] generate new examples by linear mixing two samples. In particular, the newly generated soft label also perform linear mixing. This is different from the previous label-preserving methods. Soft label of the sample indicates that the sample does not exactly belong to a certain category (or several categories). For example, if there are two pictures with the labels of cat and dog respectively, a new sample can be obtained after linear mixing. The one-hot label of this sample may be: cat -0.4 and dog 0.6. These linearity-based methods show strong performance in multiple tasks and are very easy to extend to audio domain. Mix-example data augmentation[10] is a new and more generalized form of Mixup, which gets rid of the linear way of sample mixing in Mixup and replaces it with the non-linear combination way to generate new samples. The results show that compared with the linear-based method, the Mixed sample data method has stronger adaptability and expansion space.

The successful application of Mixup and other data augmentation methods of reconstructed labels in the image task is also replicated in the audio task. However, the temporal nature of the speech signal and the special physical meaning of the spectrum make the audio data have special augmentation

methods, such as Add Gaussian noise, time stretch and pitch shift are three common augmentation methods for the raw audio signal. SpecAugment[11] is the latest data augmentation method for spectrum proposed by Google. In this method, the two-dimensional spectrum diagram is treated as an image with time on the horizontal axis and frequency on the vertical axis. Using time warping, frequency masking and time masking augment audio. This paper analyzes the physical meaning of spectral map, inspired by mixed-example data augmentation and SpecAugment, with the method of nonlinear combination alternative frequency masking, we call this method Mixed Frequency Masking data augmentation. The effectiveness of these methods are proved by the comparative experiments. The above methods will be elaborated in detail in the next section.

3. Augmentation strategies

Data pre-processing mainly includes two stages. The first stage is to clip the silent parts in the audio signal and retain the informative parts. The second stage is to filter out the noise in the audio signal and improve the quality of the signal. In the feature extraction process, if the end-to-end classification structure is adopted, the original audio clips are directly taken as the network input and no additional feature extraction is required. Currently, the mainstream processing method is to extract the expert features, mainly including MFCC and log-mel spectrogram. In addition, many literatures have proved that log-mel spectrogram has better performance as the input of convolutional neural network.

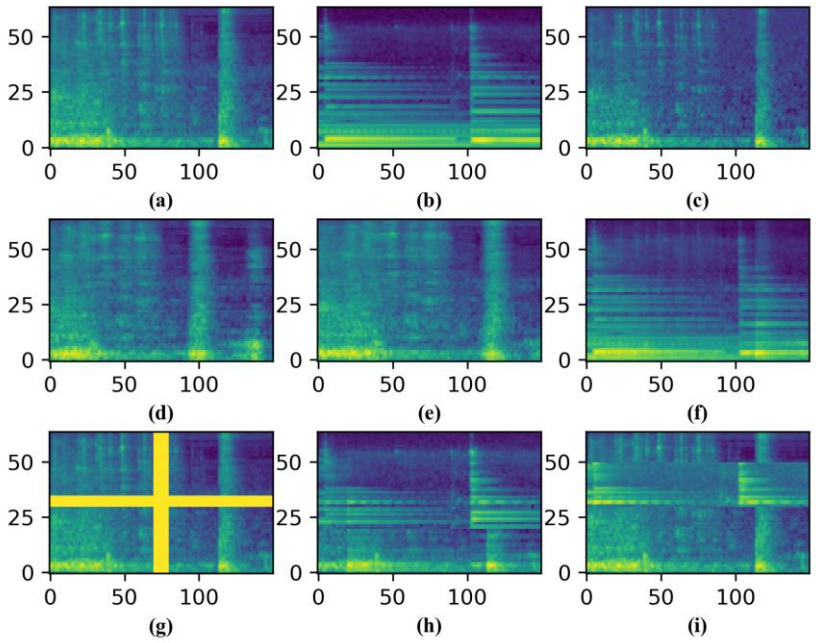

Figure 1. Visualization of of log-mel spectrogram and augmentation strategies. (a) and (b) are the log-mel spectrogram of two audio sample, (c) Add Noise, (d) Time Stretch, (e) Pitch Shift,(f) Mixup,(g) SpecAugment, (h) VH-Mixup, (i) Mixed Frequency Masking.

Data augmentation is a regularization method usually used for deep neural network input, which can be divided into two types: one is to perturb the samples to generate new samples, and then add the new ones to the original data set to explicitly expand the dataset. Other methods mix, cut and combine the samples, which does not substantially increase the number of samples. Deep neural network mainly uses convolutional neural network, including VGG[13], ResNet[14], DenseNet[15], MobileNet[16]. In particular, this paper focuses only on data augmentation methods in audio classification. Therefore, there are some commonly used audio classification techniques, such as multi-feature learning, statistical features, network structure design, ensemble learning beyond the scope of this paper. In addition, considering the calculation amount and performance comprehensively, all the data augmentation methods in this paper did not increase the number of samples, but only disturbed or mixed the samples.

The data augmentation methods on the raw signal and log-mel spectrogram are showed in figure 1, which mainly includes add Gaussian noise to the samples (Add Noise for short), time stretch the signal without changing the pitch (Time Stretch for short), pitch shift the sound up or down without changing the tempo (Pitch Shift for short), Mixup, SpecAugment and MixedSpecAugment. Among them, the SamplePairing is a special case of Mixup. MixedSpecAugment including SpecMix[17], VH-Mixup[10] and the improved method proposed in this paper Mixed Frequency Masking.

3.1. Add Noise

Add Gaussian noise to the audio samples can make the input space smoother and easier to learn. Noise can also be added to weights, gradients, and even activation functions, not only input space. The mean of Gaussian noise $n(t)$ is 0 and the variance is 1, so it can be generated easily by pseudo-random number generator. Noise amplitude σ is an important factor of noise. It is a hyper-parameter that can be configured. When the noise amplitude is too small, it is difficult to disturb, and too large, the classifier is difficult to learn. We set an acceptable range for it. $\sigma \in [0.001, 0.015]$, and obeys uniform distribution, $x(t)$ is the raw signal. The newly generated sample after Add Noise augmentation can be expressed as:

$$x(t) = x(t) + \sigma \times n(t) \tag{1}$$

3.2. Time Stretch

Time Stretch changes the tempo and length of an audio clip without changing its pitch γ is the stretch factor. $\gamma \in [0.8, 1.25]$, obeys uniform distribution. If $\gamma > 1$ then the signal is sped up. If $\gamma < 1$, then the signal is slowed down. In order for the disturbed signal to fit the input size of the network, we need to ensure that the length of the stretched signal remains the same. As a result, we apply zero padding if the time stretched audio is not long enough to fill the whole space, or crop the time stretched audio if it ended up too long. We implemented this data augmentation using the time_stretch function in the open source librosa toolkit[18].

3.3. Pitch Shift

Pitch Shift changes the pitch of a waveform by n_steps semitones without changing the tempo. It is the reciprocal process to Time Stretch. $n\_steps \in [-4, 4]$, obeys uniform distribution. We implemented this data augmentation approach using the pitch_shift function in the open source librosa toolkit[18].

3.4. Mixup

Mixup[7] adopts linear interpolation method to recombine two random samples $(x_i, y_i), (x_j, y_j)$ to generate new samples, λ is the mixed factor, and $\lambda \sim Beta(\alpha, \alpha)$,for $\alpha \in (0, +\infty)$. Unlike label preserving methods, Mixup construct the soft label of the new sample by mixing two labels. Labels are generally encoded by one-hot vector, and the newly generated soft labels can be considered to belong to multiple categories of raw samples. Mixup controls the model complexity (or implicitly increases the training data) through this linear interpolation method, thereby reducing generalization gap, inhibiting model overfitting, and improving the generalization ability of the model. The newly generated samples and labels can be expressed as:

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j \tag{2}$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j \tag{3}$$

Through observation we can know that when $\lambda = 0.5$, Mixup becomes SamplePairing this time.

3.5. SpecAugment

SpecAugment[11] can actually be regarded as an expansion of Cutout[5]or Random Erasing[6] technology in the spectrum. Since there will be no occlusion problem in the spectrum diagram, SpecAugment makes full use of the characteristics of the spectrum diagram to augment the mel spectrum

by shielding (multiple) frequency band signals (horizontal coordinates) and multiple time band signals (vertical coordinates). This allows the network to focus less on the time-frequency characteristics of a particular frequency or time, and more on the entire spectrum. This paper focuses on two groups of strategies in SpecAugment: frequency masking and time masking. Frequency masking applys mask to the f consecutive mel frequency channels $[f, f_0 + f]$, and $f \in [0, F]$ and obeys uniform distribution, F is the frequency mask parameter, $f_0 \in [0, v - f]$, where v represents the number of mel frequency channels. Similarly, time masking uses mask to the t consecutive time step $[t, t_0 + t]$, and $t \in [0, T]$, obeys uniform distribution. T is the time mask parameter, $t_0 \in [0, T - t]$.

3.6. MixedSpecAugment

MixedSpecAugment does not refer to a single data augmentation method, but a kind of method that considers the time-frequency characteristics of audio signals.Inspired by SpecAugment[11] and mixup[7], SpecMix[17] has been successfully applied to Kaggle Freesound Audio Tagging 2019 Competition. This method replaces the mask in SpecAugment algorithm by applying the frequency replacement and time replacement so that consecutive mel-frequency channels are replaced from another training sample. However, through the analysis we can find that time replacement by disturbance ordinate direction to enhance spectrum, it is very easy to lose some important spectrum information, therefore, this paper puts forward using frequency replacement, and the method named Mixed frequency Masking, the generation of samples is nonlinear, and the generation of label is linear, unlike SpecMix generate new labels, in this paper, by calculating the frequency replacements in proportion to the width of the frequency to calculate the parameters of linear interpolation, the benefits of doing so is no extra setup parameters, from the intuitive and more fits the characteristics of nonlinear composite samples.

$$\beta = \sum_{i=1}^{k} b_i / B \tag{4}$$

$$\tilde{y} = \beta y_i + (1 - \beta) y_j \tag{5}$$

Where k is the number of frequency replacements, b_i is the width of ith frequency replacements, and $b_i \in (0, 30)$. B is the number of the mel filter banks.

Table 1. Comparison of various data augmentation methods.

	Input	Sample	Label	Mode
Add Noise	raw audio	warp	preserving	—
Time Stretch	raw audio	warp	preserving	—
Pitch Shift	raw audio	warp	preserving	—
Cutout[5]	spectrogram	mask	preserving	nonlinear
Mixup[8]	spectrogram	mix	combination	linear
SamplePairing[7]	spectrogram	mix	combination	linear
SpecAugment[11]	spectrogram	mask	combination	nonlinear
SpecMix[17]	spectrogram	mix	combination	nonlinear
VH-Mixup[10]	spectrogram	mix	combination	nonlinear
Mixed Frequency Masking	spectrogram	mix	combination	nonlinear/linear

In addition, the successful application of mix-example data augmentation[10] in the field of image classification shows that non-linear data augmentation methods are of great help in narrowing the gap between training data and test data. Therefore, this paper considers the introduction of VH-Mixup, one of the most effective methods, into audio classification applications.

Table 1 summarizes and compares various data augmentation methods, the generation of new samples, the transformation of labels, and the combination patterns of samples and labels.

4. Experiment and results

In order to test the effectiveness of various audio data augmentation methods fairly, we unified various elements in the test process, including data sets, input features, data preprocessing methods, network models, etc. We repeated each experiment five times and averaged the results. The Dataset used was the Freesound Dataset Kaggle2018 (FSDKaggle2018 for short), which is a large audio file Dataset with extensive coverage and real environment collection. The data set contains 11073 Audio clips are annotated with a single ground truth label and ranges from 300ms to 30s. Among them, the number of training set samples is 9473, including 41 categories with imbalanced distribution. The number of test set samples is 1600. The classifier we use is Resnet[14], and the feature we extract is log-mel spectrogram.

We follow the FSDKaggle2018, using mean Average Precision @3 (mAP@3) as the evaluation metric. mAP@3 is computed as mean of average precisions at 3[19]:

$$mAP@3 = \frac{1}{U}\sum_{u=1}^{U}\sum_{k=1}^{min(n,3)} P(k) \tag{6}$$

Where, U is the number of scored audio files in the test data, $P(k)$ is the precision at cutoff k, and n is the number predictions per audio file.

Table 2. The mAP@3 value of each data augmentation for audio classification.

Methods	mAP@3 (%)	Performance improvement (%)
Baseline	92.46	—
Add Noise	92.73	0.27
Time Stretch	92.59	0.13
Pitch Shift	92.74	0.28
Cutout[5]	92.67	0.21
Mixup[8]	93.60	1.14
SpecAugment[11]	92.69	0.23
frequency masking[11]	92.84	0.38
time masking[11]	92.47	0.01
SpecMix[17]	93.52	1.06
VH-Mixup[10]	93.14	0.68
Mixed Frequency Masking	**93.74**	**1.28**

For the training data pre-processing part, since the duration of the raw audio signal range from 300ms to 30s, we randomly select 1.5s audio segments from the raw signal. In the feature extraction process, we uniformly use log-mel spectrogram. The number of the mel filter banks is 64 with a frame width of 80ms and the frame shift is 10ms. Last preprocessing consisted in duplicating the single channels log-mel spectrogram to 3 channels by delta features and accelerate features. The dimension of the input feature is $3\times 64\times 150$ [20]. The network structure we used for the experiment is ResNet-101[3]. Other important parameters during training are set as follows: $batch\_size = 128$, 5-fold cross-validation, $epochs = 150, \alpha = 0.2$ .In frequency masking and time masking algorithm, the maximum width of the frequency and time channel mask are 30, the number of mask set to 1. For Cutout, the min erasing area, max erasing area and min aspect ratio are 0.02, 0.4 and 0.3 respectively.

Table 2 shows the mAP@3 value of each data augmentation methods for audio classification. As can be seen from the table, the data augmentation methods carried out on the raw signal, Add Noise, Time Stretch and Pitch Shift, basically have no help for the improvement of classification performance. The Cutout method designed to solve the occlusion problem and the SpecAugment method that shine brilliantly in the automatic speech recognition did not show good performance in the test. SpecMix,

Mixup and our proposed methods Mixed Frequency Masking obtained 93.52%, 93.60% and 93.74% mAP@3s respectively, and compared to the baseline system, the performance improvement is more than 1%. Mixed Frequency Masking and Mixup get similar performance on test set. The implementation of these two methods is very simple, but the choice of hyperparameter α in Mixup has a great impact on performance, which can be verified from our previous paper[21]. However, Mixed Frequency Masking method is not sensitive to hyperparameters. The method of generating soft labels can make the algorithm more robust and reliable.

5. Conclusion

In this work, we have comprehensively compared multiple data augmentation methods in audio classification. We have experimentally proved that data augmentation methods are very helpful for the improvement of audio classification performance, especially directly use for the spectrogram. In addition, we proposed a new data augmentation method named Mixed Frequency Masking, which is simple and effective and is not sensitive to parameters. Our work can provide a reference for researchers who use deep learning methods for audio classification. For future work, we will evaluate the performance of these methods on more datasets. At the same time, we need to further explore the reasons why these data augmentation methods are effective.

Acknowledgments

This research was funded by ZK18-03-23 and ZBKY-JN-1811.

References

[1] LeCun, Y., Bengio, Y., & Hinton, G. (2015) Deep learning. Nature, 521: 436-444.
[2] Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., ... & Chen, J. (2016). Deep speech 2: End-to-end speech recognition in english and mandarin. In: International conference on machine learning. New York. pp. 173-182.
[3] Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, R. C., ... & Slaney, M. (2017) CNN architectures for large-scale audio classification. In: 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). New Orleans. pp. 131-135.
[4] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. Lake Tahoe, Nevada. pp. 1097-1105.
[5] DeVries, T., & Taylor, G. W. (2017) Improved Regularization of Convolutional Neural Networks with Cutout. arXiv preprint arXiv:1708.04552.
[6] Zhong, Z., Zheng, L., Kang, G., Li, S., & Yang, Y. (2017) Random erasing data augmentation. arXiv preprint arXiv:1708.04896.
[7] Inoue, H. (2018) Data augmentation by pairing samples for images classification. arXiv preprint arXiv:1801.02929.
[8] Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2017) mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412.
[9] Tokozume, Y., Ushiku, Y., & Harada, T. (2017) Learning from between-class examples for deep sound recognition. arXiv preprint arXiv:1711.10282.
[10] Summers, C., & Dinneen, M. J. (2019) Improved mixed-example data augmentation. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). Hawaii. pp. 1262-1270.
[11] Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019) Specaugment: A simple data augmentation method for automatic speech recognition. arXiv preprint arXiv:1904.08779.
[12] Chen, S., Dobriban, E., & Lee, J. H. (2019) Invariance reduces variance: Understanding data augmentation in deep learning and beyond. arXiv preprint arXiv:1907.10905.

[13] Simonyan, K., & Zisserman, A. (2014) Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556.
[14] He, K., Zhang, X., Ren, S., & Sun, J. (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Las Vegas. pp. 770-778.
[15] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Hawaii. pp: 4700-4708.
[16] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017) Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.
[17] Github. (2019) Freesound Audio Tagging 2019. https://github.com/ebouteillon/freesound-audio-tagging-2019.
[18] Librosa development team. (2019) LibROSA. https://librosa.github.io/librosa/.
[19] Kaggle. (2018) Freesound General-Purpose Audio Tagging Challenge. https://www.kaggle.com/c/freesound-audio-tagging/overview/evaluation.
[20] Xu K, Zhu B, Kong Q, et al. (2019) General audio tagging with ensembling convolutional neural networks and statistical features. The Journal of the Acoustical Society of America, 145: 521-527.
[21] Wei, S., Xu, K., Wang, D., Liao, F., Wang, H., & Kong, Q. (2018) Sample mixed-based data augmentation for domestic audio tagging. In: DCASE 2018 Workshop, Surrey. pp: 93-97.