Class-wise Centroid Distance Metric Learning for Acoustic Event Detection

Xugang Lu[1*], *Peng Shen*[1], *Sheng Li*[1], *Yu Tsao*[2], *Hisashi Kawai*[1]

[1]National Institute of Information and Communications Technology, Japan
[2]Research Center for Information Technology Innovation, Academic Sinica, Taiwan
xugang.lu@nict.go.jp

Abstract

Designing good feature extraction and classifier models is essential for obtaining high performances of acoustic event detection (AED) systems. Current state-of-the-art algorithms are based on deep neural network models that jointly learn the feature representation and classifier models. As a typical pipeline in these algorithms, several network layers with nonlinear transforms are stacked for feature extraction, and a classifier layer with a softmax transform is applied on top of these extracted features to obtain normalized probability outputs. This pipeline is directly connected to a final goal for class discrimination without explicitly considering how the features should be distributed for inter-class and intra-class samples. In this paper, we explicitly add a distance metric constraint on feature extraction process with a goal to reduce intra-class sample distances and increase inter-class sample distances. Rather than estimating the pair-wise distances of samples, the distances are efficiently calculated between samples and class cluster centroids. With this constraint, the learned features have a good property for improving the generalization of the classification models. AED experiments on an urban sound classification task were carried out to test the algorithm. Results showed that the proposed algorithm efficiently improved the performance on the current state-of-the-art deep learning algorithms.

Index Terms: acoustic event detection, distance metric learning, class centroids, convolutional neural network.

1. Introduction

Acoustic scene and event detection (AED) is important for audio content analysis and audio information retrieval [1, 2, 3, 4, 5, 6]. In most AED algorithms, feature extraction and classifier modeling are included in a typical pipeline. How to design discriminative features for AED is essential to obtain a good performance of AED systems, especially for the generalization ability of the systems. With successful applications of deep learning (DL) framework in image and speech processing and recognition, the DL framework also has been applied in the AED tasks. The advantage of this DL framework is that they can automatically learn discriminative features and classifiers in a joint learning framework. Many models with various types of network architectures have been proposed in the DL framework. For example, the convolutional neural network (CNN) model can explore temporal- and/or frequency-shift invariant features for AED [7, 8, 9]. The recurrent neural network (RNN) model can extract long temporal-context information in feature representation for classification. With long short term memory (LSTM) units [10] or gated recurrent units (GRU) [11], the RNN can be efficiently trained for AED. Models that combines the advantages of the CNN and RNN also have been proposed, e.g., convolutional recurrent neural network (CRNN) model, where the CNN is used to explore frequency-shift invariant feature while the RNN is used to model the temporal structure in classification [12, 13].

In the DL framework for AED, there are two basic steps in modeling, one is how to encode the acoustic signals with various time durations to fixed-dimension feature vectors, the other is how to design a classifier to model these encoded feature vectors for classification. Correspondingly, in most DL based algorithms, a feature process module that stacks several neural network layers with nonlinear transforms is used for feature extraction, and a classifier module with a softmax transform is applied on top of these extracted features to obtain normalized probability outputs. The optimization goal is directly connected to the final classification accuracy on a training data set. Since the optimization is directly connected to the classification accuracy on a training data set, there is no guarantee of whether the extracted features are discriminative or not for a test set. Features as intermediate outputs in optimization, there is no explicit constraint on how their distributions should be. In consequence, it is easy for learned models to be overfitted to training data sets with weak generalization ability to testing sets. Therefore, explicit constraints should be given to feature extraction process in the DL framework. Intuitively, distribution of discriminative features of samples should have a small intra-class variation and large inter-class variation. Based on this intuition, several algorithms have been designed to combine distance metric learning in feature extraction. For example, in a large category of machine learning, feature learning takes into account of intra- and inter-class pair-wise distance measurements [14, 15, 16]. In the DL framework, nonlinear distance metric learning has been proposed for different applications [17, 18, 19, 20, 21], they all take a similar idea in feature extraction with the pair-wise Siamese network models as originally proposed in [23, 24, 19]. As a further generalization of the idea based on pair-wise Siamese network for feature extraction, triplet loss was proposed [22]. Most of these algorithms learn features with consideration of the feature distance or relation of intra-class (or positive) and inter-class (or negative) samples.

In order to reduce the large computational complexity due to the large number of pair-wise or triple-wise sample combinations, the center loss based algorithm [25] was proposed for discriminative feature extraction. In this algorithm, the intra-class center loss was applied as a constraint in discriminative feature extraction. The center loss is defined based on intra-class sample distances to their own class centroids where the centroid is an average of all samples in each class. Inspired by this center loss based idea, in our algorithm for AED, we explicitly add a distance metric constraint in feature extraction with a goal to reduce intra-class sample distances and increase inter-class sample distances. The distances are efficiently calculated between samples and class cluster centroids. With this con-

The work is partially supported by JSPS KAKENHI No. 19K12035

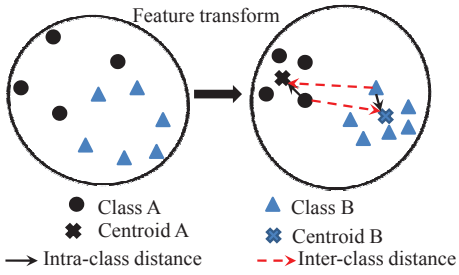

Figure 1: *Feature transform with decreasing and increasing centroid distances of intra-class and inter-class samples respectively.*

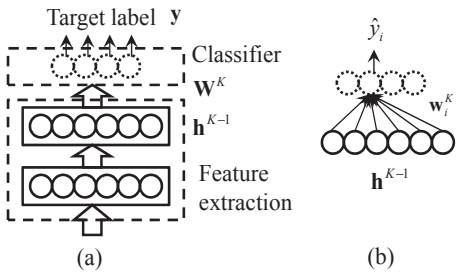

Figure 2: *Two-stage coupling framework: (a) feature extraction and classifier modeling, (b) class-specific score calculation.*

straint, the learned features have a good property for improving the generalization of the classification model. Our contributions are briefly summarized as: (1) We explicitly add intra-class and inter-class sample distance measurements in feature extraction, and define a class-wise centroid layer to calculate the distance measurement with an aim to reduce intra-class sample distances and increase inter-class sample distances; (2) Integrating the class-wise centroid distance of samples with a conventional DL framework for AED tasks which improves the performance.

2. The proposed framework for class-wise centroid distance metric learning

The basic idea is illustrated in Fig. 1. In a transformed space, the class-wise centroid is marked as the cluster average for each class. The centroid distances are defined as the distances between samples and class centroids. As illustrated in this figure, for classification (classes A and B), the purpose for feature transform is to decrease centroid distances for intra-class samples (solid line connections) while increasing them for inter-class samples (dash line connections). If adding this constraint to a conventional deep learning framework explicitly, we suppose that the learned features should be much more discriminative for improving generalization of a classification model.

2.1. Deep convolutional neural network for AED

Deep convolutional neural network (DCNN) is one of the current state of the art model architectures for AED tasks. In this paper, we also take the DCNN as our baseline modeling architecture. As explained in introduction, although the DCNN model for classification is try to learn the input-target mapping function, we can regard the processing as two coupled functions of feature extraction and classifier modeling as we did before [21]. The coupling network and classification score calculation are illustrated in Fig. 2. In a DCNN model with $K-1$ hidden layers, the output of a hidden layer is represented as

$$\mathbf{h}^k = \mathbf{f}^k\left(\mathbf{W}^k * \mathbf{h}^{k-1} + \mathbf{b}^k\right), \tag{1}$$

where $k = 1, ..., K-1$, and $\mathbf{h}^0 = \mathbf{x}$ is the input layer with feature vector $\mathbf{x}$, "$*$" is a convolutional operator. $\mathbf{W}^k$ and $\mathbf{b}^k$ are the convolutional kernel matrix and bias of the k-th hidden layer, respectively. Furthermore, $\mathbf{f}^k(.)$ is a nonlinear active function (an element-wise transform), parametric rectified linear unit (PReLU) is used in this study [26]. For an input acoustic event sample, the extracted feature representation is $\mathbf{h}^{K-1}$ (the last hidden layer output), based on this feature, the classifier model is designed as:

$$\mathbf{z} = \mathbf{W}^K\mathbf{h}^{K-1} + \mathbf{b}^K, \tag{2}$$

where $\mathbf{z} \in \mathrm{R}^C$ is the classification score vector with the i-th element z_i as a scalar score for the i-th class defined as ($i = 1, 2, ..., C$, and C is the total number of classes):

$$\begin{aligned} z_i &= \left(\mathbf{w}_i^K\right)^T\mathbf{h}^{K-1} + b_i^K \\ &= \left\langle \mathbf{w}_i^K, \mathbf{h}^{K-1}\right\rangle + b_i^K \\ &= \left\langle \hat{\mathbf{w}}_i^K, \hat{\mathbf{h}}^{K-1}\right\rangle, \end{aligned} \tag{3}$$

where $\mathbf{w}_i^K$ is the i-th column of classifier matrix $\mathbf{W}^K$ which is a class-specific weighting vector, $\hat{\mathbf{w}}_i^K$ and $\hat{\mathbf{h}}^{K-1}$ are the augmented classifier and feature vectors, respectively. A softmax transform is applied on the classification scores for normalizing the scores to probabilities as:

$$\hat{y}_i = \frac{\exp(z_i)}{\sum_{j=1}^{C}\exp(z_j)} \tag{4}$$

In learning, the loss function is defined as the cross-entropy (CE) between the estimated and true targets:

$$l_{\mathrm{ce}} \triangleq \sum_{\#\mathrm{Samples}} \mathrm{CE}(y_i, \hat{y}_i) = -\sum_{\#\mathrm{Samples}}\sum_{i=1}^{C} y_i \log \hat{y}_i, \tag{5}$$

where "#Samples" means the number of total training samples. From Eq. 3, we can see that the classification score is measured based on the inner product between the feature vector and a class specific vector where the class specific vector $\mathbf{w}_i^K$ for $i = 1, 2, ..., C$ is learned without any explicit connections to the feature vector $\mathbf{h}^{K-1}$. Actually, this class specific vector can be prototype samples of each class which should have a close and explicit relation to the samples for each class. In this paper, the class specific vectors are connected to the class centroids in a feature space.

2.2. Class-wise centroid distance metric

In a feature space, one class specific centroid vector $\mathbf{c}_i$ is obtained as the class center averaged on all samples in each class ($i = 1, 2, ..., C$). In order to learn discriminative feature, the feature distributions should satisfy small intra-class distances and large inter-class distances. The class centroid distance is defined as $d_i = \mathrm{Dist}\left(\mathbf{h}^{K-1}, \mathbf{c}_i\right)$ (refer to Fig. 3). Suppose the j-th sample is with class label y_j, then the sum of intra- and inter-class centroid distances are defined as:

$$\begin{aligned} d_{\mathrm{intra}} &\triangleq \sum_{i=1}^{C}\sum_{j\in\{y_j=i\}} \mathrm{Dist}\left(\mathbf{h}_j^{K-1}, \mathbf{c}_i\right) \\ d_{\mathrm{inter}} &\triangleq \sum_{i=1}^{C}\sum_{j\in\{y_j\neq i\}} \mathrm{Dist}\left(\mathbf{h}_j^{K-1}, \mathbf{c}_i\right), \end{aligned} \tag{6}$$

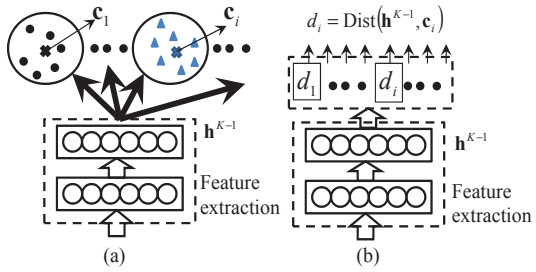

Figure 3: *Class centroid distance metric learning: (a) class-wise centroid estimation, (b) centroid distances.*

where $\mathrm{Dist}\left(\mathbf{h}_j^{K-1}, \mathbf{c}_i\right)$ is the distance between the j-th sample vector and the i-th class centroid. There are two types of definitions of the distance metric, one is l_2 norm based Euclidian distance defined as:

$$\mathrm{Dist}_{l_2}\left(\mathbf{h}^{K-1}, \mathbf{c}_i\right) = \left|\mathbf{h}^{K-1} - \mathbf{c}_i\right|_2, \tag{7}$$

and the other is inner product defined as:

$$\mathrm{Dist}_{\mathrm{inner}}\left(\mathbf{h}^{K-1}, \mathbf{c}_i\right) = \left\langle \mathbf{c}_i, \mathbf{h}^{K-1} \right\rangle \tag{8}$$

The inner product defined in Eq. 8 is a similarity metric which takes the norm lengths and angles between the feature vectors and centroid vectors. Comparing Eqs. 3 and 8, we can see that there is a close relation in the classification score calculation. The only difference is that the class-wise vector in Eq. 3 is not constrained as the class-wise centroid.

2.3. Integration of class-wise centroid distance metric learning in classification modelling

We can add the distance metric learning in conventional CE based learning (defined in Eq. 5) as a regularization term by minimizing intra-class distances d_{intra} while maximizing inter-class distances d_{inter} (negative for inner product based similarity metric). Theoretically, the learned features should be discriminative with a better generalization property than without the explicit regularization. However, due to the different dynamic ranges of the losses in CE and distance metric, it is difficult to control a good balance between the CE and distance metric learning with optimization algorithms. We further define the distance metric based classifier model with a normalized score. For a feature vector $\mathbf{h}^{K-1}$, following the definition in Eq. 4, the normalized probability score is defined for Euclidian distance based metric as:

$$\begin{aligned} s_i &= \frac{1}{1 + \frac{\sum_{j=1, j\neq i}^{C} \exp\left(-\mathrm{Dist}_{l_2}\left(\mathbf{h}^{K-1}, \mathbf{c}_j\right)\right)}{\exp\left(-\mathrm{Dist}_{l_2}\left(\mathbf{h}^{K-1}, \mathbf{c}_i\right)\right)}} \\ &= \frac{\exp\left(-\mathrm{Dist}_{l_2}\left(\mathbf{h}^{K-1}, \mathbf{c}_i\right)\right)}{\sum_{j=1}^{C} \exp\left(-\mathrm{Dist}_{l_2}\left(\mathbf{h}^{K-1}, \mathbf{c}_j\right)\right)} \end{aligned} \tag{9}$$

And similarly for inner product based metric is defined as:

$$\hat{s}_i = \frac{\exp\left(\mathrm{Dist}_{\mathrm{inner}}\left(\mathbf{h}^{K-1}, \mathbf{c}_i\right)\right)}{\sum_{j=1}^{C} \exp\left(\mathrm{Dist}_{\mathrm{inner}}\left(\mathbf{h}^{K-1}, \mathbf{c}_j\right)\right)} \tag{10}$$

In both conditions, the loss function is defined as:

$$l_{\mathrm{metric}} \triangleq \sum_{\#\mathrm{Samples}} \mathrm{CE}\left(s_i, \hat{s}_i\right) = -\sum_{\#\mathrm{Samples}} \sum_{i=1}^{C} s_i \log \hat{s}_i \tag{11}$$

In this definition, s_i can be regarded as similarity label which is the same as class label y_i. Eq. 11 can be used independently for optimization in classification modeling. In our study, the performance is a little worse than directly using l_{ce} in optimization as defined in Eq. 5. We combine them to the following loss in optimization as:

$$l_{\mathrm{total}} = l_{\mathrm{ce}} + \lambda l_{\mathrm{metric}}, \tag{12}$$

where λ is a regularization coefficient to control the tradeoff between the two losses. The method of learning the model parameters is based on minimizing this objective function defined in Eq. (12) with additional parameter regularization (e.g., weight decay) as follows:

$$l_{\mathrm{total}}\left(\Theta\right) = l_{\mathrm{ce}}\left(\Theta\right) + \lambda l_{\mathrm{metric}}\left(\Theta\right) + \alpha R\left(\Theta\right), \tag{13}$$

where $\Theta = \left\{\mathbf{W}^k, \mathbf{b}^k, k = 1, 2, ..., K\right\}$ is the model parameter set with neural weight matrix $\mathbf{W}^k$ and bias $\mathbf{b}^k$. $R\left(\Theta\right)$ is the regularization for model parameters (e.g., either L_2 or L_1 regularization) which has been shown to improve the generalization ability of the model.

3. Experiments and results

We carried out experiments on an AED task where the UrbanSound8K data corpus was used. The task is to recognize acoustic scene or event categories with given duration of acoustic signals [7, 27], and many studies carried out research work based on deep architectures to test their algorithms on this data corpus [7, 8, 27]. In this data corpus, there are 8732 sound clips (less than 4 seconds) of 10 classes labeled on clip-level as: air conditioner, car horn, playing children, dog bark, drilling, engine idling, gun shot, jackhammer, siren, street music. All sounds were organized into ten folds. In our study, samples in eight folds were used as training set, and samples in the remaining two folds were used for validation and testing sets respectively. The log compressed Mel filter band spectrum (MFBS) was used as input to the network model. In the MFBS extraction, all sounds were down-sampled with a 16 kHz sampling rate, 512-point windowed FFT with 256-point shift was used for frame-based power spectrum extraction, and 60 Mel filter bands were used for subband spectrum extraction.

Many models based on deep learning have been proposed for the AED [7, 8, 4]. Among these models, the DCNN based models perform consistently well due to their strong power in temporal-frequency invariant feature extraction. Therefore, in this paper we implemented a DCNN as a baseline model. The DCNN was composed of two convolutional blocks, and each block was with a convolution layer (256 kernels with a receptive filed as 3*3 for each), PReLU nonlinear activation function, and max-pooling. A global average pooling (GAP) was applied to the output of the last convolution block to obtain a fixed-dimension feature vector for each training sample. In order to enhance the DCNN model, batch normalization (BN) algorithm was applied to increase the training speed and improve the performance [29]. In our implementation, a little different from the original usage of the BN [29], the BN was applied in the input layer of each convolution processing. The proposed class-wise centroid distance metric based learning was integrated in

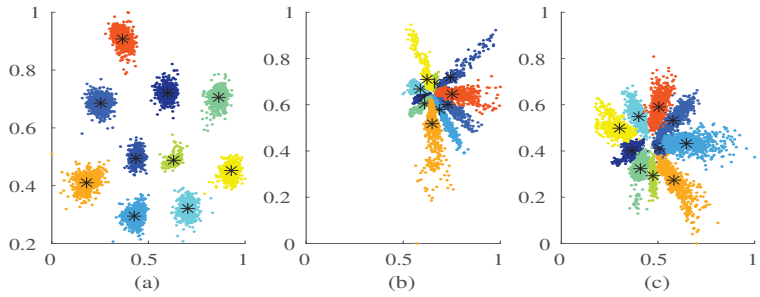

Figure 4: *Distributions of acoustic event samples in learned feature space with an DCNN model regularized with: (a) intra-class centroid Euclidian distance loss, (b) class-centroid distance based softmax loss, (c) original model*

Table 1: *Performance with different regularization parameters for different models (classification accuracy) (%)*

Methods	Valid	Test
Baseline	79.68	80.96
Distance based softmax (Dist_{l_2})	77.64	79.33
Distance based softmax ($\text{Dist}_{\text{inner}}$)	77.82	80.76
Dist_{l_2} ($\lambda = 0.1$)	80.27	81.48
Dist_{l_2} ($\lambda = 0.01$)	80.12	80.88
Dist_{l_2} ($\lambda = 0.005$)	80.64	**83.15**
Dist_{l_2} ($\lambda = 0.001$)	79.53	82.20
Dist_{l_2} ($\lambda = 0.0001$)	80.02	81.96
$\text{Dist}_{\text{inner}}$ ($\lambda = 0.1$)	78.43	81.72
$\text{Dist}_{\text{inner}}$ ($\lambda = 0.01$)	80.35	82.41
$\text{Dist}_{\text{inner}}$ ($\lambda = 0.005$)	80.76	**82.92**
$\text{Dist}_{\text{inner}}$ ($\lambda = 0.001$)	80.64	82.32
$\text{Dist}_{\text{inner}}$ ($\lambda = 0.0001$)	80.02	81.84

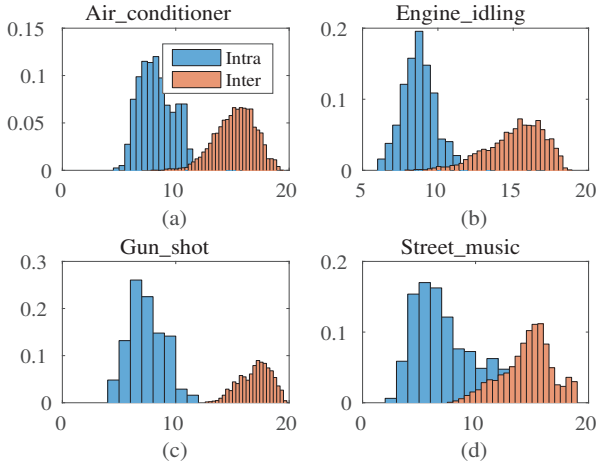

Figure 5: *Distributions of intra- and inter-class centroid distances for four types of acoustic events.*

this baseline DCNN to enhance the power of the feature extraction. In learning, a mini-batch size 32 was used, and the Adam optimization algorithm with an initial learning rate 0.001 was applied [28]. Finally, the best model parameters were selected based on the best performance on the validation set.

Before showing the AED results, we first visually check how the feature distributions are affected by the constraint of the regularization in feature learning. For convenience of checking in a 2D space, in model training, the final feature dimension was set to 2 (before input to classifier layer). The Euclidian distance metric is used, and the regularization parameter is set with $\lambda = 0.01$. The distribution of training samples of the 10 categories is shown in Fig. 4. Because with different regularization methods, the feature dynamic ranges are different. We normalized them to be in the range $[0, 1]$ for convenience of visualization. In this figure, each color represent one acoustic event category, and the black star mark is the class centroid for each category. Comparing the sample distributions from the baseline model (panel (c) of Fig. 4)), we can see that regularization with reducing sample intra-class centroid distance makes the event category cluster more compact in homogeneous directions along the feature space (panel (a) of Fig. 4)), while much more compact along some specific directions (panel (b) of Fig. 4)) when regularized with the softmax loss based on centroid distance metric. In our study, we also find a similar effect of in panel (b) when the model is regularized with softmax loss estimated based on the inner product based distance metric. For recognition, the feature dimension was set to 256, and the results are summarized in Table 1. From this table, we can see that both baseline model and distance metric based model perform well independently. When the baseline model is regularized with the distance metric based loss, the performance is effectively improved. The results indicate that integrating the distance metric based loss in feature learning helps to learn a better feature representations for model generalization.

4. Discussion and conclusion

In this paper, we confirmed that learning with a distance metric, i.e., explicitly adding constraints to reduce intra-class sample distances and increase inter-class sample distances, is effective in discriminative feature learning hence to improve generalization of classification models. For a further understanding, based on a learned model, the sample intra-class and inter-class centroid distances are calculated, and their distributions are showed in Fig. 5. From this figure, we can see that centroid distances for intra-class and inter-class samples have been clearly discriminated based on the learned features. Also from this figure, we can see that the learned features have different capacities in discriminating different class from the other left classes, for example, Gun-shot event (panel (c) in Fig. 5) is much easier to be discriminated from other event categories, while the learned features are not friend for discriminating street-music event from other event categories (panel (d) in Fig. 5). In the future, we will further develop this idea and carry out large scale experiments to examine the capacity of the algorithm.

5. References

[1] D. Giannoulisy, E. Benetosx, D. Stowelly, M. Rossignolz, M. Lagrangez and M. Plumbley, "Detection and Classification of Acoustic Scenes and Events: an IEEE AASP Challenge," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013.

[2] T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, "Context-dependent sound event detection," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 1, pp. 1-13, 2013.

[3] X. Zhuang, X. Zhou, M. A. Hasegawa-johnson, T. S. Huang, "Real-world acoustic event detection," *Pattern Recognition Letters*, vol. 31, no. 12, pp. 1543-1551, 2010.

[4] DCASE (detection and classification of acoustic scenes and events) 2016 challenge, http://www.cs.tut.fi/sgn/arg/dcase2016/index

[5] R. Grzeszick, A. Plinge, G. A. Fink, "Bag-of-Features Methods for Acoustic Event Detection and Classification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(6), pages 1242-1252, June 2017.

[6] X. Lu, Y. Tsao, S. Matsuda, C. Hori, "Sparse representation based on a bag of spectral exemplars for acoustic event detection," *ICASSP*, pp. 6255-6259, 2014.

[7] J. Salamon and J. P. Bello, "Deep Convolutional Neural Networks and Data for Environmental Sound Classification," *IEEE Signal processing letters*, Vol. 24, No. 3, pp. 279-283, 2017.

[8] K. J. Piczak, "Environmental sound classification with convolutional neural networks," *IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, 12 November 2015.

[9] A. Gorin, N. Makhazhanov, and N. Shmyrev, "DCASE 2016 sound event detection system based on convolutional neural network," *Tech. Rep., DCASE2016 Challenge*, 2016.

[10] S. Hochreiter, J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, 9 (8), pp. 1735-1780, 1997.

[11] K. Cho, B. Merrienboer, D. Bahdanau, Y. Bengio, "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches," the *8-th Workshop on Syntax, Semantics and Structure in Statistical Translation, SSST-8*, 2014.

[12] E. Cakir, G. Parascandolo, T. Heittola, H. Huttunen, T. Virtanen, "Convolutional Recurrent Neural Networks for Polyphonic Sound Event Detection," *IEEE/ACM Trans. Audio, Speech and Language Processing*, 25(6), pp. 1291-1303, 2017.

[13] K. Choi, G. Fazekas, M. Sandler, K. Cho, "Convolutional Recurrent Neural Networks for Music Classification," *ICASSP*, pp. 2392-2396, 2017.

[14] E. Xing, A. Ng, M. Jordan, and R. Russell, "Distance Metric Learning, with application to Clustering with side-information," *in proceeding of Advances in Neural Information Processing Systems*, MIT Press, pp. 521-528, 2002.

[15] K. Weinberger, J. Blitzer, L. Saul, "Distance Metric Learning for Large Margin Nearest Neighbor Classification," *Advances in Neural Information Processing Systems* 18, pp. 1473-1480, 2006.

[16] K. Weinberger, L. Saul, "Distance Metric Learning for Large Margin Classification," *Journal of Machine Learning Research*, vol. 10, pp. 207-244, 2009.

[17] M. Guillaumin, J. Verbeek, C. Schmid, "Is that you? Metric learning approaches for face identification," *the IEEE 12th International Conference on Computer Vision*, pp. 498-505, 2009.

[18] J. Hu, J. Lu, Y. Tan, "Deep Transfer Metric Learning," *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 325-333, 2015.

[19] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively with application to face verification," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 539-546, 2005.

[20] X. Lu, P. Shen, Y. Tsao, H. Kawai,"Pair-Wise Distance Metric Learning of Neural Network Model for Spoken Language Identification," *INTERSPEECH*, pp. 3216-3220, 2016

[21] X. Lu, P. Shen, Y. Tsao, H. Kawai, "Regularization of neural network model with distance metric learning for i-vector based spoken language identification," *Computer Speech and Language*, no. 44, pp. 48-60, 2017.

[22] Schroff, F., Kalenichenko, D., Philbin, J. "FaceNet: A unified embedding for face recognition and clustering," *in IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815-823, 2015.

[23] J. Bromley, I. Guyon, Y. LeCun, E. Sackinger, and R. Shah, "Signature verification using a Siamese time delay neural network," *In Proceedings of the 6th International Conference on Neural Information Processing Systems (NIPS'93)*, pp. 737-744, 1993.

[24] P. Baldi, Y. Chauvin, "Neural Networks for Fingerprint Recognition," *Neural Computation*, 5 (3). pp. 402-418, 1993.

[25] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," *In European Conference on Computer Vision*, pp. 499-515, 2016.

[26] K. He, X. Zhang, S. Ren, J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," *in Proceeding of IEEE International Conference on Computer Vision (ICCV)*, pp. 1026-1034, 2015.

[27] J. Salamon, C. Jacoby and J. P. Bello, "A Dataset and Taxonomy for Urban Sound Research," *the 22nd ACM International Conference on Multimedia*, Orlando USA, Nov. 2014.

[28] Diederik P. Kingma, Jimmy Ba, "Adam: A Method for Stochastic Optimization," *the 3rd International Conference on Learning Representations (ICLR)*, 2014.

[29] S. Ioffe, C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *In proceedings of the 32nd International Conference on International Conference on Machine Learning (ICML)*, pp. 448-456, 2015.

[30] A. J. Bell, T. J. Sejonwski, "The Independent Components of Natural Scenes are Edge Filters," Vision Res., vol. 37, no. 23, pp. 3327-3338, 1997.

[31] A. Coates, H. Lee, and A. Y. Ng, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. the 14-th International Conference on AI and Statistics*, 215-223, 2011.

[32] M. Aharon, M. Elad and A. Bruckstein, K-SVD,"An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Transactions on Signal Processing*, vol 54, no. 11, pp. 4311-4322, 2006.

[33] J. Mairal, F. Bach, J. Ponce and G. Sapiro, "Online Dictionary Learning for Sparse Coding," *International Conference on Machine Learning*, Montreal, Canada, 2009

[34] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, and C. J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871-1874, 2008.

[35] G. E. Hinton, and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504-507, 2006.

[36] Y. Bengio, and A. Courville, "Deep Learning of Representations," in *Handbook on Neural Information Processing*, Springer, Berlin Heidelberg, 2013.