A Study on Transfer Learning for Acoustic Event Detection in a Real Life Scenario

Prerna Arora, Reinhold Haeb-Umbach
Department of Communications Engineering,
Paderborn University, Germany
Email: {arora, haeb}@nt.upb.de

Abstract—**In this work, we address the limited availability of large annotated databases for real-life audio event detection by utilizing the concept of transfer learning. This technique aims to transfer knowledge from a source domain to a target domain, even if source and target have different feature distributions and label sets. We hypothesize that all acoustic events share the same inventory of basic acoustic building blocks and differ only in the temporal order of these acoustic units. We then construct a deep neural network with convolutional layers for extracting the acoustic units and a recurrent layer for capturing the temporal order. Under the above hypothesis, transfer learning from a source to a target domain with a different acoustic event inventory is realized by transferring the convolutional layers from the source to the target domain. The recurrent layer is, however, learnt directly from the target domain. Experiments on the transfer from a synthetic source database to the real-life target database of DCASE 2016 demonstrate that transfer learning leads to improved detection performance on average. However, the successful transfer to detect events which are very different from what was seen in the source domain, could not be verified.**

I. Introduction

Acoustic event detection and scene analysis are emerging fields of audio research that find roots in numerous applications like multimedia indexing, surveillance and ambient assisted living. Speaking in the language of machine learning, Acoustic Event Detection (AED) is the task of automatically labelling a particular region of the audio file as a certain event. These events can range from being short term like *clearing throat* or *cough*, to long term like *human speech* or *music*, and based on the application, can belong to any context. As compared to Automatic Speech Recognition (ASR), the field of automatic event detection is fairly new. This can be judged not only by the non-existence of a dedicated Wikipedia page, but also by the unavailability of a common and widely available sound database (excluding the very recently published Google's AudioSet [1]), and evaluation metrics for this field. AED poses several challenges like the difficulty and expense in recording and labelling the immense number of sounds in the surroundings, a deluge of data while processing in real time, presence of background noise and the concurrence of random number of different events (polyphony). However, despite the numerous challenges, efforts have been made to advance the field of AED and recognize events and their corresponding environments (Acoustic Scene Classification) on a small scale and for different applications [2]–[4]. AED has also been used immensely in the field of building smart homes with a focus on healthcare for the elderly [5], [6].

Detection and Classification of Acoustic Scenes and Events (DCASE) [7] is an IEEE Audio and Acoustic Signal Processing (AASP) challenge that provides a common platform for researchers to work with tasks of ASC, AED, and audio tagging. Unlike ASC and audio tagging, AED has to detect the start (onset) and end (offset) times of the detected event, along with its label. The real-life database in Task 3 of DCASE 2016, referred from here on as TUT-SED Real 2016 database, provides a very challenging scenario with polyphony, immense background noise and extremely limited annotated data. Therefore, we investigate a transfer learning approach for a supervised AED for this real-life database.

The technique of transfer learning, i.e., transferring knowledge or information from one or more domain(s) (source domain(s)) to another (target domain) has been employed and investigated in different machine learning problems. Different image classification tasks [8]–[11] have successfully employed transfer learning by utilizing the features learnt on a large database, for example, ImageNet to improve the visual recognition accuracy on a smaller dataset, for example, the Pascal Visual Object Classification dataset. Some studies have also used transfer learning for speech and language processing [12], where an acoustic model trained on a large database (e.g., in English) is used to improve recognition accuracy for a database with limited amount of training data (e.g., in Hebrew). However, as compared to image processing, transfer learning for speech appears to be more challenging due to a probable presence of a huge mismatch between the source and target databases corresponding to different languages, speakers, age groups, ethnicity, and, last but not the least, acoustic environments. Nevertheless, cross-lingual and multilingual transfer learning have been accompolished, not only for speech recognition [13], [14] but also for speech enhancement [15].

In comparison to image classification and speech processing, transfer learning has not yet been fully exploited in the field of AED. One very recent research for transfer learning in a similar domain has been done for ASC [16], where several publicly available sound databases, including Clear-OL [17], Real World Computing Partnership (RWCP) [18], UrbanSound [19], NOISEX [20], ETSI noise [21] and ESC-

50 [22] are used as source domains to enhance the scene classification accuracy of the TUT Acoustic Scenes 2016 dataset [7] and home surveillance environment. To the best of our knowledge, there exist only two researches in transfer learning for supervised AED. First, [23] employs speech data from Resource Management (RM) [24] and Wall Street Journal (WSJ) [25] databases for cross-acoustic transfer learning to classify 50 sound events from the RWCP database. Utilizing the similarity between the characteristics of speech and sound, their system obtains a 20% relative improvement in error rate as compared to the system trained only on the sound database. The second research in this field [26] uses transfer learning to learn audio features (AENet) from a AED task and combine them with video features to enhance the accuracy of video classification.

Most of the recent works for transfer learning, including the three mentioned above, employ Deep Neural Networks (DNN) as models to transfer the parameters from the source to the target domain. DNN facilitates multilevel feature learning where the lower layers present the capabilities to learn high-level, generic features while delegating the learning of low-level, specific features to the higher layers. However, these works ([16], [23]) utilize the basic Feed Forward (FF) layers for modeling the different events as compared to the more effective convolutional and recurrent layers as used in our work. Convolutional Neural Networks (CNN) have been proven to accentuate the image classification performance in a number of researches [27]–[30]. CNN has also been successfully utilized for visual classification using AENet features in [26]. Unlike the fully connected FF neural network, CNN introduces time and frequency invariance which allows compensating for small variations in the events. On the other hand, both the FF and CNN can only work with a small time duration of the input signal at a given time, thereby rendering the modeling of longer events ineffective. Due to this time independence, these networks are also incapable of efficiently modeling the correlations within the events. Therefore, in this work, we employ NN with convolutional layers followed by recurrent layers instead of fully connected FF layers. This type of network has been termed as Convolutional Recurrent Neural Network (CRNN) by [31], where they showcase the relative improvement of such a network over pure CNN and RNN networks for large synthetic databases.

Transfer learning for AED usually suffers from a huge mismatch between the label sets of the source and target domain. Moreover, most of the publicly available sound event databases [22], [32] are either too small containing short monophonic audio segments recorded in a synthetic environment or contain a label set which is very specific for a certain application and thus unsuitable as a source database for other applications. In our work, we therefore investigate transfer learning for a real life TUT-SED Real 2016 database by utilizing TUT-SED Synthetic 2016 database [31] as a source domain. This source database mitigates at least the above mentioned first problem by containing 100 polyphonic synthetic recordings, providing 566 minutes of data.

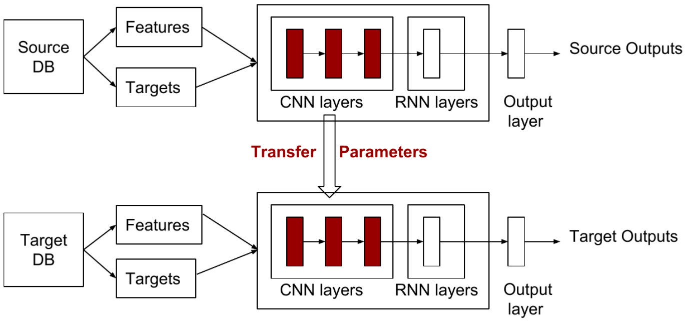

Fig. 1. A Transfer Learning Framework based on Convolutional Recurrent Neural Networks

II. Proposed Approach

A. Transfer Learning

Transfer Learning is a technique that transfers knowledge from one or more **Source** domains to a **Target** domain and can be explained mathematically using the following two terms [33]:

- **Domain** : A domain D comprises a feature space $\mathcal{X}$, and a probability distribution of the features $P(X)$ where $X = \{x_1, x_2, \cdots, x_n\} \in \mathcal{X}$
- **Task**: A task T comprises a label space $\mathcal{Y}$, and a mapping function $f(.)$, that maps the features X to the output $Y = \{y\} \in \mathcal{Y}$.

Given a source domain, D_s with its corresponding task, T_s and a target domain, D_t with its corresponding task, T_t, a transfer learning problem exists when either $D_s \neq D_t$ or $T_s \neq T_t$. The two domains are unequal when either their feature spaces are different, i.e., $\mathcal{X}_s \neq \mathcal{X}_t$ or their data distributions are different, $P(X_s) \neq P(X_t)$. In the case presented in this work, the target and source domains have different data distributions and the tasks have different label spaces. Such a transfer learning approach is called *Inductive transfer learning* [33].

B. Transfer Learning with CRNN

Our setting for transfer learning with neural networks as model is illustrated in Figure 1, where a network is trained on features extracted from a given source training database (DB) for a certain source task. Similarly at the target domain, a network is trained on features extracted from the target database with the exception that the model parameters (weights, biases, etc.) of either all or some of the layers of this network are borrowed from that of the source model.

It is hypothesized in this work that audio events are made up of a sequence of basic acoustic units as building blocks. While these building blocks are universal, it is the order in which they occur that distinguishes one event from another. For example, the event *washing dishes* can be characterized by the occurrence of basic acoustic units, such as running water, scrubbing, clinging of utensils, etc. one after another. Learning these basic units from the source domain can help learn the target events which may share some or all of the learnt acoustic units. This is analogous to speech where words manifest themselves as a sequence of building blocks called phonemes, which may be common across languages. Therefore, we use a CRNN architecture to model the source and target domains in this work. Such a model has the capacity to learn the acoustic units of the events using the initial convolutional layers, while the specific temporal order within the events is captured by the following recurrent layers. Since we would like to utilize the knowledge about the spectral characteristics of the individual events, learnt by the source model, irrespective of the time dependencies in the source recordings (because the source domain may have a different event set than the target), we transfer only the convolutional layers to the target model, dropping the last recurrent and output layers.

The target model is trained by either keeping the parameters of the borrowed source model layers frozen or fine-tuning them along with the recurrent layers to best classify the desired target events. However, how many of the source layers should be transferred to the target model and whether to keep them frozen or fine tune them, are questions that have no straightforward answers. For an inductive transfer learning, these are hyper-parameters that depend on,

- How big the source database is, as compared to the target database and,
- How similar or different are the source events from the target events.

III. Experimental Setup

A. Databases

The TUT-SED Synthetic 2016 database is employed as the source domain here. It provides 100 polyphonic signals, totalling to 566 minutes of audio, synthetically generated from isolated samples of 16 sound events. Of these 100 signals, 60% are used for training, 20% for validation and 20% for testing. Since these recordings are synthetically generated, they are free from any background noise. The label space of the source task, $\mathcal{Y}_s$={*alarms and sirens, baby crying, bird singing, bus, cat meowing, crowd applause, crowd cheering, dog barking, footsteps, glass smash, gun shot, horsewalk, mixer, motorcycle, rain, thunder*}. More information about the database can be obtained from [31].

The TUT-SED Real 2016 database is used as a target domain which provides 22 audio recordings in two real life scenes: home (10) and residential area (12). Each recording is about 3-5 minutes long, providing a total of 78 minutes of target data. Since they are recorded in a real life scenario, there is no control over the amount of background noise and the number of overlapping events. In fact, over a 100 different events, like *television, washing machine, bus engine*, etc. are present in the background superimposed with the target events in the foreground, but do not have to be detected. The target label space for both scenes together, contains 17 events in total, $\mathcal{Y}_t$={ *(object) banging, bird singing, car passing by, children shouting, people speaking, people walking, wind blowing, (object) rustling, (object) snapping, cupboard, cutlery, dishes, drawer, glass jingling, object impact, washing dishes, water tap running*}.

B. Setup and evaluation metrics

For all experiments in this work, the audio files are first downsampled to 16 kHz, converted to frequency domain using STFT frame size of $32\,\mathrm{ms}$ and frame shift of $10\,\mathrm{ms}$. Then, 40 log mel filter bank features are extracted and packed in a feature vector for every frame. Each feature vector is normalized by subtracting the sample mean and dividing by the sample standard deviation calculated on the time dimension. In parallel, annotations for these files are encoded in a 'n-hot' format, i.e., target vector for every frame is a binary vector of length equal to number of events (plus one for 'Silence') with the target value as 1 for the active events and 0 for others. Both the target and feature vectors are then used for carrying out a supervised training of the neural network. Training of every network is optimized with the learning algorithm Adam [34] at a learning rate of 10^{-6}, unless otherwise mentioned. A rectified linear unit (ReLU) activation function is applied as a non-linearity to the output of every hidden layer. The output of the last layer is thresholded with a sigmoid activation function and the loss is calculated using binary cross entropy. When no transfer learning is being performed, the weights of the network are initialized randomly. For maintaining the generalizability of the model and keeping overfitting under check, 20% of the training data is randomly chosen as validation data, which is evaluated with an error function after every training epoch. The learning rate is decreased to one-tenth of its initial value when no reduction in the validation error function happens until 15 epochs (empirically chosen).

During the testing phase, the output of the last layer is binarized by applying a threshold value of 0.35 to predict the active events in that frame. A median filter of kernel size approximately equal to $200\,\mathrm{ms}$ is applied to smoothen the final output of these networks. The value of every hyper-parameter is chosen empirically.

Evaluation of the networks is carried out using two segment based evaluation metrics: F-measure and acoustic event error rate (AEER), which are calculated as an average of intermediate metrics on 1-second segments, $k \in \{1 \cdots K\}$, as follows:

$$\text{F-measure} = \frac{2 \cdot P \cdot R}{P + R} \cdot 100, \tag{1}$$

where precision P and recall R are calculated using the total number of true positives N_{tp}, false positives N_{fp} and false negatives N_{fn} in all segments as:

$$P = \frac{\sum_{k=1}^{K} N_{tp}(k)}{\sum_{k=1}^{K}(N_{tp}(k) + N_{fp}(k))}, \tag{2}$$

$$R = \frac{\sum_{k=1}^{K} N_{tp}(k)}{\sum_{k=1}^{K}(N_{tp}(k) + N_{fn}(k))}, \tag{3}$$

and

TABLE I

CONFIGURATION USED FOR THE CNN-LSTM ARCHITECTURE. THE FILTER SIZES, STRIDES AND PADDING ARE REPRESENTED IN THE (FREQUENCY, TIME) DIMENSION.

Layers	# Filters / # HiddenUnits	FilterSize	FilterStride	Pad
cnn1	256	(5, 5)	(1, 1)	(0, 1)
maxpool1	-	(5, 1)	(5, 1)	(0, 0)
cnn2	256	(5, 5)	(1, 1)	(2, 1)
maxpool2	-	(4, 1)	(4, 1)	(0, 0)
cnn3	256	(5, 5)	(1, 1)	(1, 1)
maxpool3	-	(2, 1)	(1, 1)	(0, 0)
lstm1	256	-	-	-
output	# events	-	-	-

$$AEER = \frac{\sum_{k=1}^{K}(S(k) + D(k) + I(k))}{\sum_{k=1}^{K} N(k)}, \tag{4}$$

where $S(k)$, $D(k)$ and $I(k)$ are the number of events substituted, deleted and inserted, respectively, in the k^{th} segment as compared to the actual number of events $N(k)$ in this segment.

For detailed information about the metrics, refer to [35].

C. Neural Network configurations

The CRNN configuration utilized to model the source and the target domains is tabulated in Table I. Instead of gated recurrent units (GRU) like in [31], long short term memory (LSTM) units are used in the recurrent layers for this work, as preliminary experiments showed similar results on both [31]. The system architecture used in the rest of this paper is therefore referred to as the *CNN-LSTM* system. Every CNN layer is followed by pooling along the frequency axis, using a max-pooling layer.

D. Transfer Learning experiments

For every transfer learning experiment, the parameters of all convolutional layers of the target model are initialized with the parameters learnt for all the convolutional layers of the source model. The recurrent layers of the target model are however, always initialized from scratch. After the intialization, the training of the target model is carried out in three different ways:

- **Frozen All**, where the parameters (weights and biases) of all the convolutional layers of the source model are kept frozen and only the recurrent layers of the target model are trained with features of the target recordings.
- **Frozen One**, where the parameters of only the first convolutional layer of the source model are kept frozen, while the parameters of second and third convolutional layers are finetuned along with the recurrent layers of the target model.
- **Finetune All**, where the parameters of all the convolutional layers are finetuned along with the recurrent layers of the target model.

TABLE II

EVALUATION METRICS FOR STATE OF THE ART, CNN-LSTM BASELINE FOR NO TRANSFER LEARNING (TL) AND THE THREE TYPES OF TRANSFER LEARNING FOR SCENE INDEPENDENT AED ON THE TUT-SED REAL 2016 DATABASE. OVERALL NUMBER OF SUBSTITUTIONS (S), DELETIONS (D) AND INSERTIONS (I) HAVE BEEN NORMALIZED TO THE NUMBER OF GROUND TRUTH EVENTS.

Model	S	D	I	AEER	F measure (%)
State of the Art [31]	-	-	-	0.95	30.3
Baseline (No TL)	0.2	0.43	0.38	1.01	37.8
Frozen All	0.13	0.59	0.22	**0.94**	34.3
Frozen One	0.25	0.32	0.56	1.13	**38.2**
Finetune All	0.22	0.4	0.4	1.02	37.9

IV. Results and Discussion

A. State of the Art and Baseline

A state of the art CRNN system for the TUT-SED Real 2016 database, is provided in [31]. This system carries out a scene independent analysis, i.e., a single system is developed for both the home and the residential area scenes. Using a CRNN consisting of 3 convolutional followed by 3 GRU layers and a convolutional filter size of (3, 3), it achieves an AEER of 0.95 and F measure of 30.3% on the target database. In our work, a scene-independent CNN-LSTM system for the target database is developed with the network topology given in Table I. This CNN-LSTM system acts as a baseline for comparisons with transfer learning models, described in the next section.

As compared to the state of the art, the CNN-LSTM baseline system shows an increase in AEER along with an improvement in F measure, as reported in the first two lines of Table II. Though a direct comparison between the systems on the basis of the number of substitutions, deletions and insertions is not possible as the state of the art system does not report these intermediate metrics, one can assume that a higher AEER for the baseline is due to a larger number of insertions and deletions. The total number of correctly detected events is higher for the baseline, leading to a lower substitution value and therefore, a high F measure.

B. Transfer Learning

For all transfer learning experiments, the model trained on the source database TUT-SED Synthetic 2016, is employed. This model has been trained with the training parameters described in Section III-B and the default parameters of the optimization algorithm Adam, i.e., learning rate of 10^{-4}.

The results of the transfer learning experiments, carried out using the source model are tabulated in the last three lines of Table II and indicate the following three things:

1) Frozen All: When the parameters of the three convolutional layers of the source model are kept fixed, the AEER is improved by 7% over the baseline, however at a cost of reduction in the F-measure by 3.5%. This indicates that the convolutional layers are learning to recognize the basic acoustic units that make up different events. Overall, the 'Frozen all' system tends to produce

TABLE III

Transfer Learning (TL) results for specific target events of TUT-SED Real 2016 database for a scene-independent AED. Overall number of deletions (D) and insertions (I) have been normalized to the number of ground truth events.

Event	Model	D	I	AEER	F measure
Bird Singing	Baseline (No TL)	0.36	0.87	1.24	50.7
	Frozen All	0.39	0.63	**1.02**	54.4
	Frozen One	0.31	0.96	1.27	52.1
	Finetune All	0.32	0.77	1.09	**55.4**
Washing Dishes	Baseline (No TL)	0.74	0.77	1.51	25.5
	Frozen All	0.79	0.44	1.23	25.3
	Frozen One	0.47	1.07	1.54	**40.4**
	Finetune All	0.64	0.73	**1.38**	34.0
Car Passing by	Baseline (No TL)	0.32	0.66	**0.98**	**58.0**
	Frozen All	0.46	0.54	1.0	52.0
	Frozen One	0.31	0.8	1.1	55.7
	Finetune All	0.35	0.64	0.991	56.7

fewer events as compared to the baseline, causing the recall (not mentioned in the results, however visible by the increased deletion rate) to go much lower, thereby driving the F measure down regardless of a high precision. This is intuitive of the fact that the source network has learnt a dictionary of acoustic units from the source events but due to a limited overlap with the target events, it is detecting only few target events that seem to share these learnt basic building blocks.

2) Frozen One: When the first convolutional layer is kept frozen and the second and third convolutional layers are fine tuned along with the additional LSTM layer, the system produces more outputs, thereby increasing the number of insertions and also the number of true positives. This causes both the AEER and the F measure to increase as compared to the baseline.
3) Finetune All: However, when all the convolutional layers are fine tuned, though an improvement in AEER as compared to the previous 'Frozen one' scenario is achieved, this system does not improve the baseline system. Rather, it produces almost the same evaluation metrics as the baseline system without transfer learning. This indicates that the information of the source domain is lost if the parameters of all network layers are reestimated.

Going further, we investigate the performance of transfer learning with this source database on three exemplary target events: *Bird Singing*, which is also present in the source database, *Washing Dishes*, which has similar acoustic characteristics to the source event *rain*, and *Car Passing By*, whose acoustic characteristics do not match any sound event in the source database.

When looked closely into the evaluation metrics of these events with and without transfer learning, as presented n Table III, one can observe that the overall accuracy for detection of bird singing increases when the parameters from the source model are transferred to the target model. It is observable that just keeping the parameters of all the convolutional layers frozen, yields significant improvements (Frozen all). Moreover, fine tuning all convolutional layers produces the highest F measure but at the cost of an increased error rate (Finetune all). The AEER worsens for the 'Frozen one' scenario, which indicates that the parameters learnt by at least the first two convolutional layers complement each other. These results are in correspondence to the overall transfer learning results (Table II), where the AEER increases in the order of Frozen all, Finetune all and Frozen one experiments.

A similar improvement is observed in the target event *washing dishes* from the home scene. A significant drop in AEER occurs when the parameters of all the convolutional layers are kept frozen (Frozen all), indicating the transfer of knowledge from similar acoustic units in the source domain. However, it is for this event that freezing the parameters of the first convolutional layer and fine tuning the others (Frozen one), yields a 14.9% increase in F measure with only 3% increase in AEER as compared to no transfer learning. This can be attributed to a significantly lower deletion rate for this experiment as compared to others.

On the other hand, a negative transfer learning appears to occur for the target event *car passing by*, as the accuracy of detecting this event is slightly reduced with transfer learning. The main reason for this is probably the absence of acoustic units representative of such an event in the source database.

V. Conclusion

Transfer learning for enhancing the detection accuracy of real-life sound events has been investigated using a synthetic source database, in this work. The source domain is modelled with a convolutional recurrent neural network with the hypothesis that the convolutional layers will extract the basic acoustic units from source events, while delegating the duty of capturing the time dependencies within events to the following LSTM layer. This hypothesis, however, could only be partially verified in this work, mainly because the source database has less events and is not diverse enough to capture acoustic units or basic building blocks necessary to represent all target events. Nevertheless, this study shows that an improvement in acoustic event detection is obtained with transfer learning using the CRNN approach, if a substantial amount of source data, which is representative of a diverse set of events, is available. Future work will therefore focus on incorporating a large source database with diverse event content and studying the effects of freezing and finetuning the convolutional layer parameters more exhaustively.

Acknowledgment

This work has in part been supported by Deutsche Forschungsgemeinschaft (DFG) under contract number Ha 3455/15-1 within the research unit FOR 2457 "Acoustic Sensor Networks".

References

[1] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[2] C. Clavel, T. Ehrette, and G. Richard, "Events detection for an audio-based surveillance system," in *IEEE International Conference on Multimedia and Expo*, 2005, pp. 1306–1309.

[3] D. Zhang and D. Ellis, "Detecting sound events in basketball video archive," *Dept. Electronic Eng., Columbia Univ., New York*, 2001.

[4] G. Richard, T. Virtanen, J. Bello, N. Ono, and H. Glotin, "Introduction to the special section on sound scene and event analysis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1169–1171, 2017.

[5] P. Guyot, J. Pinquier, X. Valero, and F. Alias, "Two-step detection of water sound events for the diagnostic and monitoring of dementia," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2013, pp. 1–6.

[6] J. Schroeder, S. Wabnik, P. W. Van Hengel, and S. Goetze, "Detection and classification of acoustic events for in-home care," in *Ambient Assisted Living*, 2011, pp. 181–195.

[7] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *24th European Signal Processing Conference (EUSIPCO)*, 2016, pp. 1128–1132.

[8] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[9] A. Quattoni, M. Collins, and T. Darrell, "Transfer learning for image classification with sparse prototype representations," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

[10] Y. Zhu, Y. Chen, Z. Lu, S. J. Pan, G. R. Xue, Y. Yu, and Q. Yang, "Heterogeneous transfer learning for image classification," in *Association of Advancement of Artificial Intelligence (AAAI)*, 2011.

[11] W. Dai, Y. Chen, G. R. Xue, Q. Yang, and Y. Yu, "Translated learning: Transfer learning across different feature spaces," in *Conference on Neural Information Processing Systems (NIPS)*, 2008, pp. 353–360.

[12] D. Wang and T. F. Zheng, "Transfer learning for speech and language processing," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2015, pp. 1225–1237.

[13] P. Swietojanski, A. Ghoshal, and S. Renals, "Unsupervised cross-lingual knowledge transfer in DNN-based LVCSR," in *IEEE Spoken Language Technology Workshop (SLT)*, 2012, pp. 246–251.

[14] J. T. Huang, J. Li, D. Yu, L. Deng, and Y. Gong, "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7304–7308.

[15] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, "Cross-language transfer learning for deep neural network based speech enhancement," in *9th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2014, pp. 336–340.

[16] S. Mun, S. Shon, W. Kim, D. K. Han, and H. Ko, "Deep neural network based learning and transferring mid-level audio features for acoustic scene classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 796–800.

[17] A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu, and M. Omologo, "CLEAR evaluation of acoustic event detection and classification systems," in *International Evaluation Workshop on Classification of Events, Activities and Relationships*, 2006, pp. 311–322.

[18] S. Nakamura, K. Hiyane, F. Asano, T. Yamada, and T. Endo, "Data collection in real acoustical environments for sound scene understanding and hands-free speech recognition," in *Proceedings Eurospeech*, 1999, pp. 2255–2258.

[19] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proceedings of the 22nd ACM International Conference on Multimedia*, 2014, pp. 1041–1044.

[20] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Communication*, vol. 12, no. 3, pp. 247–251, 1993.

[21] S. M. Besen, "The European telecommunications standards institute: A preliminary analysis," *Telecommunications policy*, vol. 14, no. 6, pp. 521–530, 1990.

[22] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM International Conference on Multimedia*, 2015, pp. 1015–1018.

[23] H. Lim, M. J. Kim, and H. Kim, "Cross-acoustic transfer learning for sound event classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2504–2508.

[24] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, "The DARPA 1000-word resource management database for continuous speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1988, pp. 651–654.

[25] D. B. Paul and J. M. Baker, "The design for the wall street journal-based CSR corpus," in *Proceedings of the workshop on Speech and Natural Language*, 1992, pp. 357–362.

[26] N. Takahashi, M. Gygli, and L. Van Gool, "AENet: Learning deep audio features for video analysis," *arXiv preprint arXiv:1701.00599*, 2017.

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[30] D. Garcia Gasulla, F. Parés, A. Vilalta, J. Moreno, E. Ayguadé, J. Labarta, U. Cortés, and T. Suzumura, "On the behavior of convolutional nets for feature extraction," *arXiv preprint arXiv:1703.01127*, 2017.

[31] E. Çakır, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *arXiv preprint arXiv:1702.06286*, 2017.

[32] J. A. Stork, L. Spinello, J. Silva, and K. O. Arras, "Audio-based human activity recognition using non-markovian ensemble voting," in *26th IEEE International Symposium on Robot and Human Interactive Communication*, 2012.

[33] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[34] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[35] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.