Acoustic Event Classification Using Convolutional Neural Networks

Stefan Kahl[1], Hussein Hussein[2], Etienne Fabian[3], Jan Schloßhauer[3], Enniyan Thangaraju[1], Danny Kowerko[1], Maximilian Eibl[1]

Abstract: The classification of human-made acoustic events is important for the monitoring and recognition of human activities or critical behavior. In our experiments on acoustic event classification for the utilization in the sector of health care, we defined different acoustic events which represent critical events for elderly or people with disabilities in ambient assisted living environments or patients in hospitals. This contribution presents our work for acoustic event classification using deep learning techniques. We implemented and trained various convolutional neural networks for the extraction of deep feature vectors making use of current best practices in neural network design to establish a baseline for acoustic event classification. We convert chunks of audio signals into magnitude spectrograms and treat acoustic events as images. Our data set contains 20 different acoustic events which were collected in two different recording sessions combining human and environmental sounds. Our results demonstrate how efficient convolutional neural networks perform in the domain of acoustic event classification.

Keywords: Acoustic Event Classification, Acoustic Event Detection, Convolutional Neural Networks

1 Introduction

The demographic change in the European Union (EU) will increase the number of elderly people rapidly. Since older people suffer from several chronic conditions and often stay in their own homes, they require a long-term care solutions [St07]. The observation of the activities and health status of elderly people using automatic systems is very important, because the long-term nursing care at home is very expensive. The recognition of human activity can be realized using acoustic and/or visual information which is obtained by microphones and/or video cameras installed in the homes to assist the daily living by the collection of individual information for monitoring systems. Acoustic monitoring includes recognition and detection of acoustic events which indicate critical events for elderly or people with disabilities in Ambient Assisted Living (AAL) environments or by patients in hospital. Most people feel that acoustic monitoring has little impact on privacy in

[1] Chemnitz University of Technology, D-09107 Chemnitz, Germany, {stefan.kahl, enniyan.thangaraju, danny.kowerko, maximilian.eibl}@informatik.tu-chemnitz.de

[2] Department of Literary Studies, Free University of Berlin, 14195 Berlin, Germany, hussein@zedat.fu-berlin.de

[3] Intenta GmbH, D-09125 Chemnitz, Germany, {e.fabian, j.schlosshauer}@intenta.de

doi:10.18420/in2017_217

comparison to video-based surveillance. The automatic recognition of specific acoustic events in an audio stream is important for the analysis of human activities. Acoustic Event Classification (AEC) deals with isolated acoustic event segments (offline), Acoustic Event Detection (AED) includes the identification of timestamps as well as types of acoustic events in continuous audio streams (online or live recordings) [Te06].

Many applications implement classification and detection of different types of acoustic events. AEC is often utilized in scene recognition to recognize the location of scenes (e.g. outdoor or indoor) [Ri15]. Additionally, AEC is often applied to the field of AAL and health care environments, e.g. the recognition of activities that occur inside a bathroom [Ch05] and in critical and threatening health situations [Hu16b]. Another application for AEC is in smart homes to detect different types of events, for example, speech, walking steps, coffee spoon jingle and mouse clicks [Ts14] as well as in meeting room environments to detect events such as speech, paper work, chair movements and key jingle [Te06].

The process of event recognition is based on feature extraction and classification. Various features and classifiers have been proposed for the classification of acoustic events. Features in the frequency-domain, time-domain and cepstral features are extracted and used stand-alone or in combination. Mel Frequency Cepstral Coefficients (MFCCs) are the most popular speech perception features which are utilized with Hidden Markov Models (HMMs). However, speech features are not necessarily suitable for the classification of acoustic events [Zh08][CNK09]. The most common classification techniques used are HMMs [Ch05][Te06][Er06][TRF15], Gaussian Mixture Models (GMM) [Pe02][Ch06][Ra15], the Support Vector Machine (SVM) classifier [Te06][Ch06][Hu16b], the K-Nearest Neighborhood (KNN) classifier [Pe02][Er06][Ch06][Hu16b], and in recent years, neural networks [Pi15].

In 2012, Convolutional Neural Networks (ConvNets, CNNs) started to outperform traditional image processing methods one by one. Since then, almost every traditional technique has been rendered obsolete when it comes to semantic image understanding. With rapid evolution, more powerful neural net architectures have been introduced (e.g. [Re15] for object detection, [RFB15] for image segmentation and [KFF15] for image captioning). Adapting those architectures for the audio domain has become common practice, mostly due to the overwhelming success of CNNs for image processing. Training and classification of visual representations of raw audio signals has proven to be very effective for different scenarios such as bird identification in sound recordings [Ka17] or acoustic scene classification [HL16].

This paper is organized as follows: Section 2 gives an overview on the *localizeIT* project for object tracking and behavior analysis using audio-visual information. The selection of acoustic events and data acquisition are described in Section 3. Section 4 reviews the acoustic event classification with convolutional neural networks. The experimental results are shown in Section 5. Finally, conclusions and future work are presented in Section 6.

2 LocalizeIT

The purpose of the research project *localizeIT* (`http://www.localize-it.de`) is the localization and tracking of objects as well as the analysis of object behavior using acoustic and visual information. We installed passive sensors, including acoustic and optical sensors (cameras with stereo optics), inside the tracking area of an audio-video laboratory to locate and track objects in an indoor environment. The analysis of object behavior from acoustic information can be used and fused with the visual-based object behavior analysis. The five-year project is funded by the Federal Ministry of Education and Research in the program of Entrepreneurial Regions.

3 Data Collection

We introduce a novel data set for the purpose of acoustic event classification for AAL scenarios. The selection of acoustic events and the collection of related data are described in this section.

3.1 Selection of Acoustic Events

We focused on acoustic events, which are produced by people in critical situations, for example, calls for help or suffering injuries from collapsing on the floor. There are different scenarios in which these acoustic events occur. The basic use case is a single elderly person in a room where he/she dropped on the floor and calls for help. Another use case includes a confused person in police custody where he/she has to stay alone in a detention cell. A very important use case applies to hospital facilities where mentally ill patients show critical behavior in community areas.

For our research, we defined a number of acoustic events, including human and environmental sounds, which are characteristic for the described use cases. A total of 20 acoustic events were recorded in two different sessions. In every session we simulated and recorded ten different acoustic events:

- ***TUC***: The first session included the following acoustic events [Hu16b]: help (calling of the speech signal "Hilfe" in German), scream, whimper, crying, quiet (long period of silence detected between audio segments of music, speech or background noise), strikes (strikes with an open hand on a wood plate), vandalism (destruction of furniture, strikes on a wood plate and sometimes scream), downfall of plate (downfall of a wooden plate to the ground), dislocation of furniture (movement of furniture such as a commode on ground), and chair movement (movement of a chair on ground).
- ***Intenta***: The following acoustic events were recorded during the second session [Fa17]: movement of window handle, movement of door handle, pen on cup (strike

the cup with a pen), lighter (set fire with the lighter), sensor bracket, clapping, strikes of metal to metal, strikes on wall, trampling on carpet, and tongue clicking.

3.2 Data Acquisition

We recorded human sounds for the *TUC* data set with people of age 25 to 82. Both data sets are recorded in quiet environments at Chemnitz University of Technology and Intenta GmbH. The audio files were recorded with an sampling frequency of 44.1 kHz and a resolution of 16 bit. Two measurement microphones (Behringer ECM-8000) connected to the audio interface (Focusrite Scarlett 2i2) were used for the recording of audio data of the first set. The distance between the microphones was set to 20 cm and the distance to the acoustic source was varying between 20 to 30 cm. The second data set was recorded using three Schneider Intercom MIC Q400 microphones, which were positioned at about 2.5m height below the ceiling. The distance between the microphones and the signal source in this case ranged from 50 cm to 4 m for all signals except the movement of the window and the door handle. For this two events distance was constantly 4 m. The acquired audio data was manually annotated using the *Folker* toolkit [SS10].

3.3 Data Analysis

A total of 58 persons (11 female and 47 male) participated in the first recording session (*TUC*) to acquire acoustic events produced by humans, e.g. help, scream, whimper and crying. The run length of the acquired data for the first ten acoustic events is 54 minutes with a total of 1612 recording samples. The number of audio files is as follows: help (175), scream (129), whimper (176), crying (192), quiet (45), strikes (475), vandalism (78), downfall of plate (84), dislocation of furniture (126), and chair movement (132).

The size of acquired audio data for the second ten acoustic events (*Intenta*) is 4 minutes with a total of 704 recording samples. The number of audio files is as follows: movement of window handle (68), movement of door handle (56), pen on cup (58), lighter (100), sensor bracket (68), clapping (76), strikes of metal to metal (90), strikes on wall (56), trampling on carpet (70), and tongue clicking (62).

3.4 Data Post-processing

The annotated data set only contains the occurrence of single acoustic events without silence before and after the acoustic event. The length of most of the defined acoustic events is very short. The average length of selected acoustic events is as follows: strikes (0.15 sec), movement of window handle (0.45 sec), movement of door handle (0.25 sec), pen on cup

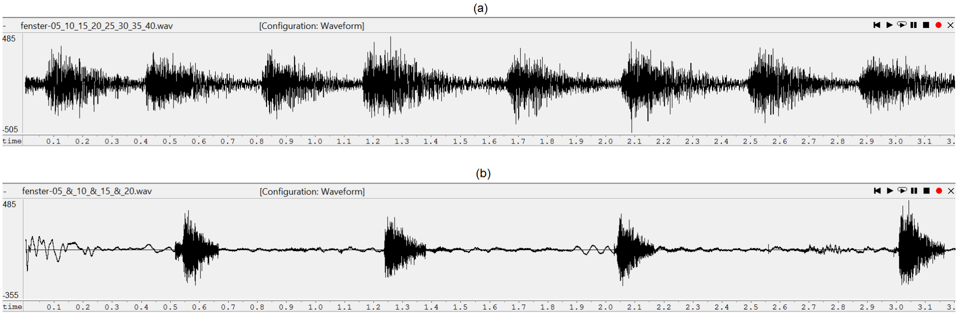

Fig. 1: Combination of acoustic events. a): original annotated acoustic events. b): post-processed acoustic events by adding a small pause before and after every acoustic event.

(0.40 sec), lighter (0.15 sec), strikes of metal to metal (0.40 sec), and tongue clicking (0.30 sec).

With real-world applications in mind, we decided to use recordings of at least three seconds of length. This way, very short recordings can be distributed over a longer period of time, which does reflect the expected distribution of acoustic events in realistic environments. We added a small pause with a random length between 0.2 and 0.5 sec before and after every acoustic event. Thereafter, we combined acoustic events of the same class to generate longer audio files with varying event distribution. Figure 1 shows the combination of original file with annotated acoustic events (a) as well as the combination of post-processed annotated acoustic events with a small pause before and after the acoustic event (b). The first four acoustic events in Figure 1 (a) are shown in Figure 1 (b) after adding the pauses. The resulting number of audio files after the combination of acoustic events is 615 and 177 for *TUC* and *Intenta*, respectively.

4 Experiments

Our experimental workflow consist of four main parts. First, we extract spectrograms from every audio recording using FFT in order to transfer the input data to the domain of image processing. Secondly, we extend our training set via data set augmentation. Thirdly, we train a convolutional neural network with a classic layout and best practice hyperparameter settings. Finally, we evaluate trained neural nets on a total of 156 test recordings using average prediction pooling.

4.1 Spectrogram Extraction

We decided to use magnitude spectrograms as visual representation of our training samples. Our experiments showed that large input resolutions with highly detailed signal transfor-

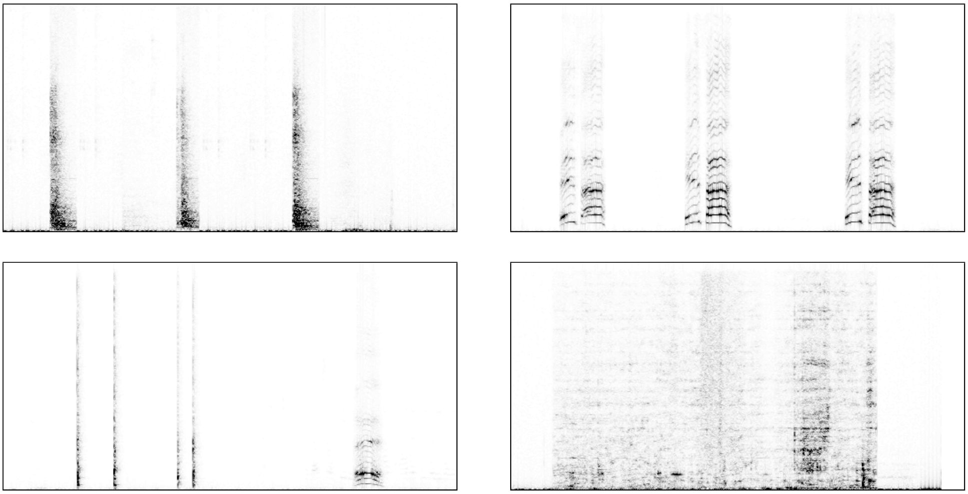

Fig. 2: Extracted magnitude spectrograms of samples for window handle (top left), calling for help (top right), vandalism (bottom left) and dislocation of furniture (bottom right). We use the framework *python_speech_features* for FFT with a window length of 0.02 and step size of 0.00585 for three-second chunks of each signal. We use a FFT length of 511 and do not crop high frequencies.

mations lead to better classification results. Most ConvNet architectures from benchmark evaluations are designed for relatively small input sizes. We decided to use non-square spectrograms with a resolution of 512x256 pixels (width x height, width being the time scale), each representing three-second chunks of the source file. Figure 2 shows some selected spectrograms for different acoustic events.

4.2 Data Augmentation

Choosing the right data set augmentation is vital to reduce the generalization error. Extending the training data aims to prevent overfitting due to a more diverse data set and should target properties of the test set underrepresented or missing in the training data. For the spectrogram domain, data set augmentation has to be selected carefully. Common geometric transformations such as horizontal flip, zoom, crop or shearing are not suitable as they might mask the original signal. We decided to use three augmentation methods: Pitch shifting vertical roll of 5%, time shifting horizontal roll of 50% and random Gaussian noise. Rolling the input image vertically or horizontally shifts the pixel values in the desired direction and thus preserves complete information as out-of-border pixel are added to the opposite image boundary. Neural nets usually learn to ignore random noise during training. We noticed that artificial noise helps to lower the generalization error despite the lack of heavy noise in the test data.

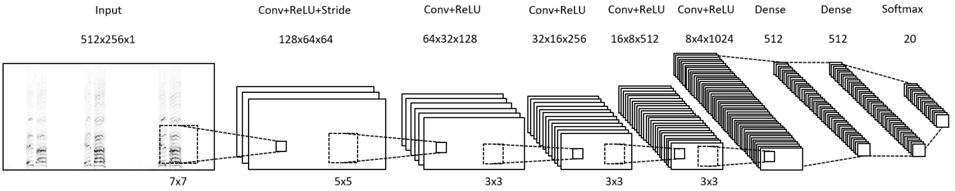

Fig. 3: Proposed ConvNet Architecture. All layers except for input and softmax layers use batch normalization, ReLu activation and are He-initialized. We use 2x2 strides in the first convolutional layer to cope with large inputs. MaxPooling of size 2 reduces spatial dimensions after every convolution.

We applied all augmentations at runtime, which significantly speeds up the training process and is more resource efficient. We implemented a multi-threaded batch loader, that operates during a forward-backward pass iteration using CPU idle time.

4.3 Training

Training neural nets with large input sizes is considerably harder and requires even better fine-tuning. We decided to follow common practices in ConvNet design and settled for a classic approach with no highway connections or shortcuts (Figure 3). Despite the simplicity of our net architecture, our simple model performed significantly better during our experiments than implementations of a ResNet-50 [He16] or DenseNet-32 [Hu16a]. This might be due to the homogeneous inputs in the spectrogram domain. Although our dataset features distinct classes with heterogeneous audio samples, 1-dimensional signal representations do not allow for much variance; most spectrogram pixels are blank or contain only little information. This observation is backed by the works of [Sp16] and [Da17].

Our eight-layer ConvNet uses He-initialization [He15] for all weights, batch normalization [IS15] and ReLu activation [NH10], except for input and softmax layers. We experimented with different kernel sizes in the first two convolutional layers and found large receptive fields in early layers beneficial for the overall classification performance. Therefore, we used 7x7 and 5x5 kernels for large inputs. Increasing the number of dense units led to heavy overfitting, so we decided to dial down the number of parameters and added more convolution filters instead. We conducted several experiments with fully convolutional neural nets, removing all densely connected layers from our architecture, which further reduces the parameter count. However, we were not able to achieve the same classification accuracy with this architecture. Further investigation has to show whether these architectures are compatible for the domain of acoustic classification.

We used a NVIDIA P6000 GPU for training of 55 epochs with a batch size of 32 for each data set (*TUC*, *Intenta* and both sets combined). Training took 7-9 sec per epoch; we used early stopping to find the best parameter setting. Reducing the learning rate during training is vital to ensure convergence of the optimization process. We used linear interpolation to lower the learning rate starting at 0.001 down to 0.000001 after each epoch. Despite the adaptive nature of the ADAM optimizer [KB14], this routine proved to be very effective.

We divided both data sets into a train (80%) and test split (20%). We used a 10% validation split of the training data to monitor the training process. Training data consisted of 1.725 spectrograms for the *TUC* set, 436 spectrograms for the *Intenta* set and 2.161 spectrograms for both sets combined.

4.4 Source Code

We implemented our code purely in Python using *NumPy*, *Theano* [Th16] and *Lasagne* [Di15] for models, objectives and solvers, *OpenCV* for image processing, *scikit-learn* for metrics and *Matplotlib* for visualizations. A refined and commented version of our code base is available for free use on GitHub[4]. We hope to provide a baseline system for further research regarding acoustic event classification and encourage research groups to contact us if any questions or remarks concerning the repository arise.

5 Results

Our test samples vary in run length and recording quality. However, since every test sample was randomly chosen from the original data set, our test set represents the original data distribution quite well. The proposed neural net architecture has proven to very efficient for our data set and validation results directly translate to the test files. Table 1 summarizes the results for three different test runs. We extracted consecutive three-second spectrograms using a two-second overlap for every test file. Combining the predictions of every spectrogram by simply applying average pooling led to excellent results and almost perfect predictions for every test sample.

	TUC	Intenta	Combined
Test Samples	123	33	156
Validation Accuracy	97,9%	100%	96,8%
Mean Average Precision	0,984	1,0	0,991
Precision at 1	0,967	1,0	0,981

Tab. 1: Results of our experiments for the *TUC* and *Intenta* training data as well as both sets combined.

Aside from the supposedly fitting model design, several circumstances have to be considered when interpreting the results. First, all test files contain only one class of acoustic events, which benefits softmax classifiers. Secondly, all test recordings where done in an artificial environment without background sounds and very low noise. Lastly, maintaining a very

[4] https://github.com/kahst/AcousticEventDetection

clean data set without distorted labels is key for successful training and classification. Most publicly available data sets do not comply with this condition. Our results clearly indicate that training convolutional neural networks for acoustic event detection is possible even with very limited data sets. Nonetheless, future experiments will have to show if these results hold up to more noisy environments, simultaneous sounds and most importantly unknown acoustic events without false detections.

6 Conclusion and Future Work

Real-time monitoring of audio signals demands a fast processing system, which can be used to extract the features of the signals and classify them effectively. Currently, this is the most crucial drawback of deep learning techniques as they consume many resources and require specialized hardware. Manufacturers are advancing their GPU technology further and eventually will incorporate enough computing power into portable devices. Until then, choices of portable or semi-portable hardware units for deep learning are limited. The NVIDIA Jetson TX2 provides the computing power sufficient for the proposed neural network presented in this paper. It is shipped with a developer board, which allows for rapid prototyping and implementation of software capable of acoustic event classification. Our conceptual workflow consists of training ConvNets on powerful GPUs and afterwards transferring trained models onto the TX2 where an audio stream is recorded and processed for acoustic events based on the classification of spectrograms. Specialized hardware for embedded applications like the Xilinx Zynq-7000 SoC ZC702 could be an alternative to the TX2 despite the lack of raw computing power because of its industry-standard FPGA Mezzanine Connectors. In any case, the quality of the detections will be influenced by the quality of the microphones, background noise and limited computing resources. However, the results presented in this paper indicate a good overall detection rate, our shallow neural net design and its small model size are well suited for less powerful devices.

Acknowledgments

This work is partially funded by the program of Entrepreneurial Regions InnoProfile-Transfer in the project group *localizeIT* (funding code 03IP608). The European Union and the European Social Fund for Germany also partially funded this research.

References

[Ch05] Chen, J.; Kam, A. H.; Zhang, J.; Liu, N.; Shue, L.: Bathroom Activity Monitoring Based on Sound. In: Proceedings of the Third International Conference on Pervasive Computing. PERVASIVE'05, Springer-Verlag, Munich, Germany, pp. 47–61, 2005.

[Ch06] Chu, S.; Narayanan, S.; Kuo, C.-C. J.; Mataric, M. J.: Where am I? Scene Recognition for Mobile Robots using Audio Features. In: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME). Toronto, Canada, pp. 885–888, Juli 2006.

[CNK09] Chu, S.; Narayanan, S.; Kuo, C.-C. J.: Environmental Sound Recognition with Time-Frequency Audio Features. Proc. of IEEE Transactions on Audio, Speech, and Language Processing, 17(6):1142–1158, August 2009.

[Da17] Dai, Wei; Dai, Chia; Qu, Shuhui; Li, Juncheng; Das, Samarjit: Very deep convolutional neural networks for raw waveforms. In: Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, pp. 421–425, 2017.

[Di15] Dieleman, Sander; Schlüter, Jan; Raffel, Colin; Olson, Eben; Sønderby, Søren Kaae; Nouri, Daniel et al.: , Lasagne: First release., August 2015.

[Er06] Eronen, A. J.; Peltonen, V. T.; Tuomi, J. T.; Klapuri, A. P.; Fagerlund, S.; Sorsa, T.; Lorho, G.; Huopaniemi, J.: Audio-Based Context Recognition. Proc. of IEEE Transactions on Audio, Speech, and Language Processing, 14(1):321 – 329, January 2006.

[Fa17] Fabian, E.: Erweiterung einer SVP-Plattform um eine Akustikkomponente und deren Anwendung zur Klassikation und Lokalisierung von Geräuschquellen. Master thesis, TU Chemnitz, February 2017.

[He15] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing; Sun, Jian: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034, 2015.

[He16] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing; Sun, Jian: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778, 2016.

[HL16] Han, Yoonchang; Lee, Kyogu: Acoustic scene classification using convolutional neural network and multiple-width frequency-delta data augmentation. arXiv preprint arXiv:1607.02383, 2016.

[Hu16a] Huang, Gao; Liu, Zhuang; Weinberger, Kilian Q; van der Maaten, Laurens: Densely connected convolutional networks. arXiv preprint arXiv:1608.06993, 2016.

[Hu16b] Hussein, Hussein; Ritter, Marc; Manthey, Robert; Schloßhauer, Jan; Fabian, Etienne; Heinzig, Manuel: Acoustic Event Classification for Ambient Assisted Living and Healthcare Environments. In (Jokisch, Oliver, ed.): Proceedings of the 27th Conference on Electronic Speech Signal Processing (ESSV). volume 81 of Studientexte zur Sprachkommunikation, TUDpress, Leipzig, Germany, pp. 271–278, March 2016.

[IS15] Ioffe, Sergey; Szegedy, Christian: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning. pp. 448–456, 2015.

[Ka17] Kahl, Stefan; Wilhelm-Stein, Thomas; Hussein, Hussein; Klinck, Holger; Kowerko, Danny; Ritter, Marc; Eibl, Maximilian: Large-Scale Bird Sound Classification using Convolutional Neural Networks. Working notes of CLEF, 2017.

[KB14] Kingma, Diederik; Ba, Jimmy: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[KFF15] Karpathy, Andrej; Fei-Fei, Li: Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3128–3137, 2015.

[NH10] Nair, Vinod; Hinton, Geoffrey E: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10). pp. 807–814, 2010.

[Pe02] Peltonen, V.; Tuomi, J.; Klapuri, A.; Huopaniemi, J.; Sorsa, T.: Computational Auditory Scene Recognition. In: Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Florida, USA, May 2002.

[Pi15] Piczak, K. J.: Environmental Sound Classification with Convolutional Neural Networks. In: IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP). Boston, MA, USA, pp. 1–6, September 2015.

[Ra15] Raboshchuk, G.; Jančovič, P.; Nadeu, C.; Lilja, A. P.; Köküer, M.; Mahamud, B. M.; Veciana, A. R.: Automatic Detection of Equipment Alarms in a Neonatal Intensive Care Unit Environment: A Knowledge-Based Approach. In: Proc. of Sixteenth Annual Conference of the International Speech Communication Association (Interspeech 2015). Dresden, Germany, September 2015.

[Re15] Ren, Shaoqing; He, Kaiming; Girshick, Ross; Sun, Jian: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99, 2015.

[RFB15] Ronneberger, Olaf; Fischer, Philipp; Brox, Thomas: U-net: Convolutional networks for biomedical image segmentation. arXiv preprint arXiv:1505.04597, 2015.

[Ri15] Ritter, M.; Rickert, M.; Chenchu, L. J.; Kahl, S.; Herms, R.; Hussein, H.; Heinzig, M.; Manthey, R.; Richter, D.; Bahr, G. S.; Eibl, M.: Technische Universität Chemnitz at TRECVID Instance Search 2015. In: Proceedings of TRECVID Workshop. Gaithersburg, MD, USA, November 2015.

[Sp16] Sprengel, Elias; Jaggi, Martin; Kilcher, Yannic; Hofmann, Thomas: Audio Based Bird Species Identification using Deep Learning Techniques. In: CLEF (Working Notes). pp. 547–559, 2016.

[SS10] Schmidt, T.; Schütte, W.: FOLKER: An Annotation Tool for Efficient Transcription of Natural, Multi-party Interaction. In (Calzolari, N.; Choukri, K.; Maegaard, B.; Mariani, J.; Odijk, J.; Piperidis, S.; Rosner, M.; Tapias, D., eds): Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). Valletta, Malta, pp. 2091 – 2096, May 2010.

[St07] Staff, European Commission: , Europe's demographic future: Facts and figures on challenges and opportunities, October 2007.

[Te06] Temko, A.; Malkin, R.; Zieger, C.; Macho, D.; Nadeu, C.; Omologo, M.: Acoustic Event Detection and Classification in Smart-Room Environment: Evaluation of CHIL Project Systems. In: Proc. of IV Jornadas en Tecnología del Habla - The IV Biennial Workshop on Speech Technology. Zaragoza, Spain, November 2006.

[Th16] Theano Development Team: Theano: A Python framework for fast computation of mathematical expressions. arXiv e-prints, abs/1605.02688, May 2016.

[TRF15] Transfeld, P.; Receveur, S.; Fingscheidt, T.: An Acoustic Event Detection Framework and Evaluation Metric for Surveillance in Cars. In: Proc. of 16th Annual Conference of the International Speech Communication Association (Interspeech 2015). Dresden, Germany, September 2015.

[Ts14] Tsiami, A.; Rodomagoulakis, I.; Giannoulis, P.; Katsamanis, A.; Potamianos, G.; Maragos, P.: ATHENA: a Greek Multi-Sensory Database for Home Automation Control. In: Proc. of 15th Annual Conference of the International Speech Communication Association (Interspeech 2014). Singapore, pp. 1608–1612, September 2014.

[Zh08] Zhuang, X.; Zhou, X.; Huang, T. S.; Hasegawa-Johnson, M.: Feature Analysis and Selection for Acoustic Event Detection. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP). Caesars Palace, Las Vegas, Nevada, USA, pp. 17–20, 2008.