IAEO3 - COMBINING OPENL3 EMBEDDINGS AND INTERPOLATION AUTOENCODER FOR ANOMALOUS SOUND DETECTION

Technical Report

Sascha Grollmisch[1,2]*, David Johnson*[1]*, Jakob Abeßer*[1]*, Hanna Lukashevich*[1]

[1] Fraunhofer IDMT, Industrial Media Applications, Ilmenau, Germany
{goh, jsn, abr, lkh}@idmt.fraunhofer.de
[2] Technische Universität Ilmenau, Institute of Media Technology, Ilmenau, Germany
sascha.grollmisch@tu-ilmenau.de

ABSTRACT

In this technical report, we present our system for task 2 of the IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE2020 Challenge): Unsupervised Detection of Anomalous Sounds for Machine Condition Monitoring. The focus of this task is to detect anomalous industrial machine sounds using an acoustic quality control system, which is only trained with sound samples from the normal (machine) condition. The dataset covers a variety of machines ranging from stable sound sources such as car engines, to transient sounds such as opening and closing valves. Our proposed method combines pre-trained OpenL3 embeddings with the reconstruction error of an interpolation autoencoder using a gaussian mixture model as the final predictor. The optimized model achieved 88.5% AUC and 76.8% pAUC on average over all machines and types provided with the development dataset, and outperformed the published baseline by 14.9% AUC and 17.2% pAUC.

Index Terms— Anomalous sound detection, industrial sound analysis, neural networks, openl3

1. INTRODUCTION

The goal of anomalous sound detection (ASD) is to identify anomalous sounds when only sounds of the "normal" condition are available beforehand. An important application field is Industrial Sound Analysis (ISA) [1] where these methods are integrated into acoustic quality control systems. Such systems can be used for predictive maintenance where failures need to be detected during machine runtime, or for end-of-line testing in which case no faulty products should be shipped to the customer. Since creating datasets with all possible faults is costly or sometimes impossible for every machine and type, ASD systems can be trained with audio samples recorded in the normal machine state. These training examples are often cheap to collect. This makes ASD a promising practical solution for non-invasive fault detection. Recently, two datasets ToyADMOS [2] and MIMII [3] have been published covering a wide variety of machines and sound characteristics. These datasets have been combined for the DCASE2020 task 2 [4]. The dataset includes sounds from toy cars, toy conveyors, fans, pumps, sliders, and valves. For each machine, recordings of up to four different machine types are included in the development set. These types differ slightly in their sound properties. Each recording was mixed with different environmental noises to simulate real environments. The provided ASD baseline system uses a feed forward neural network autoencoder with four hidden layers with 128 units each in the encoder/decoder and a bottleneck layer of size 8. On the input side, five mel-scaled spectrogram frames were concatenated and the reconstruction error was used as the prediction value for anomalous sounds. As evaluation metrics the area under the receiver operating characteristic curve (AUC) and the partial-AUC (pAUC) are used. pAUC is especially important since it demonstrates the performance of the system with a low false-positive-rate (FPR), and is set to 0.1 for this task. The baseline system achieved 73.6% AUC and 59.6% pAUC averaged over all machines and types using the provided evaluation code.

*This work has been partly supported by the German Research Foundation (BR 1333/20-1, CA 2096/1-1).

2. RELATED WORK

Pre-training neural networks on large datasets and using the output of intermediate layers as feature representations for training new classifiers has proven to be an effective strategy for several datasets with different classification tasks [5][6][7]. These so-called embeddings can be achieved by pre-training neural networks in a fully supervised manner or by creating an auxiliary task for self-supervised training.

OpenL3 embeddings were trained in a self-supervised fashion and published in [6] as an extension to the *L3-Net* [8]. Here, the auxiliary task is to classify the similarity of video and audio inputs using two separate neural network branches. The output of the audio branch is used for extracting embeddings of new datasets. Training a linear Support Vector Machine (SVM) on these embeddings has achieved good results on several different datasets which also contained ISA tasks such as classifying the operational state of an electric engine [9].

The autoencoder provided as the baseline system for this task used 5 time frames as an input and had to reconstruct the same data as the target. Suefusa et al. [10] proposed an alternative way of training an autoencoder by leaving out the middle frame on the input side and making it the target for the output. The authors showed that the Interpolation Deep Neural Network (IDNN) outperformed autoencoder-based approaches for non-stationary machine sounds such as valves. These valves are also part of DCASE2020 task 2. Since it is still an autoencoder, we name it interpolation autoencoder (IAE) in our proposed system to emphasize the difference to other neural network architectures.

Table 1: Parameter settings for general IAEO3 system and specific settings for each machine in IAEO3_opt. SR is sampling rate in kHz. OL3 specifies if OpenL3 has been used in the optimized system.

Machine	FFT	Hop	Mel	SR	IAE Encoder	OL3
IAEO3	1024	512	128	16	[256,128,64,32]	yes
Car	1024	512	128	16	[256,128,64,32]	yes
Conveyor	512	256	128	16	[256,128,64,32]	yes
Fan	1024	512	256	4	[128,64,32]	yes
Pump	2048	1024	128	16	[256,128,64,32]	yes
Slider	1024	512	128	16	[256,128,64,32]	yes
Valve	1024	512	128	16	[256,128,64,32]	no

3. PROPOSED METHODS

For this task, we submit two systems, a general system with the same hyperparameters for all machines and one system with machine specific hyperparameters. This section explains the general architecture of our submitted system and the hyperparameter adjustments that have been made for the machine-specific submission.

3.1. General Approach

As shown in Figure 1, the proposed **IAEO3** system combines two parallel branches. In the first branch, OpenL3 embeddings are extracted for each file using the published model trained on data from the environmental domain with 512 output features.[1] The 512-dimensional embedding vectors are averaged over the entire recording and normalized to zero mean and standard deviation of 1 for all files. We apply Principal Component Analysis (PCA)[2] to decorrelate the averaged embedding vectors and keep the first 50 principal components as features. This forms the first set of features for the feature vector of the final anomaly predictor.

In the second branch, we compute the reconstruction error of an IAE model for each machine and type.[3] Mel-spectrograms are extracted from the audio files using a Short-Time Fourier Transform (STFT) of 1024 samples, a hop size of 512 samples, and 128 Mel bands.[4] For the input, five consecutive spectral frames are concatenated and the middle frame is removed and used as the reconstruction target. The IAE consists of four feed forward layers with 256, 128, 64, and 32 units in the encoder, respectively. The decoder attempts to reconstruct the left-out frame using 64, 128, and 256 units per layer. This leads to a total of roughly 250k parameters to train. The the OpenL3 weights are frozen and do not change during training.

Each IAE was trained for 30 epochs using Adam optimizer [11] with a learning rate of 0.001. Afterwards, the mean squared error (MSE) of the reconstructed IAE output was calculated for each of T frames on the training data. To obtain a single output per file, several statistical parameters were calculated on the MSE over all time

[1] https://pypi.org/project/openl3/

[2] Implementation from scikit-learn (0.22.2): https://scikit-learn.org/

[3] Using Keras (keras.io) and Tensorflow (www.tensorflow.org)

[4] Implementation from librosa (0.7.2): https://librosa.github.io/

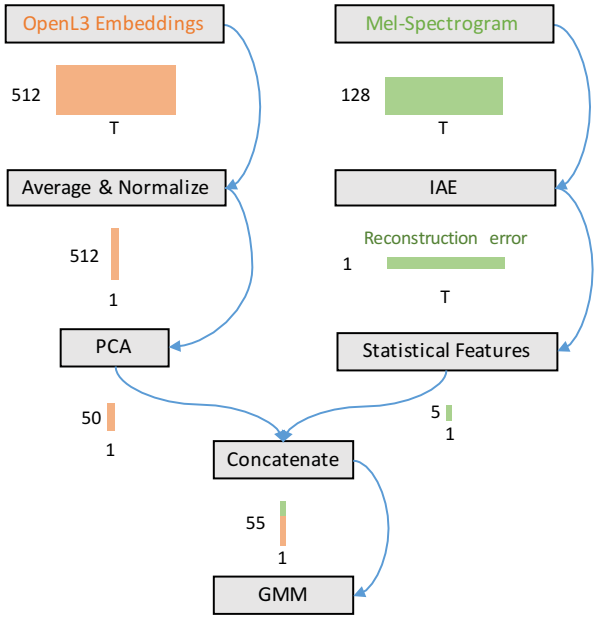

Figure 1: Flow-chart of the proposed two-branch IAE03 system for anomaly detection. T denotes the number of frames of a given input file.

frames of each file: mean, standard deviation, median, maximum, and interquartile range.

These five statistical values were combined with the 50 features extracted from the OpenL3 embeddings to form the final feature vector. Using the feature vectors of all files of the training set a Gaussian mixture model (GMM)[2] with three components was trained as the final predictor. This was later applied on the test data to perform the anomaly detection with the weighted log probabilities as anomaly score. To account for randomness during training of the IAE, the training process was repeated five times for each machine/type and the minimum anomaly score out of five inference runs was taken for the final submission.

3.2. Machine-specific adjustments

Since the machines covered in the dataset strongly vary in their specific acoustic properties and corresponding anomalies, one parameter setting seemed unlikely to be optimal. Therefore, a second submitted system (IAEO3_opt) was designed in which the parameters were set individually for each machine using the results from the development set. The detailed parameters are shown in Table 1. For valves the system was simplified by discarding OpenL3 embeddings and the GMM predictor since the IAE itself already performed well using the maximum of the reconstruction error per file.

4. RESULTS

The results on the development dataset are shown in Table 2. The general IAEO3 systems outperforms the baseline on all machines both in AUC and pAUC with the highest performance gains for cars and valves. Conveyors and fans on the other hand still have lot of room for improvement. Optimizing the parameters in IAEO3_opt

Table 2: AUC and pAUC values in % for both submitted systems and the baseline on the development set.

Machine	ID	IAEO3_opt	IAEO3	Baseline
Car	1	93.9, 82.5	93.9, 82.5	81.4, 68.4
Car	2	96.5, 89.7	96.5, 89.7	86.0, 77.7
Car	3	87.4, 69.6	87.4, 69.6	63.3, 55.2
Car	4	99.5, 97.3	99.5, 97.3	84.5, 69.0
Avg.		**94.3, 84.8**	**94.3, 84.8**	78.8, 67.6
Conveyor	1	85.1, 71.8	84.2, 68.5	78.1, 64.3
Conveyor	2	71.5, 55.8	70.3, 55.5	64.2, 56.0
Conveyor	3	83.4, 66.5	79.4, 62.8	75.4, 61.0
Avg.		**80.0, 64.7**	78.0, 62.3	72.5, 60.4
Fan	0	65.5, 53.9	64.9, 52.67	54.4, 49.4
Fan	2	83.3, 64.4	80.9, 63.0	73.4, 54.8
Fan	4	71.4, 62.1	67.2, 54.3	61.6, 53.3
Fan	6	98.1, 90.2	96.6, 82.8	73.9, 52.4
Avg.		**79.6, 67.6**	77.4, 63.2	65.8, 52.4
Pump	0	84.4, 62.9	82.6, 60.0	67.2, 56.7
Pump	2	77.8, 68.8	75.9, 64.6	61.5, 58.1
Pump	4	98.0, 90.9	98.4, 92.9	92.9, 67.1
Pump	6	78.9, 66.3	79.1, 65.8	74.6, 58.0
Avg.		**84.8, 72.2**	84.0, 70.8	72.9, 60.0
Slider	0	95.9, 79.6	95.9, 79.6	96.2, 81.4
Slider	2	84.0, 60.1	84.0, 60.1	79.0, 63.7
Slider	4	97.9, 88.9	97.9, 88.9	94.3, 72.0
Slider	6	85.9, 54.6	85.9, 54.6	69.6, 49.0
Avg.		**90.9, 70.8**	**90.9, 70.8**	84.8, 66.5
Valve	0	100.0, 100.0	100.0, 99.9	68.8, 51.7
Valve	2	99.7, 98.6	99.4, 97.0	68.2, 51.8
Valve	4	99.8, 99.0	99.0, 95.8	74.3, 52.0
Valve	6	98.8, 94.2	93.8, 79.90	53.9, 48.4
Avg.		**99.6, 97.9**	98.1, 93.1	66.3, 51.0
Total avg.		**88.5, 76.8**	87.5, 74.7	73.6, 59.6

slightly improves AUC from 87.5% to 88.5% and pAUC from 74.7% to 76.7%. This indicates that the IAEO3 approach needs to be extended for better results on the problematic cases such as fans instead of only tuning the hyperparameters. For the recorded valves and corresponding anomalies, the IAE by itself already leads to nearly perfect results.

5. CONCLUSIONS

We proposed an anomaly detection system which combines OpenL3 embeddings with an interpolation autoencoder for the DCASE2020 task 2. This approach has outperformed the baseline system on all machines in the development set and improved the baseline AUC from 73.6% to 88.5% and pAUC from 59.6% to 76.8%. Furthermore, the proposed system achieved good results for stationary and non-stationary sounds.

6. REFERENCES

[1] S. Grollmisch, J. Abeßer, J. Liebetrau, and H. Lukashevich, "Sounding Industry: Challenges and Datasets for Industrial Sound Analysis," in *Proceedings of the 27th European Signal Processing Conference (EUSIPCO)*, A Coruña, Spain, 2019.

[2] Y. Koizumi, S. Saito, H. Uematsu, N. Harada, and K. Imoto, "ToyADMOS: A dataset of miniature-machine operating sounds for anomalous sound detection," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, 2019, pp. 308–312. [Online]. Available: https://ieeexplore.ieee.org/document/8937164

[3] H. Purohit, R. Tanabe, T. Ichige, T. Endo, Y. Nikaido, K. Suefusa, and Y. Kawaguchi, "MIMII Dataset: Sound dataset for malfunctioning industrial machine investigation and inspection," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, New York, NY, USA, November 2019, pp. 209–213. [Online]. Available: http://dcase.community/documents/workshop2019/proceedings/DCASE2019Workshop_Purohit_21.pdf

[4] Y. Koizumi, Y. Kawaguchi, K. Imoto, T. Nakamura, Y. Nikaido, R. Tanabe, H. Purohit, K. Suefusa, T. Endo, M. Yasuda, and N. Harada, "Description and discussion on DCASE2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring," in *arXiv e-prints: 2006.05822*, June 2020, pp. 1–4. [Online]. Available: https://arxiv.org/abs/2006.05822

[5] A. Kumar, M. Khadkevich, and C. Fugen, "Knowledge Transfer from Weakly Labeled Audio Using Convolutional Neural Network for Sound Events and Scenes," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Calgary, Alberta, Canada, 2018, pp. 326–330.

[6] J. Cramer, H.-h. Wu, J. Salamon, and J. P. Bello, "Look, Listen, and Learn More: Design Choices for Deep Audio Embeddings," in *Proceedings of the IEEE International Conference on Audio, Speech, and Signal Processing (ICASSP)*, Brighton, UK, 2019, pp. 3852–3856.

[7] K. Choi, G. Fazekas, M. Sandler, and K. Cho, "Transfer learning for music classification and regression tasks," *arXiv:1703.09179*, 2017.

[8] R. Arandjelovic and A. Zisserman, "Look, Listen and Learn," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 609–617.

[9] S. Grollmisch, E. Cano, C. Kehling, and M. Taenzer, "Analyzing the Potential of Pre-Trained Embeddings for Audio Classification Tasks," in *27th European Signal Processing Conference (EUSIPCO)*, Amsterdam, The Netherlands, 2020.

[10] K. Suefusa, T. Nishida, H. Purohit, R. Tanabe, T. Endo, and Y. Kawaguchi, "Anomalous Sound Detection Based on Interpolation Deep Neural Network," in *Proceedings of the IEEE International Conference on Audio, Speech, and Signal Processing (ICASSP)*, Barcelona, Spain, 2020, pp. 271–275.

[11] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.