A prototype deep learning system for the acoustic monitoring of intensive care patients

Athanasios Lykartsis
Audio Communication Group
TU Berlin
Berlin, Germany
athanasios.lykartsis@tu-berlin.de

Markus Hädrich
Audio Communication Group
TU Berlin
Berlin, Germany
markus.haedrich@tu-berlin.de

Stefan Weinzierl
Audio Communication Group
TU Berlin
Berlin, Germany
stefan.weinzierl@tu-berlin.de

***Abstract*—We present a prototype system for the acoustic monitoring of artificially ventilated patients in intensive care. A device placed in the patient room detects sounds indicating an emergency situation and notifies a pager of the care staff. The staff can react more quickly and take appropriate action, as well as provide feedback on the prediction for continual learning. A microphone array with adaptive beamforming and an integrated microcomputer is employed, autonomously performing recording, audio preprocessing as well as deep learning based inference. The training dataset originates from a variety of patients and spatial and sonic environments, accommodating for different patterns of background noise and distortions. Mel spectrograms of short length are extracted and used for training a convolutional neural network. An initial evaluation of the system shows an accuracy of 80% for a binary, balanced dataset. The system is deployed in several intensive care facilities and can easily be adapted to other types of medically relevant sounds.**

***Index Terms*—acoustic monitoring, beamforming, convolutional neural network, mel spectrograms**

I. Introduction

Acoustic monitoring has gained importance in recent years, including a variety of applications related to the detection and evaluation of audio events, in predictive maintenance, surveillance or home assistance devices. Acoustic monitoring for medical applications, however, has not seen comparable progress, mostly due to the challenge of reliably detecting complex medical conditions based on sound and the lack of sufficient open source data available for training. The current project wanted to address this gap and demonstrate that it is possible to build an integrated audio processing system which can be used to detect emergency situations requiring intervention by medical staff. To achieve that, appropriate data for a specific medical application was gathered in cooperation with an industrial and a care service provider in order to develop a deep learning framework and a prototype hardware device on which the model may run in order to detect emergency situations based on audio data only.

The specific application includes patients who do not have the ability to move or speak, and who occasionally have a build-up of internal secretions in their pulmonary-throat system. In such cases, the patients have to be suctioned by intensive care staff. When patients experience this situation, their normal breathing noises change so that a characteristic wheezing sound ensues, indicating the existence of an emergency situation to be addressed. If the nursing staff, however, is not standing right next to the patient at that moment, it is difficult to recognize such a build-up early enough to avoid patient distress. In order to detect this category of sounds, a convolutional neural network (CNN) was employed, trained on sound data recorded in the patient care institutions and labeled by experienced nursing staff. This data comes from different patients and different spatial and sonic environments to ensure the generalizability of the model. The trained model has the task of classifying sounds as critical or non-critical.

The acoustic monitoring device was developed as an integrated hardware and software system, consisting of

- an audio preprocessing and training system, which can be used to train models on a standard workstation,
- a prediction module, which runs on the microcomputer device itself and is used for inference only, and
- a hardware device for recording, including beamforming with a microphone array, basic audio processing and process handling, as well as wireless communications and system calls.

II. State of the Art

Acoustic monitoring can be considered as a special case of auditory scene analysis and audio event detection. Most works in this area employ CNNs for detection of human and machine sounds [1] or Convolutional Recurrent Neural Networks (CRNNs) [2] used for tasks with a strong temporal dependence between the samples and providing reliable predictions even for datasets which are weakly labeled.

For acoustic monitoring, similar approaches have been used for animal species identification [3] or for industrial sounds such as rail condition monitoring [4], [5]. There are very few reports on medical applications based on audio data. A recent study aiming at detection of COVID19 from voice [6] uses transformer networks and RNNs, achieving promising results. Due to the high relevance of this subject, similar COVID19 related studies have emerged in the latter time, also using RNNs to detect COVID19 from basic acoustic features such as MFCCs derived from acoustic signals, such as coughing or

This project has been funded by the German Federal Ministry for Economic Affairs and Energy.

breathing, with high accuracy [7]; by crowdsourcing patient data and comparing respiratory sounds to those of healthy individuals, [8] could use transfer learning and basic audio features to reach an Area-Under-Curve (AUC) higher than 80%; and fully combined software and mobile systems [9], comparing the results of an acoustic cough probe to database samples in real time, in order to rapidly detect an infectious situation. Furthermore, our work has a similar goal to the one in [10] where a system for the detection of asthmatic patients is presented, while in terms of methodology we are aligned with more recent approaches using space-time audio representations and deep learning to detect tuberculosis-related cough samples [11]. Since the signals to be detected in our case are in essence a special category of breathing noise, our task is similar to the one in [12], where a combination of spectrogram and transducer data was used in conjunction with CNNs and RNNs to detect different categories of abnormal breathing signals.

Systems similar to those mentioned above have also been coupled to smart hardware devices or sensor networks, in order to expand their field of application. These include approaches involving IoT-Cloud technologies, specifically using a distributed network of sensors to gather Electro-EncephaloGraphy (EEG) and other data from patients and employing deep learning to make a prediction about an underlying pathology [13] such as the detection of epileptic seizures [14]. Another approach [15] uses a wireless sensor network, receiving a variety of physiological data (e.g., blood oxygenation level and pulse rate) from end-devices which can be used to provide an estimation of whether there is an emergency situation for a patient in a hospital setting.

In comparison to the papers mentioned above, our study has the novelty of combining a very light but versatile microcomputer system with networking capabilities, which employs an expandable and improvable deep learning system for classifying emergency situations based only on audio.

III. Methods

A. Overview

The proposed system was developed for intensive care units organized as shared apartments rather than in hospitals, with one device per room. It is designed to communicate with a central workstation using a local wireless-based network.

Each device will function and record continuously in one room to provide a prediction for an audio slice just seconds after it has been recorded. Because the on-site situation differs from patient room to patient room, and it is not possible to guarantee a fixed positioning of the device in every room, we decided to use a beamforming microphone array to be adapted to the specific acoustic scene and to reduce ambient noise. The recorded and preprocessed audio is forwarded to the data processing and ML-based inferencing part of the system, which performs a Short-Time-Fourier-Transform (STFT) and creates Mel spectrograms used an an input to the inference system. The latter uses a pre-trained CNN model and provides a binary output together with its probability, indicating whether

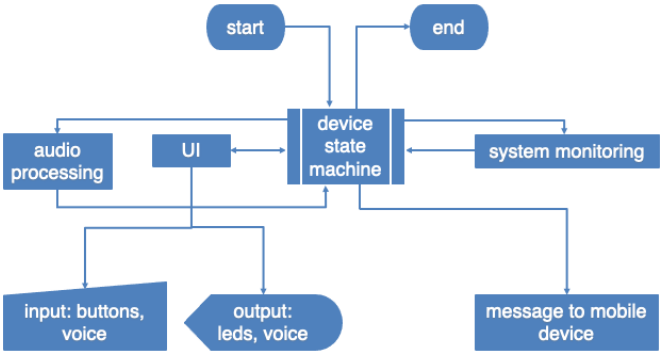

Fig. 1. System components of the Microcomputer.

there is an emergency or not. This result is forwarded to a custom-made pager, by which the staff is notified. After medical interventions have been performed, the staff can press a button on the device to provide feedback necessary to improve the model using continual learning methods, but also to facilitate an overview of the data over time from the staff via a graphical user interface (GUI). The device also offers a locally executed hands-free user interaction by voice-controlled commands and feedback on the current device status, which is also displayed visually via the integrated LED ring. An overview of the system is shown in Figure 1. A medical device must not have undefined states. A device state machine designed as borg idiom acts as central controller unit. States can be *listening, suspended, error, setup* and *alarm*. In setup mode e.g. the direction of the main lobe can be adjusted. As a hands-free option a local voice detection [16] runs at the whole program life cycle. No recordings are made here, no data is transmitted over the internet and only commands preceded by an artificial hotword in the sequence *Hotword + Command + Parameter* are handled and confirmed with acoustic and visual feedback.

B. Beamforming

The array dimensions determine the limits of spatial acoustic performance. A low directivity can be expected for frequencies with wavelengths that are relatively large in relation to the extension of the microphone array. On the other hand, for frequencies with wavelengths that are small compared to the array diameter, spatial aliasing will occur leading to unwanted constructive interference for sound incidence directions that are not of interest leading to decreasing directivity. The frequency range unaffected by both restrictions lies between 1 and 3.7 kHz. Figure 2 shows the typical spectral and temporal shape of the signal of interest, the rattling sound of secretion accumulations that impair breathing, and the directivity index of the microphone array.

It can be seen that the frequency region of increased directivity corresponds quite well to the spectral shape of the signals of interest. This can be the case when where a high noise rejection from other angles of incidence is most required. Considering the limited resources of an Edge ML

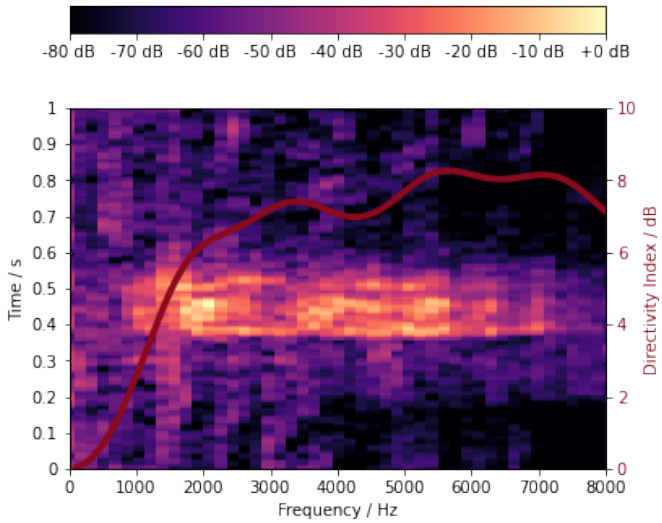

Fig. 2. Spectrogram of a typical positively labeled audio example (wheezing breath, 1 s) and directivity index over frequency for the 6-channel microphone array with a radius of $r = 0.047$ m.

system, a delay-and-sum (DS) algorithm was chosen, which is robust and not very demanding in terms of the required computational power. The audio recording is organized in a modular, chained manner in several threads. To improve access times the recording directory is placed in a ramdisk, which acts as a LiFo buffer for 1 s audio slices. From there, the inference module can load the audio slice in single-digit milliseconds as soon as the recording is finished. Both processes - beamformed recording and inferencing - run in parallel.

C. Audio Processing and Classification

For classification, we used a basic CNN architecture with an increasing number of nodes (16, 32, 64, 128) with filters of size (8, 4) pixels (in order to perform a frequency reduction) followed by two fully connected layers of a 200 and 100 nodes, adding up to a total number of just over 1.5M parameters, which makes up for a very lightweight system. After every convolutional layer batch normalization was employed before a RELU activation function followed by (2, 2) max pooling. For the fully connected layers, no dropout was used and the activation was RELU apart from the last layer which uses softmax activation for the two output nodes. The complete architecture is shown in Fig. 3. The model was trained with a learning rate of 3 x 10^{-5} for a total of 12 epochs, until we could not observe a further change in validation loss.

The full training pipeline and prediction system consists of two parts, one for training on a standard workstation, and one for the inference on the hardware device.

After training a model, it becomes serialized as a tf-lite file and the inference system only runs a prediction on the Mel spectrogram from the currently recorded sound file. The prediction is performed in about 215 ms, the longest part of the processing (ca. 80%) being the spectrogram extraction.

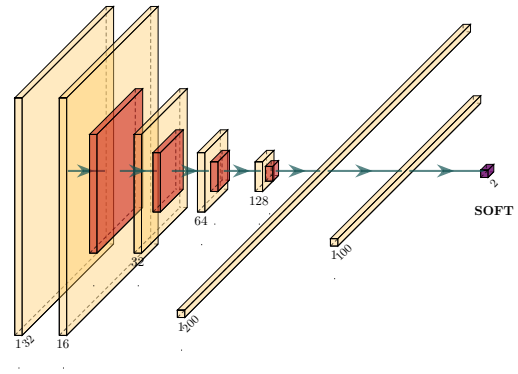

Fig. 3. Architecture for the employed CNN. The system consists of 4 convolutional layers with rectangular filters, followed by two fully connected layers and a binary output softmax.

D. Data

For the model training we created a custom dataset of sounds recorded in the patient care institutions. The recordings took place in 4 different care homes and 10 patients, resulting in a total of over 10 TB of data gathered over 12 months. The data was collected by the patient care staff during their everyday service. The nursing staff kept a log of the timing of emergency situations, based on which events were labeled with their exact start and end times by members of the audio communication group. Based on these annotations together with the raw audio data, the training dataset was created including a sufficient amount of ambient sounds from the patients rooms such as device sounds, television or radio noise and sounds from the patients, resulting in a diverse database of soundscapes.

Using these recordings, a balanced dataset of 2000 1 s long samples was created by cutting and storing respective chunks denoting the time of occurrence, the type of emergency detection confidence (*positive, medium* or *negative*) and the type of background noise (*noise, speech* or *music*). The chunks do not have any overlap and they were selected with a clear assignment as emergency or non-emergency sounds to ensure maximum separability of the ground truth data classes. The length of 1 s was chosen as a trade-off between retaining enough acoustic information in the chunk and ensuring fast processing. No further pre-processing was applied. From these audio files the Mel spectrograms for the input of the CNN but also the labels for classification system can be created. Since the care provider explicitly desired an unambiguous response of the prediction (emergency or no emergency), but also to increase classification performance, we decided to employ a binary classification scheme (*emergency* or *non-emergency*).

In the first iteration, due to the smaller number of positive (emergency) samples, we decided to consider all the medium cases as positive, which can also be expected to increase the system's sensitivity.

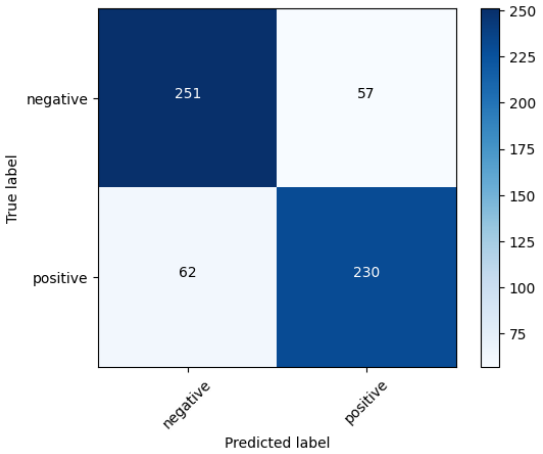

Fig. 4. Classification results for the best model, Accuracy 80.2%, F-Score 79.4%, training with 1400, validation with 600 samples.

E. Hardware

The hardware consists of two sub-components:

- A *Seeed Studio ReSpeaker Core v2* development module [17].
- A microcontroller-based pager unit (TTGO Lora32 Oled) with redundant wireless connectivity.

In contrast to a conventional microphone, a beamforming microphone array can change its directional characteristic under software control (steering). In contrast to a linear array, with a circular array, due to its geometry, a distinction can also be made between the directions of sound incidence from behind and from the front. The ReSpeaker Core v2 [17] is an uniform circular MEMS microphone array with integrated microcomputer. More precisely, a System-on-a-Chip [SoC] rock chip RK3229 (Quad Core A7) running a modified [18] light, command-line-only Linux (Debian 10). The device provides wireless connectivity via Wifi and Bluetooth, speaker output and further interfaces e. g. serial or I2C.

As hardware of the prototype pager system a Lilygo TTGO Lora ESP32 OLED [19] module was chosen, providing LoRa (Long Range) as an energy-saving long-distance radio connection with low power consumption (863 MHz, AES 128 bit), WiFi (2.4 GHz, WPA 2) as fall-back, and an integrated display. The transceiver unit on the microphone array sends its detection and device states as addressed encrypted messages to all pager units and vice versa. The WiFi network is configured as a mesh for a good coverage and is only used by the pager if the Lora radio connection is interrupted. Due to the large range of 150–300 m indoors [20] [21], however, this should rarely occur. Since the microphone arrays and their transceiver modules are powered with mains voltage, they keep both networks in parallel, while the battery-operated pagers preferably use the Lora connection. A complete connectivity failure in both networks is indicated by the pager both visually and acoustically, as well as an alarm due to positive inference events.

IV. Results and Discussion

We conducted a 3-fold cross-validation using all samples produced an total average accuracy of 80.2% on the validation set, with a recall of 78.7% and a precision of 80.1%, which leads to an F-Score of 79.4%. The confusion matrix is shown in Fig. 4. The evaluation showed that a system with relatively little training data can achieve an accuracy which is comparable to the results of similar studies using audio to detect medically relevant situations [6]. The care provider communicated that they would consider a prediction with the achieved accuracy as a valuable support for their response to an emergency. Especially the relatively high recall means that very few true positive cases are missed. However, the precision being just over 75% means that roughly one in every four predicted samples has a high probability of being a false positive. By employing statistical aggregation methods over several samples in a time period (e.g. 10 s) we will be able to adjust the prediction threshold and expect that this will improve the overall accuracy of the system.

Field tests of our system will be performed in order to see how we can raise its performance in real life operation. By employing continual learning and specializing the currently still generic model on the specific sounds of the respective patients and their environment, we expect to be able to increase the accuracy of the predictions. At the moment, the system provides lower accuracy than the results presented in [7], but close to those presented in [6]. A possible reason for this is that the above systems concentrated only on COVID19-related rather than more heterogeneous sounds using a smaller dataset than most systems in [22]. To improve the inferencing time of a single audio we will involve model quantization, by restructuring the model and reducing the number precision from 32-bit floats to 8-bit integers. This not only could lead to a reduction in the model size by up to 75% [23], but - more importantly in our case - to a faster execution of the inferencing, especially when it is outsourced to an external Edge TPU co-processor [24]. The study showed that an integrated system for emergency detection solely based on audio can achieve sufficient performance to be used to successfully assist intensive care personnel to shorten the time to start a medical procedure. A study evaluating the impact of the system on everyday work in nursing facilities has already begun. This will also help to tune the system parameters and the detection thresholds so as to increase the usability and acceptance of the device by the medical staff. Furthermore, it is planned to include other bio-parameters of the patients (e.g., oxygen saturation levels) in order to increase the reliability of the emergency prediction.

Acknowledgment

This project has been funded by the German Federal Ministry for Economic Affairs and Energy and managed by the knowledge network Ambucare (https://ambu-care.de/). Special thanks go to the care provider RENAFAN and especially to Benjamin Schubert for his help with the patient institutions, and to Yuchen Wang for the processing of the audio data.

REFERENCES

[1] I.-Y. Jeong, S. Lee, Y. Han, and K. Lee, "Audio event detection using multiple-input convolutional neural network," *Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2017.

[2] V. Morfi and D. Stowell, "Deep learning for audio event detection and tagging on low-resource datasets," *Applied Sciences*, vol. 8, no. 8, p. 1397, 2018.

[3] W. Xu, X. Zhang, L. Yao, W. Xue, and B. Wei, "A multi-view cnn-based acoustic classification system for automatic animal species identification," *Ad Hoc Networks*, p. 102115, 2020.

[4] X. Zhang, K. Wang, Y. Wang, Y. Shen, and H. Hu, "An improved method of rail health monitoring based on cnn and multiple acoustic emission events," in *2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE, 2017, pp. 1–6.

[5] G. Dernbach, A. Lykartsis, L. Sievers, and S. Weinzierl, "Acoustic identification of flat spots on wheels using different machine learning techniques," in *Fortschritte der Akustik–DAGA'20*, 2020.

[6] G. Pinkas, Y. Karny, A. Malachi, G. Barkai, G. Bachar, and V. Aharonson, "Sars-cov-2 detection from voice," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 1, pp. 268–274, 2020.

[7] A. Hassan, I. Shahin, and M. B. Alsabek, "Covid-19 detection system using recurrent neural networks," in *2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI)*. IEEE, 2020, pp. 1–5.

[8] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring automatic diagnosis of covid-19 from crowdsourced respiratory sound data," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 3474–3484.

[9] A. N. Belkacem, S. Ouhbi, A. Lakas, E. Benkhelifa, and C. Chen, "End-to-end ai-based point-of-care diagnosis system for classifying respiratory illnesses and early detection of covid-19: A theoretical framework," *Frontiers in Medicine*, vol. 8, 2021.

[10] S. Yadav, M. Keerthana, D. Gope, P. K. Ghosh *et al.*, "Analysis of acoustic features for speech sound based classification of asthmatic and healthy subjects," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6789–6793.

[11] I. D. Miranda, A. H. Diacon, and T. R. Niesler, "A comparative study of features for acoustic cough detection using deep architectures," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 2601–2605.

[12] V. S. Nallanthighal and H. Strik, "Deep sensing of breathing signal during conversational speech," in *Proceedings of the 20th Annual Conference of the International Speech Communication Association, Interspeech*, 2019, pp. 4110–4114.

[13] S. U. Amin, M. S. Hossain, G. Muhammad, M. Alhussein, and M. A. Rahman, "Cognitive smart healthcare for pathology detection and monitoring," *IEEE Access*, vol. 7, pp. 10 745–10 753, 2019.

[14] M. Alhussein, G. Muhammad, M. S. Hossain, and S. U. Amin, "Cognitive iot-cloud integration for smart healthcare: case study for epileptic seizure detection and monitoring," *Mobile Networks and Applications*, vol. 23, no. 6, pp. 1624–1635, 2018.

[15] J. Ko, J. H. Lim, Y. Chen, R. Musvaloiu-E, A. Terzis, G. M. Masson, T. Gao, W. Destler, L. Selavo, and R. P. Dutton, "Medisn: Medical emergency detection in sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 10, no. 1, pp. 1–29, 2010.

[16] P.-B. et al. pocketsphinx. [Online]. Available: https://github.com/cmusphinx/pocketsphinx

[17] S. Studio. Respeaker core v2.0. [Online]. Available: https://wiki.seeedstudio.com/ReSpeaker_Core_v2.0/

[18] P. Z. Baozhu Zuo. rk-linux-develop. [Online]. Available: https://github.com/respeaker/rk-linux-develop

[19] L. Shenzhen Xin Yuan Electronic Technology Co. Lilygo® ttgo lora. [Online]. Available: https://is.gd/4f8IOx

[20] J. Haxhibeqiri, A. Karaagac, F. Van den Abeele, W. Joseph, I. Moerman, and J. Hoebeke, "Lora indoor coverage and performance in an industrial environment: Case study," in *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017, pp. 1–8.

[21] J. Petäjäjärvi, K. Mikhaylov, M. Hämäläinen, and J. Iinatti, "Evaluation of lora lpwan technology for remote health and wellbeing monitoring," in *2016 10th International Symposium on Medical Information and Communication Technology (ISMICT)*, 2016, pp. 1–5.

[22] G. Deshpande and B. Schuller, "An overview on audio, signal, speech, & language processing for covid-19," *arXiv preprint arXiv:2005.08579*, 2020.

[23] P. Warden and D. Situnayake, *Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers*. O'Reilly Media, 2019.

[24] G. LLC. Coral usb accelerator. [Online]. Available: https://coral.ai/products/accelerator