ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414881

COUGHWATCH: REAL-WORLD COUGH DETECTION USING SMARTWATCHES

Daniyal Liaqat, Salaar Liaqat, Jun Lin Chen, Tina Sedaghat, Moshe Gabel, Frank Rudzicz, Eyal de Lara

Department of Computer Science, University of Toronto, Toronto ON, Canada

ABSTRACT

Continuous monitoring of cough may provide insights into the health of individuals as well as the effectiveness of treatments. Smartwatches, in particular, are highly promising for such monitoring: they are inexpensive, unobtrusive, programmable, and have a variety of sensors. However, current mobile cough detection systems are not designed for smartwatches, and perform poorly when applied to real-world smartwatch data since they are often evaluated on data collected in the lab.

In this work we propose CoughWatch, a lightweight cough detector for smartwatches that uses audio and movement data for in-the-wild cough detection. On our in-the-wild data, CoughWatch achieves a precision of 82% and recall of 55%, compared to 6% precision and 19% recall achieved by the current state-of-the-art approach. Furthermore, by incorporating gyroscope and accelerometer data, CoughWatch improves precision by up to 15.5 percentage points compared to an audio-only model.

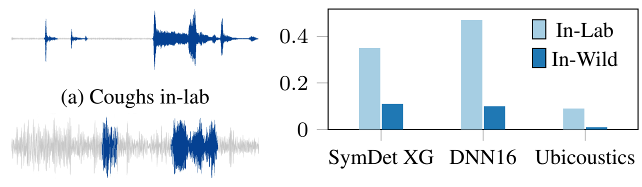
Index Terms— audio classification, cough detection

1. INTRODUCTION

Coughing is a common reflex that can sometimes indicate illness or worsening health. In individuals with lung disease, for example, an increase in coughing frequency may be associated with the onset of an episode, or general worsening, of their disease [1]. Therefore, continuously monitoring coughs could be highly valuable for monitoring the health of people prone to, developing, or suffering from lung disease [2, 3]. Similarly, cough monitoring could have a role in COVID-19 detection and monitoring since coughing is a common symptom [4]. For healthy individuals, cough monitoring could provide a baseline for health and indicate changes from this baseline.

Unfortunately, current mobile cough detection systems used by the medical community rely on either manual cough counting [5] or specialized, standalone hardware [6, 7] making them burdensome to use. Recently, the European Respiratory Society has stated that there is an urgent need for continuous cough detection systems [8].

The popularity of commodity mobile devices has given rise to a potential alternative: these lightweight, ubiquitous, inexpensive, and unobtrusive devices can be used as sensing platforms for cough detection. However, while there has been some work on using mobile phones and smartwatches for cough detection [9, 10, 11], that work is usually developed and evaluated using in-lab data, due to the lack of public, high-quality, labeled real-world datasets. In-the-wild data collected from smartwatches is quite different from lab data: audio is noisy, its properties change with the environment, and microphone position is affected by arm movements [12]. Furthermore, collecting in-the-wild data is challenging and labelling particularly so [13]. The difference between coughs recorded in an in-lab setting and those recorded in an in-the-wild setting is illustrated in Figure 1(a) and Figure 1(b). This results in cough detectors developed on in-lab data not transferring well to data collected in the wild,



(a) Coughs in-lab

(b) Coughs in-wild

(c) Detector F1 score.

Fig. 1. While in-lab audio is clear (a), in-the-wild audio is often noisy (b), negatively impacting performance of existing cough detectors (c), even when designed for in-the-wild audio [15] or retrained on our own in-the-wild dataset [14, 10].

even when retrained on in-the-wild data. For example, Figure 1(c) shows the difference in F_1 score for three existing works [14, 10, 15] when trained and evaluated on in-lab data compared to in-the-wild data. Despite having 9× more data in the wild dataset, SymDet and DNN16 models have a 3.2 and 4.7 times lower F_1 score, with Ubicoustics scoring even lower.

We propose CoughWatch: a cough detector designed to run on smartwatches and operate on real world data, enabling continuous and unobtrusive cough monitoring. We use smartwatches to collect and label 97 hours of continuous, in-the-wild audio and motion data from 16 participants, and show that CoughWatch achieves 5.7 to 6.7 times higher F_1 score on in-the-wild data than existing cough detectors. We also demonstrate the feasibility of running CoughWatch continuously on a smartwatch for a full day.

2. DATA COLLECTION AND PREPARATION

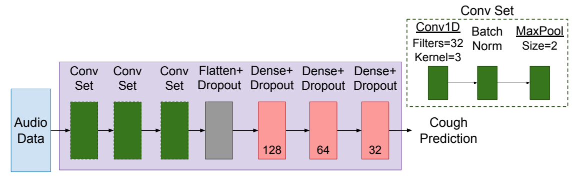
To develop and evaluate our model, we collected two datasets: a large in-the-wild dataset for developing models, and a smaller in-lab dataset for comparisons to existing work. All studies were approved by the University Health Network Research Ethics Board (REB 15-9068 and 18-5462).

In-the-wild Dataset: We recruited 16 participants who had a chronic lung disease (4 female, 12 male, mean age 69.3), for a 3-month in-the-wild study. Each participant was given a smartwatch and a smartphone equipped with our data collection applications. As part of informed consent, participants were made aware of the data being collected by the smartwatch application and shown how to use various privacy features built in to the app. Participants went about their normal lives, wearing the smartwatch during the day and charging it at night. While charging, data was relayed by the smartphone to our server.

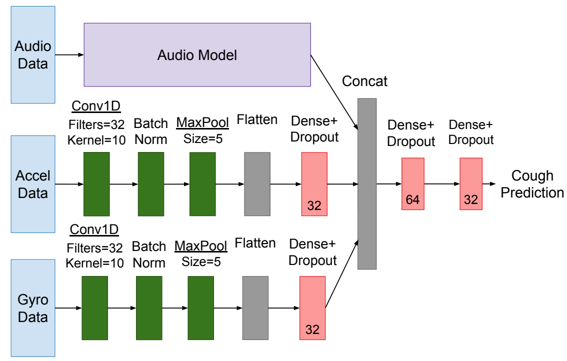
In-lab Dataset: For the in-lab portion, we recruited an additional 13 participants with the same chronic lung disease (exact selection criteria listed on our `ClinicalTrials.gov` entry NCT03857061). To match existing works [10, 11], we record audio data from the smartphone rather than a smartwatch. Participants either held the smartphone or placed it on the table in front of them. The participant is guided through several exercises, tests and tasks, such as speaking, walking, lung function tests, and voluntary coughing.

Audio Preprocessing and Labelling: To label our data, we first remove silence by applying A-Weighting filter [16], squaring the result, applying a low pass filter, and thresholding on 0.5 second windows. These non-silent segments are then labelled by paid annotators. Each segment is labeled as having a cough or not having a cough. We labelled 10% of non-silent segments twice (different annotators) to allow us to compute inter-rater agreement.

Resulting Datasets: We collected 4225 hours of sensor and audio data from the in-the-wild study. Of the 4225 hours of audio, 1726 (41%) hours were non-silent. To date we have labelled 97 hours (2.3% of total), identifying 912 positive (cough) and $65,062$ negative examples (1:72 class balance) with an inter-rater agreement of 95%. The in-lab study resulted in a total of 12 hours of in-lab audio of which 5.8 hours (48%) were non-silent. We labelled all 5.8 hours, resulting in 288 positive and $6,779$ negative examples (1:23 class balance). The proportion of coughs in the in-lab dataset is much higher than in-the-wild dataset because the in-lab study contained a voluntary coughing session, a common practice in existing a work.



(a) CoughWatch Audio Only (AO) model



(b) CoughWatch Sensor Fusion (SF) model

Fig. 2. Cough detection models.

3. COUGHWATCH

We define the following classification task: given a 10 second audio segment (and, potentially, corresponding accelerometer and gyroscope measurement), determine whether the segment contains a cough. We build two cough detection models for this task: CoughWatch Audio Only (AO) relies solely on audio data, while the CoughWatch Sensor Fusion (SF) also uses data from IMU sensors (gyroscope and accelerometer).

Input Data: As input, we use smartwatch audio, accelerometer and gyroscope data. Audio is sampled as 16-bit monochannel PCM at 16 kHz, and is divided to 10-second clips (zero-padded as needed). Audio clips are pre-processed using a 24-length gammatone filterbank applied to 20ms frames and converted into a spectrogram. These features were described and extensively evaluated for cough detection by Saba [17]. Accelerometer and gyroscope data is sampled at 20 Hz, and is fed into the models without any preprocessing, similar to how previous work has used IMU data for machine learning [18].

Data Augmentation: We augment the audio data in two ways. First, we drop every second audio sample and linearly interpolate the dropped samples. We apply this interpolation-based augmentation method twice, once dropping even samples and once dropping odd samples. The second augmentation method adds white Gaussian noise, scaled such that the amplitude of the noise is 1% the amplitude of the original audio. Combining these two augmentation methods quadruples the size of our training data. We found that varying parameters of an augmentation or applying it repeatedly did not improve performance. IMU data is not augmented.

Audio Only Model: For CoughWatch AO, we use a convolutional neural network (CNN) as shown in Figure 2(a). The audio spectrogram is passed through three *convolutional sets* with the final set connecting to a flatten layer. Each convolutional set consists of a convolutional layer, a batch normalization layer, and a max pooling layer. The flattened output of the convolution is passed to a dense network with three layers of 128, 64, and 32 neurons each. To reduce overfitting, we add dropout layers with rate of 0.2 between every dense layer, as well as the start and end of the dense network. The final dropout layer connects to the output layer where we have two nodes corresponding to our prediction. All layers use rectified linear activations except the output layer, which uses softmax. The parameters for the model were chosen using a systematic hyperparameter optimization procedure on a small subset of the data using standard grid search.

Sensor Fusion Model: CoughWatch SF consists of three subnetworks that feed into a single dense network as shown in Figure 2(b). The audio sub-network is identical to the trained CoughWatch AO model, and its weights are frozen during training of the SF model. Freezing the weights allows the AO subnetwork to be used independently. This opens the possibility of dynamically switching between the AO and SF models during execution to save battery life by turning off IMU sensors, or to avoid accuracy degradation when a distribution shift is detected in IMU data (e.g., due to physical activity or travel). We feed the accelerometer and gyroscope data into two identical networks, consisting of a convolutional set, a flatten layer, and a dense layer with dropout. The output of all three networks is then concatenated and passed through two more dense layers with dropout, before the final output layer. The structure of the output layer, the activation function, and the optimization are the same as CoughWatch AO.

4. EVALUATION

Our main method of comparison is through precision-recall curves. Previous works [14, 15], which relied on data collection in controlled settings has often relied on ROC curves or overall accuracy, both of which are poor indicators of performance in completely wild settings where data classes are heavily imbalanced. To allow comparision with previous works, we do also present ROC curves.

We train models using adaptive moment estimation [19] (Adam) with $lr = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and decay of 0. We reduce the learning rate by a factor 10 if validation loss does not decrease for 3 epochs. We also use early stopping: if validation loss has not decreased for 15 epochs. Our models generally train for fewer than 30 epochs. We use stratified Monte Carlo cross validation

Model	Parameters	FLOPS	AUC	F_1
DNN16 [14]	0.04M	0.5M	0.759	0.095
Ubicoustics [15]	72M	1082M	0.504	0.010
Coughwatch AO	0.5M	8M	0.835	0.638
Coughwatch SF	0.6M	9M	0.855	0.550

Table 1. Performance of CNN-based models on in-the-wild data.

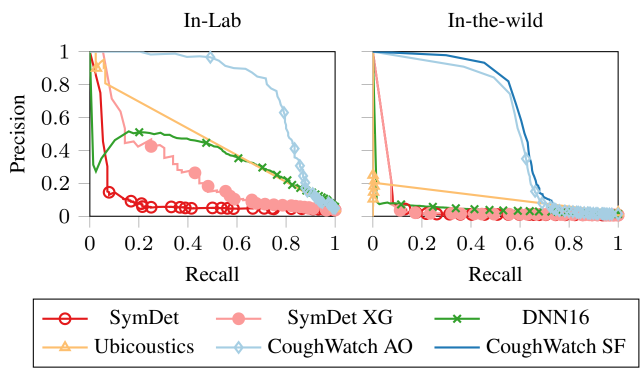


Fig. 3. P-R curve for in-lab and in-the-wild data.

with 5 rounds. In each round, we use a 64:16:20 train, validate, test split, with each set preserving the proportion of coughs. We use the procedure proposed by Forman and Scholz [20] to aggregate multiple rounds of training into a single precision-recall curve.

4.1. Cough Detection Accuracy

We evaluate the performance of CoughWatch AO and CoughWatch SF on our labeled datasets, and compare it to three state-of-the-art cough detectors. The first is SymDetector [10], which computes features on an audio segment and uses a support vector machine (SVM) to detect coughs. We found that replacing the SVM with a gradient-boosted tree [21] resulted in better performance, and we therefore also compare against a version of SymDetector with XGBoost (SymDet XG). The second model, denoted DNN16 [14], uses short-time Fourier transforms and a CNN-based architecture. We implement these models as closely as possible based on details from their respective works, as the implementations were not publicly available. We validated our implementation by comparing the ROC curve and ROC AUC score of the CNN model proposed in DNN16 [14]. The ROC curves for all models are shown in Figure 5. The third model we compare to is Ubicoustics [15]. It was trained using public datasets, including AudioSet [22], and claims to work off-the-shelf for in-the-wild cough detection. Thus, we used the pre-trained model for evaluation. Table 1 shows the size, complexity and performance of these models. CoughWatch is an order of magnitude smaller and faster than the SotA [15], with superior performance.

Detection Performance: Figure 3 shows the precision-recall curve for both the in-lab and in-the-wild datasets. For each curve, we find the point that maximizes F_1 score and find that CoughWatch substantially outperforms existing cough detection systems on both our in-the-wild data and in-lab data. On the in-lab dataset, CoughWatch AO achieves a maximum F_1 score of 0.773 with a precision of 0.838 and recall of 0.717, higher than DNN16 ($F_1 = 0.466$), Ubicoustics ($F_1 = 0.09$), SymDetector ($F_1 = 0.124$) and SymDetector with XGBoost ($F_1 = 0.352$). On the in-the-wild dataset, CoughWatch SF achieves a maximum F_1 score of 0.660 (precision of 0.820 and recall of 0.552), 5.7 times higher than SymDetector with a maximum F_1 score of 0.111, 6.7 times higher than DNN16 with a maximum F_1 score of 0.095 and substantially higher than Ubicoustics which had a maximum F_1 score of 0.01. CoughWatch AO is also superior to prior work, achieving a maximum F_1 score of 0.638 (0.743 precision and 0.559 recall). Ubicoustics is a general sound activity detector, rather than a cough detector, and is designed to work out-of-the-box for in-the-wild data. We hypothesize that this, along with their slightly more controlled data collection process, is why it achieved a high precision, but low recall and F_1 score on our dataset.

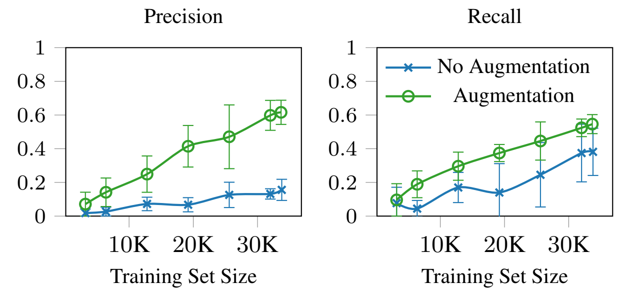


Fig. 4. Learning curves with and without augmentation.

Effect of IMU Data: As shown in Figure 3 the CoughWatch SF model, which combines audio data with accelerometer and gyroscope data, outperforms the audio-only model. To better quantify this difference, we compare the precision of CoughWatch SF and CoughWatch AO models at the same recall. Between 40% and 70% recall, we observe a 4 to 15.5 percentage point increase in precision when using the IMU data. While the difference in the precision-recall curve may seem small, because coughing is a rare event, the increase in precision becomes substantial. For instance, at 50% recall, the audio only model would have 11.7 false positives in a day whereas CoughWatch would have 5.6.

Effect of Data Augmentation: Figure 4 shows how precision and recall grow when we increase the size of the training set, with and without data augmentation. Data augmentations boosts precision and recall substantially: it yields 3.5 to 6.2 times higher precision than when using un-augmented data. Similarly, augmented data results in between 1.3 to 4.3 times higher recall. Additionally, we observe that precision and recall scores have not plateaued with the amount of annotated data, implying that additional training data is likely to yield further improvements.

4.2. Running on a Smartwatch

CoughWatch is designed to run on smartwatches, which are battery- and CPU-constrained. To evaluate how well-suited CoughWatch is to continuous monitoring, we implement it on a smartwatch, and measure runtime and the effect on battery life. In our testing, we use three Android Wear smartwatches representing three generations of wearable processors: LG Urbane (Snapdragon 400), Huawei Watch 2 (Snapdragon 2100) and Misfit Vapor X (Snapdragon 3100).

We modify the application used for in-the-wild data collection to run continuously instead of with a duty cycling scheme. The cough detector first pre-processes the audio data, converting it into a gammatone spectrogram using a cross-compiled gammatone library [23] and JTransforms [24]. The spectrogram and IMU data, is fed into

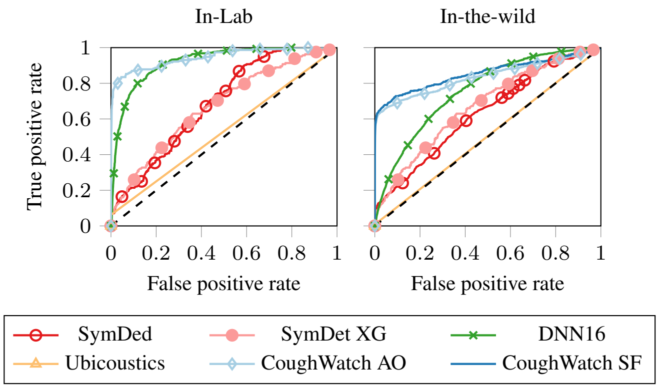
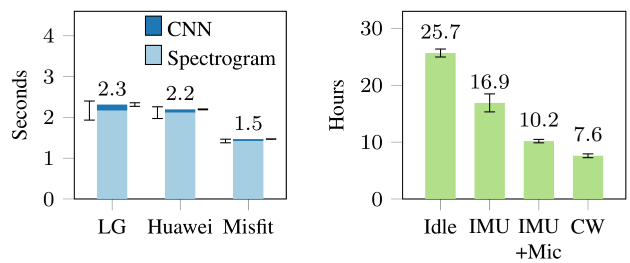


Fig. 5. ROC curve for in-lab and in-the-wild data



(a) Runtime.

(b) Battery life.

Fig. 6. Runtime of running our system on three smartwatches (a), and resulting battery life on LG Urbane under different operating modes (b). Error bars show standard deviation.

the CoughWatch SF CNN built with TensorFlowLite 2.1.0.

First, we evaluate whether the pre-processing and CNN can run in real time on a smartwatch. We run our application and log the start and end timestamp of the pre-processing calculation and the CNN inference on 10 second data. We collected at least 2000 runs for each watch. The mean and standard deviation of the runtime for all three of our watches is shown in Figure 6(a). We observe that all watches are able to run our cough detection system in real time, taking 1.5 to 2.3 seconds to process 10 seconds worth of data. Interestingly, the CNN is much faster than computing the spectrogram – accounting for only 3–10% of the overall runtime.

Next, we evaluate the effect on battery life of running CoughWatch SF on a smartwatch. To do so, we configure our application to run under four different modes. We conducted each mode six times where we charged the watch to over 95% and then ran the application continuously until the battery fully discharged. In the first mode, *idle*, our application periodically records only battery levels. In the *IMU* mode, our application records data from the accelerometer and gyroscope. In the third mode, *IMU+Mic*, we record IMU and audio data. Finally, the *CW* mode both collects and preprocesses the data and then runs CoughWatch SF. Figure 6(b) shows the mean and standard deviation of the battery life for these modes when running continuously on the LG Urbane. We expect the other two watches to have longer battery lives as they have larger batteries and newer processors. In the idle condition, we measured the LG Urbane to have a 25.7 hour battery life. Recording data from the IMU drops this down to 16.9 hours and recording from the IMU and microphone reduces battery life further to 10.2 hours. Running CoughWatch reduced battery life by a further 2.4 hours, showing that running the model drains the battery 24% faster than simply recording the data.

Given that in our data collection studies we rely on duty cycling (record data for a short time then sleep for a set time) to even simply record data, we wanted to estimate the battery life of running cough detection in a duty cycling scheme. To do so, we follow the simulation approach proposed in our prior work [18, 25]. The simulator starts at minute zero with a full battery, and estimates what the battery level should be every minute by sampling from a normal distribution described by the mean and standard deviation of the idle condition, or the mean and standard deviation of the CW condition, based on the duty cycle state. Based on 100 runs, we estimate that running CoughWatch on a smartwatch with a 2 min record/8 min sleep duty cycling scheme would provide 17.4 hours of battery life (SD: 0.48 hours), which is enough to last a full day. Using silence detection to reduce the number of times preprocessing and the CNN have to run would likely result in additional battery life.

In summary, CoughWatch can run on a smartwatch in real-time. Using a duty cycling scheme, which is already required for simply recording data, CoughWatch can be run while still providing a full day of battery life.

5. RELATED WORK

Beyond the three cough detection systems we previously discussed [10, 14, 15], many other cough detection systems have been developed. Some systems, including Ubicoustics, have been developed using publicly available audio clips of coughs [26, 22]. However, these models may not generalize to real-world applications; for example, the AudioSet dataset [22] contains dog coughs, and may not be representative of real-world coughs. Other systems have designed and conducted studies in order to collect datasets. For instance, MobiCough [27] used a collar based microphone to collect their data. Similarly, [11] and [28] used a neck-worn device to record audio. While these ideally placed devices are better suited to pick up cough sounds, they are less practical for long term use. In our data collection study design, we used a smartwatch for cough detection as smartwatches are unobtrusive, readily available and have been shown to be a feasible method for monitoring [2].

In-the-wild data collection studies introduce many privacy concerns. While IMU data has been used for in-the-wild clinical monitoring [18], audio, despite being a rich source of information, is underused. One reason for this is that audio data contains sensitive information that impacts the privacy of the participants [29]. We have shown that the trained model can run directly on the smartwatch, eliminating the privacy concerns of uploading audio recordings to a remote server. Alternatively, [11] and [9] have proposed solutions for privacy-preserving cough detection when running the model on a remote server is essential.

6. CONCLUSION

Our goal was to build a practical cough detection system. To achieve this goal, we paid close attention to how the model would be deployed. This led us to collect in-the-wild data from a smartwatch with participants living their life as usual. We also designed the system to be lightweight enough to run on the smartwatch. The resulting system has a 5.7 to 6.7 times higher F_1 score than prior work. Closing the loop, we show that running this model on a smartwatch is feasible in terms of battery life and compute requirements.

7. REFERENCES

[1] J. Smith and A. Woodcock, "Cough and its importance in COPD," *Int J Chron Obstruct Pulmon Dis*, vol. 1, no. 3, pp. 305–314, Sept. 2006.

[2] R. Wu, D. Liaqat, E. de Lara, T. Son, F. Rudzicz, H. Alshaer, P. Abed-Esfahani, and A.S. Gershon, "Feasibility of using a smartwatch to intensively monitor patients with chronic obstructive pulmonary disease: Prospective cohort study," *JMIR mHealth and uHealth*, vol. 6, no. 6, pp. e10046, 2018.

[3] D. Liaqat, E. Nemati, M. Rahman, and J. Kuang, "A method for preserving privacy during audio recordings by filtering speech," in *2017 IEEE Life Sciences Conference (LSC)*, Dec 2017, pp. 79–82.

[4] Eduardo Martin-Sanz, Juan Riestra, Laura Yebra, Alba Larran, Fiorella Mancino, Joaquin Yanes-Diaz, Maria Garrote, Marta Colmenero, Esther Montiel, Cristina Molina, Daniel Moreno, Antonio Rodriguez, Gerardo Monedero, Ricardo Sanz-Fernández, Rocio Gonzalez, and Jonathan Esteban-Sanchez, "Prospective study in 355 patients with suspected covid-19 infection: Value of cough, subjective hyposmia, and hypogeusia," *The Laryngoscope*, vol. 130, no. 11, pp. 2674–2679, 2020.

[5] K. McGuinness, K. Holt, R. Dockry, and J. Smith, "P159 validation of the VitaloJAK™ 24 hour ambulatory cough monitor," *Thorax*, vol. 67, no. Suppl 2, pp. A131–A131, Dec. 2012.

[6] S. S. Birring, T. Fleming, S. Matos, A. A. Raj, D. H. Evans, and I. D. Pavord, "The Leicester cough monitor: Preliminary validation of an automated cough detection system in chronic cough," *Eur. Respir. J.*, vol. 31, no. 5, pp. 1013–1018, May 2008.

[7] E. Vizel, M. Yigla, Y. Goryachev, E. Dekel, V. Felis, H. Levi, I. Kroin, S. Godfrey, and N. Gavriely, "Validation of an ambulatory cough detection and counting application using voluntary cough under different conditions," *Cough*, vol. 6, no. 1, pp. 3, May 2010.

[8] A. H. Morice, E. Millqvist, K. Bieksiene, S. S. Birring, P. Dicpinigaitis, C. D. Ribas, M. H. Boon, A. Kantar, K. Lai, L. McGarvey, D. Rigau, I. Satia, J. Smith, W.-J. Song, T. Tonia, J. W. K. van den Berg, M. J. G. van Manen, and A. Zacharasiewicz, "ERS guidelines on the diagnosis and treatment of chronic cough in adults and children," *Eur. Respir. J.*, Jan. 2019.

[9] D. Liaqat, E. Nemati, M. Rahman, and J. Kuang, "A method for preserving privacy during audio recordings by filtering speech," in *2017 IEEE Life Sciences Conference (LSC)*, Dec. 2017, pp. 79–82.

[10] X. Sun, Z. Lu, W. Hu, and G. Cao, "SymDetector: Detecting sound-related respiratory symptoms using smartphones," in *Proc. UbiComp '15*, 2015.

[11] E. C. Larson, T. Lee, S. Liu, M. Rosenfeld, and S. N. Patel, "Accurate and privacy preserving cough sensing using a low-cost microphone," in *Proc. UbiComp '11*, 2011.

[12] Daniyal Liaqat, Robert Wu, Andrea Gershon, Hisham Alshaer, Frank Rudzicz, and Eyal de Lara, "Challenges with real-world smartwatch based audio monitoring," in *Proceedings of the 4th ACM Workshop on Wearable Systems and Applications*, 2018, pp. 54–59.

[13] D. Liaqat, R. Wu, S. Liaqat, E. de Lara, A.S. Gershon, and F. Rudzicz, "The Ground Truth Trade-Off in Wearable Sensing Studies," *arXiv:2001.09738 [cs]*, Jan. 2020.

[14] J. Amoh and K. Odame, "Deep neural networks for identifying cough sounds," *IEEE Trans. Biomed. Circuits Syst.*, vol. 10, no. 5, pp. 1003–1011, Oct. 2016.

[15] "Ubicoustics: Plug-and-play acoustic activity recognition," *In The 31st Annual ACM Symposium on User Interface Software and Technology*, pp. 213–224, 2018.

[16] H. Fletcher and W. A. Munson, "Loudness, its definition, measurement and calculation," *J. Acoust. Soc. Am.*, vol. 5, no. 2, pp. 82–108, Oct. 1933.

[17] E. Saba, *Techniques for Cough Sound Analysis*, Ph.D. thesis, University of Washington, 2018.

[18] D. Liaqat, M. Abdalla, P. Abed-Esfahani, M. Gabel, T. Son, R. Wu, A.S. Gershon, F. Rudzicz, and E. de Lara, "Wearbreathing: Real world respiratory rate monitoring using smartwatches," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 3, no. 2, June 2019.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980 [cs]*, Dec. 2014.

[20] G. Forman and M. Scholz, "Apples-to-apples in cross-validation studies: Pitfalls in classifier performance measurement," *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 1, pp. 49, Nov. 2010.

[21] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. KDD '16*, 2016.

[22] 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 *Proc. ICASSP 2017*, 2017.

[23] M. Bernard, "libgammatone," `https://github.com/mmmaat/libgammatone`, 2016.

[24] P. Wendykier, "Jtransform," `http://github.com/wendykierp/JTransforms`, 2015.

[25] D. Liaqat, S. Jingoi, E. de Lara, A. Goel, W. To, K. Lee, I. De Moraes Garcia, and M. Saldana, "Sidewinder: An energy efficient and developer friendly heterogeneous architecture for continuous mobile sensing," in *Proc. ASPLOS '16*, 2016.

[26] L. Kvapilova, V. Boza, P. Dubec, M. Majernik, J. Bogar, J. Jamison, J. C Goldsack, D. J Kimmel, and D. R Karlin, "Continuous sound collection using smartphones and machine learning to measure cough," *Digit Biomark*, vol. 3, no. 3, pp. 166–175, 2019.

[27] C. Pham, "MobiCough: Real-time cough detection and monitoring using low-cost mobile devices," in *IIDS*, 2016, pp. 300–309.

[28] P. Kadambi, A. Mohanty, H. Ren, J. Smith, K. McGuinnes, K. Holt, A. Furtwaengler, R. Slepetys, Z. Yang, J. Seo, J. Chae, Y. Cao, and V. Berisha, "Towards a wearable cough detector based on neural networks," in *Proc. ICASSP 2018*, 2018.

[29] V. G. Motti and K. Caine, "Users' privacy concerns about wearables," in *Financial Cryptography and Data Security*, 2015, pp. 231–244.