TripletCough: Cougher Identification and Verification From Contact-Free Smartphone-Based Audio Recordings Using Metric Learning

Stefan Jokić, David Cleres, Frank Rassouli, Claudia Steurer-Stey, Milo A. Puhan, Martin Brutsche, Elgar Fleisch, and Filipe Barata

***Abstract*—Cough, a symptom associated with many prevalent respiratory diseases, can serve as a potential biomarker for diagnosis and disease progression. Consequently, the development of cough monitoring systems and, in particular, automatic cough detection algorithms have been studied since the early 2000s. Recently, there has been an increased focus on the efficiency of such algorithms, as implementation on consumer-centric devices such as smartphones would provide a scalable and affordable solution for monitoring cough with contact-free sensors. Current algorithms, however, are incapable of discerning between coughs of different individuals and, thus, cannot function reliably in situations where potentially multiple individuals have to be monitored in shared environments. Therefore, we propose a weakly supervised metric learning approach for cougher recognition based on smartphone audio recordings of coughs. Our approach involves a triplet network architecture, which employs convolutional neural networks (CNNs). The CNNs of the triplet network learn an embedding function, which maps Mel spectrograms of cough recordings to an embedding space where they are more easily distinguishable. Using audio recordings of nocturnal coughs from asthmatic patients captured with a smartphone, our approach achieved a mean accuracy of 88% (± 10% SD) on two-way identification tests with 12 enrollment samples and accuracy of 80% and an equal error rate (EER) of 20% on verification tests. Furthermore, our approach outperformed human raters with regard to verification tests on average by 8% in accuracy, 4% in false acceptance rate (FAR), and 12% in false rejection rate (FRR). Our code and models are publicly available.**

***Index Terms*—Cough monitoring, metric learning, mobile sensing, remote patient monitoring, speaker identification, speaker verification, triplet network.**

I. Introduction

COUGHING is associated with many prevalent respiratory diseases, ranging from minor ailments like the common cold to more serious chronic illnesses such as chronic bronchitis, chronic obstructive pulmonary disease (COPD), asthma, tuberculosis, gastroesophageal reflux, and cystic fibrosis [1]. It is among the most common complaints for which individuals seek medical advice [2]. According to surveys in the United Kingdom and Japan, the prevalence of chronic cough in the general population is estimated to be 10.2% and 12% respectively [3], [4]. Further, the use of over-the-counter cough remedies has a significant financial impact, which is estimated at $156 million and $6.8 billion sales from 2012 to 2013 in the United Kingdom and the United States, respectively [5].

Cough counts over a prolonged period (e.g., weeks, days, or hours) are clinically meaningful and can serve as potential biomarkers for diagnosis, disease progression, and the analysis of the effect of treatment in patients with respiratory conditions [6]. Cough counts by a dedicated listener are, however, found unreliable if self-reported [7] or too laborious and time-consuming [8]. Consequently, researchers have been devising methods for assessing cough counts since the 1950s [9]. In particular, cough monitoring systems, which count the number of coughs from audio recordings by employing an automatic cough detection algorithm, have been proposed. Such algorithms enable distinguishing coughs from other sounds such as speech or background noise, thereby serving as a preliminary step in a cough monitoring system to ensure that coughs can be counted reliably. Thus, they allow for an objective measure of cough frequency and are generally preferred over traditional methods involving patient self-reports. Prior works on cough

Manuscript received July 16, 2021; revised January 18, 2022; accepted February 13, 2022. *(Stefan Jokić and Filipe Barata contributed equally to this work.) (Corresponding author: Filipe Barata.)*

Stefan Jokić is with the Department of Computer Science, ETH Zurich, 8092 Zurich, Switzerland (e-mail: jokics@student.ethz.ch).

David Cleres and Filipe Barata are with the Centre for Digital Health Interventions, ETH Zurich, 8092 Zurich, Switzerland (e-mail: dcleres@ethz.ch; fbarata@ethz.ch).

Frank Rassouli and Martin Brutsche are with the Lung Center, Cantonal Hospital St. Gallen, 9007 St. Gallen, Switzerland (e-mail: frank.rassouli@kssg.ch; martin.brutsche@kssg.ch).

Claudia Steurer-Stey is with the Epidemiology, Biostatistics and Prevention Institute, University of Zurich, 8006 Zurich, Switzerland, and also with the mediX Group Practice, 8037 Zurich, Switzerland (e-mail: claudia.steurer@medix.ch).

Milo A. Puhan is with the Epidemiology, Biostatistics and Prevention Institute, University of Zurich, 8006 Zurich, Switzerland (e-mail: miloalan.puhan@uzh.ch).

Elgar Fleisch is with the Centre for Digital Health Interventions, ETH Zurich and University of St. Gallen, 8092 Zurich, Switzerland (e-mail: efleisch@ethz.ch).

Digital Object Identifier 10.1109/JBHI.2022.3152944

detection have achieved sensitivity and specificity values of over 90% [10]. More recent approaches have also employed mobile technologies such as smartphones and wearables for cough detection [11], [12]. This is motivated by the widespread use of mobile technologies, which have the potential to meet the health monitoring needs of the ever-growing number of chronic respiratory patients [13].

Although the large amount of research conducted in this field shows that proof of concept for cough detection is well established [12], [14], one problem, in particular, remains largely unresolved for contact-free cough detection. That is, to determine to whom a given cough belongs. This can be problematic in situations where a monitored patient is sharing a room with other individuals who cough. In such a scenario, the presence of coughs from others, so-called ambient coughs, will cause an exceedingly high and thus, incorrect number of coughs to be attributed to that patient. Therefore, algorithms capable of assigning coughs to the individuals that produced them are necessary. From a technical perspective, this brings the additional challenge of personalization, i.e., in addition to cough detection, the monitoring system must have the ability to recognize the patient.

In literature, speaker recognition refers to the task of identifying individuals from the characteristics of their voices. More precisely, one distinguishes between speaker verification and identification, which we define as follows.

- *Verification* refers to the process of comparing a speaker's utterance to those of a reference individual to determine whether the speaker is the same person as the reference.
- *Identification* refers to the process of comparing a speaker's utterance to those of multiple reference individuals to determine which one of the references the speaker corresponds to or is the most resembling.

Furthermore, the reference utterance samples are also known as enrollment samples.

On this basis, our objective is to investigate to what extent a machine-learning-based approach can enable patient identification and verification from contact-free smartphone-recorded cough sounds. Whereas first evidence demonstrates the feasibility of patient verification from cough sounds [15], our proposed approach differs from previous research. It presents a novel triplet-learning-based approach for patient verification and identification from voluntary and reflex cough sounds. The novelty of our approach is manifold. First, to the best of the authors' knowledge, this is the first work to investigate the feasibility of patient verification from *contact-free* cough recordings. Second, we propose a novel convolutional neural network architecture that we employ within a triplet network and evaluate our results on two data sets containing smartphone-recorded voluntary and asthmatic reflex coughs. Third, we propose a simple and fast heuristic for mining triplets, which are used as training data for the triplet network. Fourth, we conduct a series of analyses such as the impact of the recording device used to capture cough sounds, the distance between the microphone and the cougher, and the number of coughers and enrollment samples on the performance of our approach. We also compare the performance of our approach with that of 19 human raters on cougher verification tests. The applicability of our approach in offline devices, which are limited in resources such as storage and computational power, is further facilitated by not having to retrain on new enrollment samples. Our research benefits patients with respiratory diseases associated with coughing by enabling contact-free cough monitoring of multiple patients in shared environments applicable to scalable and cost-efficient technologies such as smartphones. In addition, we make our code and models publicly available (cf. Table I), so that other researchers can use and apply them. Finally, in this paper, we focus on identifying and verifying patients from cough sounds and assume that cough detection has been reliably achieved. We acknowledge that complex practical challenges were left out by bypassing the detection step and refer the reader to our previous works on this topic [12], [16].

TABLE I
CODE AND MODEL ACCESS

License	GNU GPLv3
Implementation	Python 3.8+
Libraries	tensor-flow 2.5.0, keras 2.4.3
Code repository	`https://github.com/ADAMMA-CDHI-ETH-Zurich/TripletCough`

The rest of the paper is organized as follows. Section II summarises related work in the domain of speaker recognition using utterances and coughs. Section III describes the methodology applied for pre-processing cough audio data, the design of our neural network architecture for cougher recognition, and the training and evaluation of our network. Section IV is devoted to the experimental results, which are subsequently discussed in Section V. Finally, Section VI outlines the work presented in this paper.

II. RELATED WORK

Conventional approaches presented in prior work on speaker recognition involve first extracting features from utterances, typically by using Mel-frequency cepstral coefficients (MFCC) or linear predictive coding (LPC)-based features, and then modeling the distribution of the feature vectors using a Gaussian mixture model (GMM) [17]. The parameters of the GMM are then concatenated to form a *supervector*, which can subsequently be used for classification. Dehak *et al.* [18] have instead proposed the use of *i-vectors*, which are obtained by employing factor analysis (FA) as a dimensionality reduction technique for the GMM *supervectors*. These *i-vectors* enable to combine speaker and channel factors into a single, low-dimensional space. For a long time, approaches involving the classification of *i-vectors* using support vector machines (SVM), cosine similarity measure-based scoring [18] or probabilistic linear discriminant analysis (PLDA) [19] have been the state-of-the-art in speaker recognition algorithms. However, these approaches come with the disadvantage of relying on handcrafted feature engineering.

On the other hand, deep learning, a specific class of machine learning algorithms based on artificial neural networks (ANN)

with representation learning, ultimately replaces the need for handcrafted feature engineering by shifting the task of feature extraction towards model optimization. Additionally, deep learning architectures have achieved promising results in the domain of speech recognition [20], where they are usually employed as encoders that produce embeddings out of utterances, which are then compared. For instance, Nagrani *et al.* [21] trained a Siamese network with a contrastive loss consisting of a CNN based on a ResNet-inspired architecture. Then, they used cosine similarity to classify the embeddings. Similarly, Li *et al.* [22] used a neural network architecture based on ResNet but also experimented with an architecture consisting of gated recurrent units (GRU). They, too, employed cosine similarity to compare embeddings but trained their networks using the triplet loss. Snyder *et al.* [23] exploited a time-delay neural network architecture with a statistical pooling layer to produce embeddings called *x-vectors*, which they finally compared using PLDA scores.

While there is a considerable amount of research conducted on speaker recognition, "cougher" recognition, in contrast, remains mostly unexplored. Zhang *et al.* [24] have designed a neural network architecture involving convolutional and time-delayed layers for extracting speaker features. Although their proposed approach was not specifically tailored for cougher recognition, it still performed well on the cougher verification task, achieving an equal error rate of 10.99% when using PLDA scores. They, however, evaluated their model solely on voluntary coughs as opposed to natural reflex coughs. Voluntary coughs produced by one person usually sound very similar, while reflex coughs produced by the same person are subject to greater variation. Whitehill *et al.* [15] introduced a novel approach for cougher recognition using natural, in-the-wild reflex cough data. Their method is based on multitask learning and, in particular, consists of training a neural network with a ResNet architecture on both cough and speech data. Training for two distinct yet related tasks helps their model generalize better. Their model yielded an accuracy of 82% on 4-way identification tests using ten enrollment samples per cougher and a 23% EER on verification tests. Their data collection, however, cannot be generalized to the case studied in this work. The coughs were recorded by having participants carry a smartphone in their shirt pocket or strapped around their neck [15]. Hence, not only is the smartphone always located at a small and approximately the same distance to the participant, but it is also directly in contact with the person's chest. In consequence, the microphone membrane of the smartphone may also capture the chest vibrations waves caused by coughing, which may notably reduce the difficulty of distinguishing ambient cough from patient cough compared to using contact-free recordings.

In this paper, we aim to design a neural network for cougher recognition using voluntary cough data as well as natural reflex cough data from patients with asthma. These reflex coughs were recorded with a smartphone placed anywhere in the patient's room, at different distances from the patient, and without contact. More information on the cough data sets and how they were collected, is presented in the following section.

III. Methodology

A. Data Sets

This work used two different cough data sets. The first data set is a labeled set of audio recordings composed of voluntary cough sounds collected in a lab study [12]. The corresponding study protocol was approved by the ethics commission of ETH Zurich in April 2016 (EK 2016-N-15). A studio microphone and four different devices with built-in microphones simultaneously recorded each participant which was instructed to voluntarily cough at two distinct recording distances: close and distant. In the close setup, the microphones were placed on a table directly in front of the cougher, at a distance of 0.15 m. In the distant setup, the microphones were initially placed at the same location as for the close recordings but shifted towards one side of the table by 1 m. Due to missing data from two devices (HTC M8 and Apple iPhone 4) as a result of human error during data collection (e.g., errors in the use of the recording technology), we only retained the following devices: the Samsung Galaxy S6 smartphone, Google Nexus 7 tablet, and the Røde NT1000 studio microphone. Furthermore, we left a participant out of the data set if the number of recorded coughs for that participant was not equal across the recording devices. This resulted in a data set of 637 and 629 close and distant cough recordings, respectively, from 38 participants (28 female, 10 male). We then partitioned it into a roughly 50/25/25 split for training, validation, and testing, respectively. Hence, we used recordings of 18 participants for training, 10 for validation, and 10 for testing.

The second data set is a labeled set of audio recordings consisting of nocturnal reflex coughs from asthmatic patients collected in a multicenter, longitudinal, observational study [25]. The corresponding study protocol was approved by the ethics commission responsible for research involving humans in eastern Switzerland in November, 2017 (BASEC ID: 2017–01872). Data collection involved equipping patients with a study smartphone, the Samsung Galaxy A3 (2017), and instructing them to keep it in their bedroom at night, with no specific restrictions on where to place it. The smartphone's built-in microphone recorded the nocturnal coughs for 28 nights. The original data set included 79 adult asthmatic patients, but we discarded recordings of patients who shared the bedroom with another person. Furthermore, after removing participants whose number of recordings was too scarce, the resulting data set contained a total of 9221 cough recordings from 46 subjects (29 female, 17 male) approximately divided into a 60/20/20 split for training, validation, and testing, respectively. Hence, the training, validation, and testing sets contained 26, 10, and 10 patients, respectively.

Both data sets are summarized in Table II. The voluntary cough data set serves to demonstrate the feasibility of cougher recognition since the voluntary coughs were recorded in the same controlled acoustic environment, thereby reducing the variability across cough samples. In addition, the voluntary cough data allows us to investigate the influence of the quality of the microphone of the recording device and the distance between cougher and microphone on the performance of our approach.

TABLE II
SUMMARY OF THE VOLUNTARY COUGH (VC) AND THE REFLEX COUGH (RC) DATA SETS

	Partition	# of subjects	# of M/F subjects	# of smokers	AVG ± SD of age	# of coughs	Total length of CS [s]	AVG ± SD of length of CS [s]
VC	Training	18	6 / 12	2	27.22 ± 4.75	614	299.51	0.49 ± 0.14
	Validation	10	2 / 8	0	24.30 ± 4.55	334	192.73	0.58 ± 0.21
	Test	10	2 / 8	2	24.90 ± 5.32	318	168.30	0.53 ± 0.21
	Overall	38	10 / 28	4	25.84 ± 4.91	1266	660.54	0.52 ± 0.18
RC	Training	26	9 / 17	10	43.23 ± 20.67	5784	1620.22	0.28 ± 0.13
	Validation	10	4 / 5	3	40.10 ± 16.86	1616	386.94	0.24 ± 0.09
	Test	10	4 / 6	3	43.70 ± 12.78	1821	565.31	0.31 ± 0.13
	Overall	46	17 / 29	16	42.65 ± 18.13	9221	2580.46	0.28 ± 0.13

"#" stands for number, "M" for male, "F" for female, "CS" for cough signals, "AVG" for average and "SD" for standard deviation.

The reflex cough data set, on the other hand, resembles a more realistic scenario for performing cougher recognition.

B. Data Pre-Processing

The data pre-processing pipeline mostly follows the approach described in previous works on cough detection at our lab [12], [16]. The purpose of the pipeline is to frame the cough signals to have the same length, to extract their most essential information, i.e., the characteristic explosive sounds, and to transform the raw signals into an adequate representation before passing them as input to our network. The pipeline can be summarized as follows: first, we apply an anti-aliasing filter to the cough signal and downsample it to 22.05 kHz. In doing so, we reduce the size of the data and training time without loss of essential information as the characteristic sounds of a cough are associated with frequencies below 10 kHz [26]. The anti-aliasing filter is a pre-computed low-pass filter provided by librosa [27], a python package for music and audio analysis. The filter is designed using a Kaiser window with $\beta = 8.56$ and a roll-off frequency of $0.85 * f_{nyquist}$, where $f_{nyquist} = 11.025\ kHz$ is the Nyquist frequency, i.e. half of the target sampling rate. Next, we extract a window of 1.2 s around the maximum amplitude of the cough recording. We deliberately chose the window size to be longer than the average duration of a cough to avoid excluding characteristic features of the cough signal that might be useful for training our network. Subsequently, we apply min-max normalization to the extracted signal. Finally, we compute a mel-scaled spectrogram of the signal with 80 bands, a hop-length of 112 samples, and a 2048 point Fast Fourier Transform (FFT). Mel-scaled spectrograms, i.e., two-dimensional visual representations of the frequency spectrum of a signal as it varies over time, have been shown to outperform other time-frequency representations in conjunction with CNNs [28].

C. Triplet Network Architecture

As described in Section II, speaker verification and identification systems based on deep learning architectures usually train an encoder that takes an utterance as input and outputs an embedding, i.e., a low-dimensional continuous vector representation. Based on a computed similarity score or distance between embedded utterances, the model determines whether

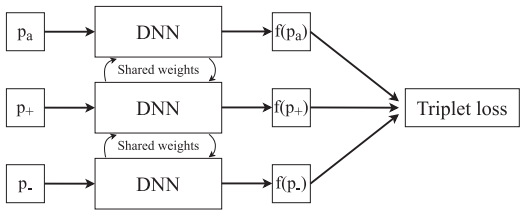
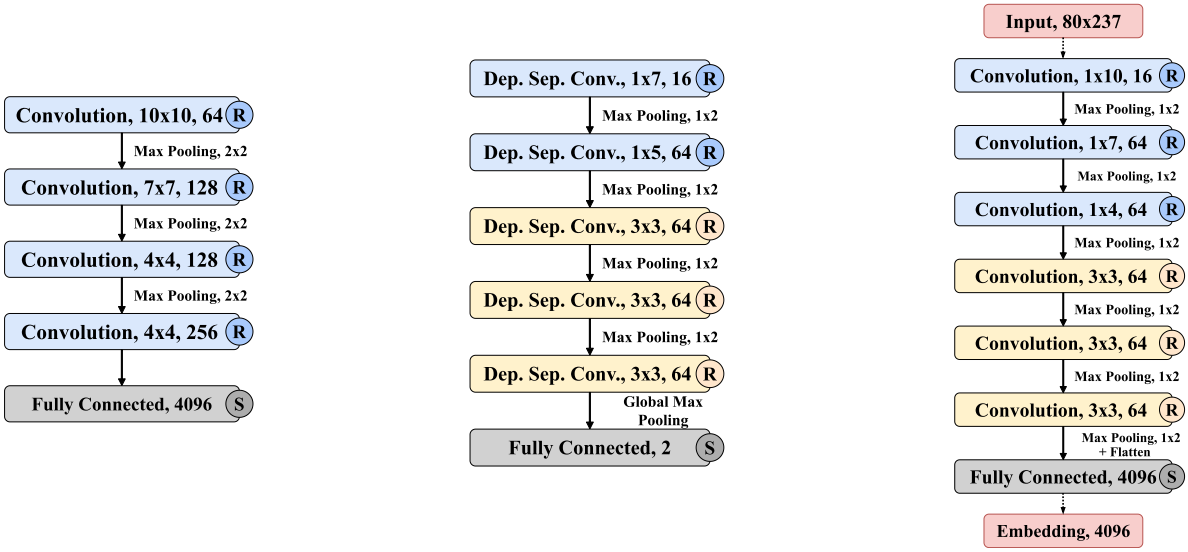


Fig. 1. The triplet network architecture.

different utterance samples originate from the same person or two different individuals. Thus, with respect to a distance metric, the encoder's goal is to learn an embedding function that maps samples (utterances) of the same class (speaker) closer to each other while mapping samples belonging to different classes further away from each other. This goal can be reformulated to be equivalent to a *metric learning* problem, where the objective lies in adapting a pairwise real-valued distance metric d to obtain a new distance metric $\tilde{d}$ such that with respect to $\tilde{d}$, distances between samples of the same class are minimized and distances between samples of different classes are maximized. More precisely, the objective is to learn a new distance metric $\tilde{d}(x, y) = d(f(x), f(y))$ given an original distance metric $d(x, y)$, where f is an embedding function.

Triplet networks have shown promising results when applied to metric learning tasks, especially in image similarity ranking [30], [31]. They have also shown to outperform the Siamese network, a popular alternative approach, with regard to image similarity learning tasks [32]. As illustrated in Fig. 1, a triplet network is composed of three deep neural networks (DNN) with identical architecture and shared weights. It follows a weakly supervised learning paradigm as the training data takes on the form of unlabeled triplets of samples (p_a, p_+, p_-), where p_a is the *anchor* sample, p_+ is the *positive* sample that should be similar to the anchor, and p_- is the *negative* sample that should be dissimilar to the anchor. Therefore, these triplets define the relative similarity relationship between three given samples. Each of the samples in a triplet is separately passed as input to each of the three DNNs of the triplet network, which in the following compute the embeddings $f(.)$ of the respective samples. During training, the triplet network seeks to minimize the hinge loss for a triplet, or *triplet loss* function, which is



(a) CNN architecture by Koch et al. [29] (b) CNN architecture by Barata et al. [12] (c) Proposed CNN architecture

Fig. 2. CNN architectures. "R" stands for the rectifying linear unit, "S" for the sigmoid activation function and "Dep. Sep. Conv." for depthwise separable convolution. (a) CNN architecture by Koch *et al.* [29]. (b) CNN architecture by Barata *et al.* [12] (c) Proposed CNN architecture.

defined as follows,

$$\ell(p_a, p_+, p_-) = max\left\{0,\ d\Big(f(p_a), f(p_+)\Big) - d\Big(f(p_a), f(p_-)\Big) + g\right\}, \quad (1)$$

where d is the chosen distance metric, f is the embedding function that the DNNs learn and g is a chosen gap parameter that regularizes the gap between the two pairs, (p_a, p_+) and (p_a, p_-), respectively [30]. The triplet loss incurs a larger penalty for not respecting the similarity relationship between pairs of similar samples (p_a, p_+), i.e., by assigning larger distances between them, and analogously for not respecting the dissimilarity relationship between pairs of dissimilar samples (p_a, p_-). In this work, we leverage the triplet network architecture in conjunction with CNNs to design a novel approach for cougher recognition.

D. CNN Architecture

In the literature, CNNs are commonly employed as the components of a triplet network [30]–[32]. They are a class of DNNs whose use has come to bolster the state-of-the-art in the domain of computer vision, in particular for image classification, segmentation, and object detection [33]. CNNs have also proven effective for audio event detection [34], especially when spectrograms are used as representations for audio data. A typical CNN comprises several convolutional and pooling layers in an alternating fashion, followed by some form of nonlinearity applied before or after pooling and finally a small number of fully connected layers.

Inspired by the architecture proposed by Koch *et al.* [29] (cf. Fig. 2(a)), which was employed in a Siamese network, we designed the CNN architecture for our triplet-network-based cougher recognition system in the following. Siamese networks follow a similar architecture to triplet networks; the main difference is that they consist of only two identical neural networks that share the same weights and usually employ a pairwise ranking loss. They have achieved state-of-the-art performance on one-shot classification tasks using handwritten character data [29]. In our application, however, spectrograms serve as input rather than natural images. Spectrograms differ from natural images in that they contain a temporal dimension and can, therefore, be regarded as sequential data [35]. For this reason, we designed a new CNN architecture by first adjusting the architecture proposed by Koch *et al.* [29] to employ one-dimensional kernels in the convolutional layers for capturing the temporal information of spectrograms. Next, we removed the last convolutional layer, which reduced the number of parameters. Then, similar to the CNN architecture employed in the cough detection model developed by Barata *et al.* [12] (cf. Fig. 2(b)), we exploited a characteristic design choice of the VGG architecture [36], i.e., we added several successive convolutional layers comprising 3x3 kernels and an equal number of channels per layer as the final convolutional layers of our architecture. We optimized the number of these VGG-inspired convolutional layers and determined three layers to be the best choice for our network. We also optimized the number of channels per layer, which we subsequently set to 64. Details regarding the optimization of the architecture can be found in Section III-F. Furthermore, we inserted a max-pooling layer after each convolutional layer. We flattened the output of the last max-pooling layer into a vector which is passed on to a fully connected layer composed of 4096 units to produce an embedding. Additionally, we applied an L2 regularizer to the weights of the last layer to prevent overfitting. We employed the rectifying linear unit (ReLU) function and the sigmoid function as the activation functions for all convolutional layers and the final fully connected layer, respectively. Our proposed CNN architecture is depicted in Fig. 2(c).

E. Triplet Mining

In accordance with the triplet network architecture, the training data must take on the form of triplets (p_a, p_+, p_-) as described in Section III-C. Each triplet fits into one of the following three categories,

- *Hard* triplet: $\tilde{d}(p_a, p_-) < \tilde{d}(p_a, p_+)$,
- *Semi-hard* triplet: $\tilde{d}(p_a, p_+) < \tilde{d}(p_a, p_-) < \tilde{d}(p_a, p_+) + g$,
- *Easy* triplet: $\tilde{d}(p_a, p_-) > \tilde{d}(p_a, p_+) + g$,

where $\tilde{d}$ and g refer to the learned distance metric and the gap parameter, respectively. For all but easy triplets, the loss in (1) is nonzero.

Triplet mining refers to the process of selecting triplets for training and plays a crucial role as it has a great impact on the performance of a triplet network. Training only on hard triplets makes training unstable by producing noisy gradients and converging to local optima, resulting in a collapsed model, i.e., $f(x) = 0$ [31]. While Schroff *et al.* [31] suggest mining only semi-hard triplets, Wu *et al.* [37] show that mining only semi-hard triplets makes little progress as the number of available semi-hard triplets for mining decreases. Instead, they propose an approach based on distance-weighted sampling, which mines a mixture of easy, semi-hard, and hard triplets, resulting in a better performance [37], [38]. Inspired by their approach, we employed a simple and fast online heuristic for mining a mixture of triplets. Online mining refers to selecting triplets within a randomly sampled batch, called mini-batch [31].

The proposed heuristic can be described as follows: Let n be the training batch size and the number of triplets that we want to select. We first create a mini-batch of size $2n$ by randomly and uniformly selecting triplets from the training data. Next, using the current weights of the triplet network at the given training iteration, we evaluate the loss for each triplet in the mini-batch. Subsequently, we sort the triplets of the mini-batch in descending order with respect to their associated loss and select the first $\frac{n}{2}$ triplets from the sorted mini-batch. Finally, we select the remaining $\frac{n}{2}$ triplets by sampling uniformly at random from the remaining $2n - \frac{n}{2}$ triplets in the mini-batch. Thus, one half of the resulting batch contains the hardest triplets from the mini-batch, and the other half contains triplets associated with a broader range of losses. We execute the proposed heuristic within each training iteration during batch construction.

F. Training & Optimization

We chose the squared Euclidean distance function $d(x, y) = ||x - y||_2^2$ as the distance metric for our triplet network. We trained the network using the stochastic optimization algorithm Adam [39] and the back-propagation scheme for computing gradients. Furthermore, we initialized the weights of the network using the Glorot initialization [40] scheme. We found the best network hyperparameters and made our CNN architecture design choices by using a held-out validation set. More precisely, for each combination of hyperparameters (including architecture design choices), we trained our network for 5000 iterations on the training set of the voluntary cough data set from the Røde NT1000 microphone and validated every 100 iterations on a batch of 50 randomly sampled triplets from the Røde microphone's validation set. Subsequently, we saved the model weights at the training iteration yielding the lowest validation loss as defined in (1). We then evaluated the saved models on the validation set via two-way one-shot identification tests (cf. Section III-G1) and selected the optimal hyperparameters based on the highest achieved mean accuracy on the tests. In such a manner, we found the learning rate and batch size to be optimal for the values 10^{-3} and 64, respectively. We optimized the gap parameter g separately and set it to 1. Ultimately, we trained our network on a given data set with the optimized hyperparameters described above for 5000 iterations to build a final model.

We implemented our neural network using Keras, an open-source Python software library for deep learning that is built on top of TensorFlow [41]. We trained the network on the cluster infrastructure of ETH Zurich. The code and models are publicly available (cf. Table I).

TABLE III
EVALUATION SCENARIOS USED IN THE IDENTIFICATION AND VERIFICATION TESTS

ID	Evaluation Scenario	Used Data
I.1	Varying number of enrolled coughers N and number of enrollment samples per cougher K	Voluntary cough data recorded with the Røde NT1000
I.2	Varying distance between microphone and cougher	Voluntary cough data recorded in close and distant proximity with the Røde NT1000
I.3	Varying type of recording device (including analysis of the generalizability of the model to data from an unseen device)	Voluntary cough data recorded with the devices, Røde NT1000, Samsung Galaxy S6, and Google Nexus 7
I.4	In-the-wild nocturnal reflex coughs from asthmatic patients	Reflex cough data recorded with the Samsung Galaxy A3
V.1	In-the-wild nocturnal reflex coughs from asthmatic patients (including comparison to human performance)	Reflex cough data recorded with the Samsung Galaxy A3

I and **V** refer to identification and verification, respectively. Note that *all* the samples used for the tests in each evaluation scenario originate from the test set of the corresponding data set indicated in this table and the network trained on the corresponding training set of the same data set is used for inference on these test samples. Hence, a different model is used for each evaluation scenario. The models were built by training our network using the same optimized hyperparameters described in section III-F.

G. Evaluation

We investigated the performance of our approach in identification and verification tests in different evaluation scenarios on test sets that present yet unseen cough samples to the model. We designed the evaluation scenarios **I.1**–**I.3** (cf. Table III) to allow the study of the influence of a varying parameter (e.g., number of coughers, number of enrollment samples, recording distance, and recording device) on the performance of the network. The evaluation scenarios **I.4** and **V.1** (cf. Table III) represent a more realistic application scenario with data from asthmatic patients. Additionally, we generated the evaluation scenarios **I.1–I.4** for the identification tests by using data from the voluntary and reflex cough data sets. For the verification evaluation scenario **V.1**, we

only used the reflex cough data set. We omitted verification testing with the voluntary cough data because better results with a data set of recorded coughs in a laboratory setting can be expected and argue that the reflex cough data set allows us to emulate a more realistic scenario of performing cougher verification "in-the-wild". Table III gives an overview of the different evaluation scenarios.

1) Identification Tests: We emulated cougher identification tests via N-way K-shot classification tasks, where N refers to the number of different classes (coughers) and K refers to the number of samples (coughs) per class. In our case, an N-way K-shot classification task involves comparing a given sample x_{eval} with a set consisting of K enrollment samples of each of N different coughers and deciding, based on the comparisons, which cougher x_{eval} belongs to. Thus, this set of enrollment samples with which x_{eval} is compared, contains a total of $N * K$ different samples. In the context of N-way K-shot classification, this set of enrollment samples is referred to as the *support set*. Note that both x_{eval} and all the samples in the support set are randomly sampled from the test set of the data set indicated in Table III. We randomly sampled x_{eval} from the test set while constraining it to belong to one of the N coughers present in the support set and ensuring that it differs from the specific samples in the support set.

Using our model, we compared a given sample x_{eval} of a given cougher to K samples $x_1, \ldots, x_K$ of another cougher by first computing the distance between x_{eval} and each of the K samples using the learned distance metric $\tilde{d}(x, y) = ||f(x) - f(y)||_2^2$ and subsequently calculating the mean of all the computed distances. We repeated this procedure for each of the coughers in the support set. Finally, we selected the cougher associated with the lowest mean distance to x_{eval} as the predicted cougher to whom x_{eval} belongs. More precisely, we employed the following decision rule:

$$i^* = \underset{i \in \{1,\ldots,N\}}{\operatorname{argmin}} \frac{1}{K} \sum_{x \in \mathcal{S}_i} \tilde{d}(x_{eval}, x) \tag{2}$$

where $\mathcal{S}_i$ is the support set consisting of K cough samples of cougher $c_i \in \{c_1, \ldots, c_N\}$. Consequently, c_{i^*} is the predicted cougher.

a) Evaluation Scenario I.1: To evaluate our approach, we generated N-way K-shot tasks with different values for N and K and then evaluated them using the above decision rule. Finally, we reported the classification accuracy on these tasks.

To generate the classification tasks, we randomly sampled N coughers from the test set for each classification task. We then randomly selected the samples for the support set from the sampled coughers. We randomly chose one sample from any of the sampled coughers as x_{eval}. In this fashion, we generated a total of 10000 classification tasks.

b) Evaluation Scenario I.2: In this evaluation scenario, we investigated the impact that the distance between microphone and cougher and the quality of the recording device may have on the performance of our approach. To do so, we trained our network on data containing voluntary coughs (cf. Section III-A) recorded with the Røde NT1000 microphone in "close" and "distant" proximity, respectively, thereby producing two models, one for each recording distance. We also trained our network on data containing both voluntary coughs recorded in "close" and "distant" proximity. Subsequently, we evaluated each trained model separately on the data recorded in "close" and "distant" proximity, respectively. We carried out all evaluations via two-way one-shot classification tasks, i.e., identification tests with two enrolled coughers and only one sample per cougher.

c) Evaluation Scenario I.3: For this evaluation scenario we trained our network on voluntary cough data from each of three different recording devices (Samsung Galaxy S6, Google Nexus 7, Røde NT1000) separately and on data that combines the recordings of all devices. We then evaluated each trained model on data from each of the devices separately. Additionally, to investigate the generalizability of our approach to cough data from a yet unseen device, we trained our network on the data of all but one device and thereafter evaluated it on the test data of the held-out device. All evaluations were carried out using two-way one-shot tasks. We repeated this procedure for each of the devices.

d) Evaluation Scenario I.4: Finally, we investigated the performance of our approach when applied to reflex coughs. Consequently, we trained our network on the nocturnal reflex cough data set from asthmatic patients and evaluated the resulting model using two-way K-shot tasks with different choices for K.

Since the evaluation scenarios **I.2**–**I.4** only include tests with just two enrolled coughers, i.e., $N = 2$, we adopted a more rigorous evaluation approach for these evaluation scenarios than the one presented in evaluation scenario **I.1**, which would have been too computationally demanding for $N > 2$: first, we generated a set of all permutations of two coughers from the test set. Then, for each pair of coughers (c_i, c_j) from the set of permutations, we generated 125 two-way K-shot tasks by randomly selecting K samples from c_i and c_j, respectively, as the samples of the support set and one sample from c_i as the test sample. Given that the number of coughers in our test set for the voluntary and reflex cough data set was 10, this resulted in a total of ${}^{10}P_2 * 125 = 11250$ classification tasks for each type of data set. Finally, we computed the mean accuracy and standard deviation of all the classification accuracies associated with each pair of coughers.

2) Verification Tests: First, we randomly selected 10 samples from the reference cougher as enrollment samples from the test set. Next, we randomly selected 10 additional test samples, half of which were randomly sampled from the reference cougher and the other half from coughers different from the reference. Subsequently, we iterated over the newly selected samples (i.e., half reference cougher, half other coughers) and determined for each of them whether they belonged to the reference. In each iteration, we denoted the sample to be evaluated as x_{eval}. We computed the mean distance between the x_{eval} and each of the enrollment samples using the learned distance metric $\tilde{d}$. If the resulting mean distance for a test sample x_{eval} is below a specified threshold, our model predicts that x_{eval} belongs to the reference cougher. Otherwise, it predicts that x_{eval} belongs to a cougher different from the reference. Formally, our model

applies the following threshold-based decision rule:

$$\frac{1}{K}\sum_{x\in\mathcal{S}}\tilde{d}(x_{eval},x)=\begin{cases}<t\Rightarrow\text{same cougher}\\ \geq t\Rightarrow\text{different cougher}\end{cases} \quad (3)$$

where $\mathcal{S}$ is the set of K enrollment samples of the reference cougher to be verified. As noted above, we chose $K=10$. Additionally, we chose the threshold $t=2.2$ as this resulted in the highest achieved classification accuracy on verification tests performed on the validation set of our reflex cough data set.

a) Evaluation Scenario V.1: Using our reflex cough data set, we generated verification tests according to the procedure described above, by iterating over the 10 coughers in the test set and using a different cougher as the reference in each iteration. However, for two of the coughers in the test set only 9 rather than 10 samples were used for enrollment, due to insufficient samples for these coughers. Since 10 test samples were used in each of the 10 iterations, we generated a total of 100 verification tests to be evaluated by our model. Finally, we report the classification accuracy of our model on the verification tests, along with the false acceptance rate (FAR), the false rejection rate (FRR), and the equal error rate (EER), which were computed as follows,

$$ACC = \frac{TP+TN}{TP+TN+FP+FN}, \quad (4)$$

$$FAR = \frac{FP}{FP+TN}, \quad (5)$$

$$FRR = \frac{FN}{FN+TP}, \quad (6)$$

where TP, FP, TN, and FN denote the number of true and false positives and true and false negatives, respectively. The EER corresponds to the FAR and FRR when they are equal, which we determined by varying the threshold.

Furthermore, we established a human baseline with which we compared our model's performance. We instructed each of our 19 human raters (13 male, 6 female, age 22-31) to perform the same verification tests as our model. More precisely, for each of the ten coughers in the test set, the rater was first allowed to listen to the same enrollment samples presented to our model as often as desired. The rater then had to listen to the same 10 test samples that our model performed its predictions on and decide for each of them, whether it belongs to the reference cougher. Hence, every rater had to classify 100 test samples in total. Afterward, we reported the mean, median, standard deviation, best and worst FAR, FRR, and classification accuracy of all the raters.

IV. Results

In this section, we present the results of applying our evaluation methodology to our models. This includes the results of the identification tests in the various evaluation scenarios **I.1**–**I.4** and the results of the verification tests for the evaluation scenario **V.1**.

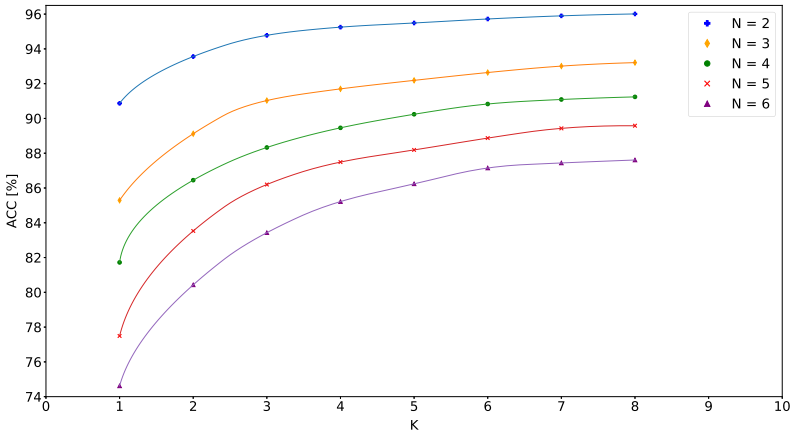
A. Identification Tests

*1) Evaluation Scenario **I.1**:* Fig. 3 shows the results of the N-way K-shot classification tasks for different choices and combinations of N and K. Consider first the case where $N=2$, i.e. identification tests with only two enrolled coughers and K samples per enrolled cougher. The results show that an accuracy of 90.87% can already be achieved with only a single sample per cougher, i.e., $K=1$, and that accuracies of more than 95% can be achieved if K is large enough, i.e., $K\geq 4$. Furthermore, it can be observed that the accuracy saturates at roughly $K=5$, after which there are diminishing returns. The most significant increase in mean accuracy between successive values of K is 2.7% which occurs between $K=1$ and $K=2$. In general, the results show that for a given choice of N, increasing the number of samples per enrolled cougher K increases accuracy, but the amount of improvement gradually declines as K is increased. Overall, accuracies of over 90% can be achieved for $N\leq 5$ when K is sufficiently large. For $N=6$, the highest accuracy achieved is over 87%, when $K\geq 6$. Additionally, for all the choices of N, diminishing returns can be observed starting at roughly $K=6$. As expected, the accuracy decreases when N is increased for a fixed K. In particular, it decreases by roughly 2% to 4% each time N is incremented by one, for a fixed K. Furthermore, the largest increase in accuracy between successive values of K for any choice of N occurs between $K=1$ and $K=2$, but the improvement is more substantial the larger N is.

*2) Evaluation Scenario **I.2**:* Table V shows the results of training on close and on distant voluntary cough data, as well as on the combination thereof, and subsequent evaluation of our model on the close and distant test data via two-way one-shot classification tasks. Overall, the results suggest that the recording distance has a negligible effect on the mean accuracy of the classification tasks, which ranges from as low as 89.64% to as high as 92.87%.

*3) Evaluation Scenario **I.3**:* Table IV shows the results of training on voluntary cough data from each of three different devices separately and a data set combining the data of all devices and then evaluating our model on the test data of each of the devices.

The results demonstrate that mean accuracies over 90% can be achieved not only with a studio microphone as shown before, but also with a smartphone. In fact, our approach achieves a mean accuracy of 94.15% when training and testing on cough data recorded with the Samsung S6. This is the best mean accuracy achieved among all combinations of data used for training and testing. Against expectation, our approach performs better on the Samsung S6 data than on the data from the Røde NT1000 studio microphone. Also note that for the model trained on the Samsung S6 data, the average of all mean accuracies obtained when testing on the data from each of the three devices separately is almost 90%.



		N				
		2	3	4	5	6
K	1	90.87	85.29	81.72	77.49	74.62
	2	93.56	89.12	86.45	83.53	80.43
	3	94.78	91.03	88.33	86.20	83.43
	4	95.25	91.70	89.46	87.49	85.22
	5	95.49	92.19	90.24	88.19	86.24
	6	95.72	92.64	90.83	88.87	87.15
	7	95.90	93.01	91.09	93.01	87.44
	8	96.01	93.21	91.24	93.21	87.61

Fig. 3. Evaluation scenario **I.1**. Plot and table of the accuracies [%] obtained on N-way K-shot classification tasks with different values for N and K. Quadratic interpolation is applied to obtain the curves. Each curve is associated with each choice of N.

TABLE IV
EVALUATION SCENARIO **I.3**

		Testing			
		Røde NT1000	Samsung S6	Google Nexus 7	Mean
Training	Røde NT1000	91.05 ± 9.55	89.69 ± 10.22	79.02 ± 12.66	86.59 ± 10.89
	Samsung S6	90.12 ± 10.07	94.15 ± 7.08	85.30 ± 12.97	89.86 ± 10.32
	Google Nexus 7	84.44 ± 11.89	87.75 ± 10.34	82.58 ± 12.22	84.92 ± 11.51
	All devices	90.09 ± 11.10	91.08 ± 8.27	81.00 ± 14.17	87.39 ± 11.44

Mean and standard deviation [%] of the accuracies obtained on two-way one-shot classification tasks when training and testing on all combinations of data from different devices. "All devices" is the data set combining the data from all devices.

TABLE V
EVALUATION SCENARIO **I.2**

		Testing		
		Close	Distant	Mean
Training	Close	91.05 ± 9.55	89.64 ± 11.87	90.35 ± 10.77
	Distant	89.67 ± 10.37	90.92 ± 10.32	90.30 ± 10.35
	Both	90.84 ± 8.76	92.87 ± 9.91	91.86 ± 9.35

Mean and standard deviation [%] of the accuracies obtained on two-way one-shot classification tasks when training and testing on all combinations of close/distant cough data. "Both" is the data combining close and distant cough data.

TABLE VI
EVALUATION SCENARIO **I.3**

		Testing on held-out device
Training	All but Røde NT1000	87.97 ± 10.69
	All but Samsung S6	90.54 ± 7.99
	All but Google Nexus 7	80.48 ± 12.00

Mean and standard deviation [%] of the accuracies obtained on two-way one-shot classification tasks when training on all but one device and testing on the held-out device.

Whereas the quality of the recording device employed to capture the cough samples used for training our model is certainly important, the quality of the device used to capture cough samples for testing is equally crucial. This is highlighted by the Google Nexus 7 tablet, the device with the poorest quality built-in microphone of the three [12]. When training on the data of any given device, the lowest mean accuracy is obtained when testing on data from the Google Nexus 7. Furthermore, when training on data from the Google Nexus 7, the mean accuracies are consistently below 88%, irrespective of what device's data is used for testing. The lowest mean accuracy of 79% is obtained when training on data from the Røde NT1000 and testing on data from the Google Nexus 7.

In contrast, when testing on data from the Røde microphone or the Samsung S6, mean accuracies of at least 84.44% are achieved, no matter which device's data were used for training. The influence of the quality of the recording device on the performance of our model is further illustrated by the observation that when training with data from all devices combined and testing on data from the Røde microphone or the Samsung S6, mean accuracies of about 90% are obtained for both devices. On the other hand, when training on the data of all devices combined and testing on the data of the Nexus 7 tablet, the mean accuracy drops by almost 10% in comparison. Furthermore, whereas the best mean accuracies for the Røde microphone and the Samsung S6 are obtained when training and testing on their own data, this does not hold for the Nexus 7 tablet. When training on the Nexus 7 tablet data, the best possible mean accuracy of 87.75% is achieved when testing on data from the Samsung S6.

Table VI shows the result of training on data from all but one device and subsequently evaluating our model on test data from the held-out device. The results suggest that our model generalizes well with respect to cough data from yet unseen

TABLE VII
EVALUATION SCENARIO **I.4**

N = 2			
K = 1	K = 2	K = 4	K = 6
73.08 ± 9.92	78.62 ± 10.55	83.16 ± 10.65	85.26 ± 10.46
K = 8	K = 10	K = 12	K = 13
86.75 ± 9.96	87.44 ± 9.79	88.29 ± 9.53	88.66 ± 9.48

Mean and standard deviation [%] of the accuracies obtained on two-way K-shot classification tasks using reflex cough data with different values for K.

TABLE VIII
EVALUATION SCENARIO **V.1**

	Mean ± SD	Median	Best	Worst
Accuracy	72 ± 6	73	83	56
FAR	28 ± 9	28	12	42
FRR	28 ± 10	26	16	54

Mean and standard deviation, median, best and worst accuracy, FAR and FRR [%] obtained on the verification tests performed by the human raters.

devices, with the lowest mean accuracy of 80.48% achieved on the data of the Nexus 7 tablet and the highest mean accuracy of 90.54% achieved on the data of the Samsung S6. Moreover, note that the obtained mean accuracies for each held-out device (87.97%, 90.54%, and 80.48% for the Røde NT1000, Samsung S6, and Google Nexus 7, respectively) do not differ substantially from those obtained when training and testing our model on the data of the respective devices, i.e., the mean accuracies on the diagonal of Table IV (91.05%, 94.15% and 82.58% for the Røde NT1000, Samsung S6 and Google Nexus 7, respectively).

*4) Evaluation Scenario **I.4**:* Table VII shows the results of two-way K-shot classification tasks with different choices for K when using nocturnal reflex coughs from asthmatic patients. Whereas overall, the achieved mean accuracies are, as expected, worse than those obtained with the voluntary cough data set, they can still reach over 88% with a sufficiently large K, i.e., $K \geq 12$. For $K \geq 4$, mean accuracies over 80% can be achieved. With only one cough sample available per enrolled cougher, i.e., $K = 1$, a mean accuracy of 73% is obtained. As in the previous scenario, the mean accuracy increases as K is increased, and the improvement is greatest when K is increased starting from a small value. Diminishing returns can be observed at $K = 12$.

B. Verification Tests

*1) Evaluation Scenario **V.1**:* The results of the verification tests performed by the human raters are listed in Table VIII. They show that human raters could determine whether a cough belonged to a particular individual with better than random accuracy on average. Even the worst-performing rater achieved an accuracy of 56%. Indeed, in general, humans are good at identifying idiosyncrasies of a person's voice, even by means of coughs [17]. The average FAR and FRR of 28%, however, highlight that this is certainly not a trivial task.

The results of our model on the verification tests are depicted in Table IX, along with the average results of the human raters, for comparison. They show that the average human rater is

TABLE IX
EVALUATION SCENARIO **V.1**

	Accuracy	FAR	FRR	EER
Our model	**80**	**24**	**16**	**20**
Human raters	72	28	28	N/A

Comparison between performance of our model and human raters on the verification tests. We report the accuracy, FAR and FRR [%]. For the human raters, the mean of each metric is depicted. For our model, the EER [%] is also given.

outperformed by our model with respect to every metric. Our model outperforms the human raters on average by 8% in accuracy, 4% in FAR, and 12% in FRR. The best-performing human rater, however, was able to beat our model's performance with regard to accuracy by 3% and with regard to the FAR by 12% (cf. Table VIII) but achieved a comparable FRR to the model. Thus, the human raters still constitute a very competitive baseline compared to our model.

V. DISCUSSION

A. Principal Findings

We have demonstrated the feasibility of cougher identification using our approach. Our approach can achieve mean accuracies of over 90% on identification tests involving two enrolled coughers and only one enrollment sample per cougher when applied to voluntary coughs. Remarkably, these high-performance values have not only been achieved using cough data from a studio microphone but also using cough data from a smartphone. By increasing the number of enrollment samples per cougher, the mean accuracy on the two-way identification tests can be improved further to values over 95% when using cough data from the Røde NT1000. However, the extent of improvement decreases as the number of enrollment samples is increased. For identification tests with more than two enrolled coughers, there is inevitably a decrease in achieved accuracy. Nevertheless, this can again be mitigated to some extent by increasing the number of enrollment samples. In doing so, the accuracy obtained on identification tests with five or fewer coughers can reach over 90%, as long as K is sufficiently large.

We have determined that our approach retains its performance regardless of the distance between the microphone and cougher used to capture coughs. The quality of the recording device, however, plays an important role, as shown by the lower mean accuracies obtained on identification tests using cough data from the Google Nexus 7 as opposed to the higher mean accuracies obtained using cough data from the Røde NT1000 or the Samsung S6. Additionally, our approach generalizes well to cough data from new devices, as testing on cough data from an unseen device results in only a slight decrease in mean accuracy compared to training and testing on the device that was previously held out.

When applying our approach to a data set of reflex coughs instead, a decline in the mean accuracy achieved on the two-way identification tests can be observed. Although our reflex cough data set contains a larger number of cough samples compared to our voluntary cough data set (cf. Section III-A), this result

can be explained by the fact that the reflex cough data set includes several sources of variability that are not present in the voluntary cough data set. This is because the reflex cough samples were recorded with a smartphone in acoustic environments that differed among study participants in terms of premises, distance to the smartphone, and background noise. With only one enrollment sample per cougher, the mean accuracy on the two-way identification tests is 73.08%, which means that it has dropped by almost 18% compared to using voluntary cough data (cf. Table VII and Fig. 3). Increasing the number of enrollment samples per cougher, however, not only improves the accuracy on identification tests but also reduces the loss of accuracy compared to using voluntary cough data. Indeed, with eight enrollment samples per cougher, the achieved mean accuracy on the two-way identification tests using reflex cough data is 86.75%, which corresponds to an increase of 13.67% compared to using only one sample per enrolled cougher and a decrease of 9.26% compared to using voluntary cough data. Mean accuracies over 88% can be obtained if $K \geq 12$.

In addition, we have demonstrated that our approach is capable of performing cougher verification using reflex cough data. It has achieved respectable results on the verification tests, with an accuracy of 80% and an EER of 20%, while outperforming the human raters on average.

B. Practical Implications

As noted in Section I, the main issue with conventional contact-free cough monitoring systems is their inability to distinguish between coughs of different individuals, potentially resulting in incorrect cough counts. Hence, a more reliable cough monitoring system capable of distinguishing coughs could be designed by incorporating our cougher recognition algorithm alongside an automated cough detection algorithm. First, the cough detection algorithm determines whether the input audio recording contains a cough. If it does not, no cough is counted. Otherwise, the cough's audio recording is passed as input to our cougher recognition algorithm to assign it to a monitored individual. The number of counted coughs is then incremented. For the automated cough detection algorithm, the efficient CNN-based cough detection model developed by Barata *et al.* [12] is especially suitable, as it achieves state-of-the-art results and is efficient enough to run on smartphones.

We attempted to keep the number of layers in our CNN architecture relatively small while trying to avoid significant decreases in performance. Our network comprises 19.54 million parameters, of which 99% are those of the final dense layer of our network. At inference, 498.34 million floating-point operations (MFLOPs) and 32.3 MB of memory are required to produce a cough embedding. Granted that these numbers are significantly higher than those of the cough detection algorithm developed at our lab [12], which only requires 1.74 MFLOPs and 1.232 MB of memory for its execution, we argue that the deployment of our pre-trained cougher recognition models on modern smartphones should still be feasible. Recent advancements in smartphone hardware have facilitated the deployment of deep learning models on smartphones, especially neural processing units (NPUs) designed to accelerate machine learning applications. For example, the Samsung Galaxy S21 smartphone has 8 GB of RAM and a graphics processing unit that boasts 1.53 trillion floating operations per second and a tri-core NPU, which boasts 26 trillion operations per second. Such hardware specifications should be more than sufficient for running our model. Even on the software level, efforts have been made to accelerate deep learning inference on mobile devices [42]. In addition, it must be emphasized that non-cough sounds are much more common than cough sounds, and only when a cough occurs would our cougher recognition algorithm need to be triggered. However, it is unclear how much energy our pre-trained models would consume on a smartphone. Hence, for practical applications, it may be preferable to have the smartphone constantly plugged into an electrical outlet. The deployment of a cough monitoring algorithm consisting of our proposed cougher recognition algorithm on mobile devices would provide a cost-efficient and scalable solution to contact-free continuous cough monitoring that can function reliably even in the presence of ambient coughs.

Finally, we would like to note that our approach also allows simultaneous monitoring of multiple people in a room shared with other, non-monitored people. In this case, cougher identification and verification must be combined. Identification must first be carried out to determine the individual whose enrollment coughs most resemble the input cough. Then, the identified individual must be confirmed via verification to determine whether the input cough truly belongs to that individual, since it may have originated from a non-monitored person, whose coughs are very resembling. Additionally, the number of cough samples required for enrollment is adjustable and may be leveraged to boost the performance of our approach in practical applications.

C. Limitations & Future Work

One limitation of our approach is that the hyperparameters were tuned exclusively using voluntary cough data, which has likely caused a decrease in performance of our models when applied to reflex coughs rather than voluntary coughs. We believe that better results in terms of identification and verification tests with reflex coughs could be obtained if we optimized the hyperparameters on the reflex cough data set instead. We only optimized the threshold of the decision rule employed for verification using reflex cough data.

Additionally, there is a possibility that when the network was trained with the cough reflex data, it learned the characteristics of the acoustic channel in the recordings rather than the characteristics of the cough signals themselves. To avoid this issue and ensure equal acoustic channels across recordings, we would have to obtain recordings with multiple coughers (preferably more than just two) present in a same room at the same time.

Finally, the feasibility of using our pre-trained models on mobile devices needs to be experimentally verified and more thoroughly investigated. For instance, it would be beneficial to know the exact inference time of our model and the energy consumption it induces on different smartphones to assess its applicability. In future work, we plan to integrate our proposed method along with cough detection into a mobile application to evaluate its capabilities in a real-world scenario with patients.

VI. Conclusion

In this work, we present a novel triplet-network-based approach for cougher identification and verification using voluntary and reflex coughs which were recorded using several devices, including a smartphone and tablet. Thus, our work contributes to automated mobile cough monitoring by developing an algorithm which, in conjunction with a cough detection algorithm, would enable contact-free monitoring of multiple patients, even in the presence of ambient coughs. We have found that although the quality of the recording device remains a limiting factor, our approach can achieve high-performance values in both identification and verification tests. In addition, increasing the number of enrollment samples ensures good performance even in identification tests with a large number of coughers or when reflex coughs are used. In light of these results, our work constitutes an additional step towards scalable, cost-efficient, and contact-free cough monitoring for patients in shared environments.

References

[1] J. J. Benich and P. J. Carek, "Evaluation of the patient with chronic cough," *Amer. Fam. Physician*, vol. 84, no. 8, pp. 887–892, Oct. 2011.

[2] A. H. Morice, "ERS guidelines on the assessment of cough," *Eur. Respir. J.*, vol. 29, no. 6, pp. 1256–1276, 2007.

[3] A. C. Ford, D. Forman, P. Moayyedi, and A. H. Morice, "Cough in the community: A cross sectional survey and the relationship to gastrointestinal symptoms," *Thorax*, vol. 61, no. 11, pp. 975–979, 2006.

[4] M. Fujimura, "Frequency of persistent cough and trends in seeking medical care and treatment-results of an internet survey," *Allergol. Int.*, vol. 61, no. 4, pp. 573–581, 2012.

[5] R. S. Irwin, "Overview of the management of cough: CHEST guideline and expert panel report," *Chest*, vol. 146, no. 4, pp. 885–889, 2014.

[6] C. Infante, "Use of cough sounds for diagnosis and screening of pulmonary disease," in *Proc. IEEE Glob. Humanitarian Technol. Conf.*, 2017, pp. 1–10.

[7] L. Archer and H. Simpson, "Night cough counts and diary card scores in asthma," *Arch. Dis. Childhood*, vol. 60, no. 5, pp. 473–474, 1985.

[8] S. Subburaj, L. Parvez, and T. Rajagopalan, "Methods of recording and analysing cough sounds," *Pulmonary Pharmacol.*, vol. 9, no. 5-6, pp. 269–279, 1996.

[9] H. A. Bickerman and S. E. Itkin, "The effect of a new bronchodilator aerosol on the air flow dynamics of the maximal voluntary cough of patients with bronchial asthma and pulmonary emphysema," *J. Chronic Dis.*, vol. 8, no. 5, pp. 629–636, 1958.

[10] S. Birring, T. Fleming, S. Matos, A. Raj, D. Evans, and I. 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, 2008.

[11] D. Liaqat, "Coughwatch: Real-world cough detection using smartwatches," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2021, pp. 8333–8337.

[12] F. Barata, K. Kipfer, M. Weber, P. Tinschert, E. Fleisch, and T. Kowatsch, "Towards device-agnostic mobile cough detection with convolutional neural networks," in *Proc. IEEE Int. Conf. Healthcare Informat.*, 2019, pp. 1–11.

[13] J. B. Soriano, "Prevalence and attributable health burden of chronic respiratory diseases 1990-2017: A systematic analysis for the global burden of disease study 2017," *Lancet Respir. Med.*, vol. 8, no. 6, pp. 585–596, 2020.

[14] J.-M. Liu, M. You, Z. Wang, G.-Z. Li, X. Xu, and Z. Qiu, "Cough detection using deep neural networks," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, 2014, pp. 560–563.

[15] M. Whitehill, J. Garrison, and S. Patel, "Whosecough: In-the-wild cougher verification using multitask learning," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 896–900.

[16] F. Barata, "Automatic recognition, segmentation, and sex assignment of nocturnal asthmatic coughs and cough epochs in smartphone audio recordings: Observational field study," *J. Med. Internet Res.*, vol. 22, no. 7, 2020, Art. no. e18082.

[17] J. H. L. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 74–99, Nov. 2015.

[18] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 4, pp. 788–798, May 2011.

[19] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proc. Odyssey*, 2010, pp. 14–21.

[20] G. Hinton, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[21] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Comput. Speech Lang.*, vol. 60, 2019, Art. no. 101027.

[22] C. Li, "Deep speaker: An end-to-end neural speaker embedding system," *CoRR*, vol. abs/1705.02304, 2017, [Online]. Available: https://arxiv.org/abs/1705.02304

[23] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proc. Interspeech*, 2017, pp. 999–1003.

[24] M. Zhang, Y. Chen, L. Li, and D. Wang, "Speaker recognition with cough, laugh and 'wei'," in *Proc. IEEE Asia-Pacific Sig. Inf. Process. Assoc. Ann. Summit Conf.*, 2017, pp. 497–501.

[25] P. Tinschert, "Prevalence of nocturnal cough in asthma and its potential as a marker for asthma control (MAC) in combination with sleep quality: Protocol of a smartphone-based, multicentre, longitudinal observational study with two stages," *Brit. Med. J. Open*, vol. 9, no. 1, 2019, Art. no. e026323.

[26] J. Korpáš, J. Sadlonová, and M. Vrabec, "Analysis of the cough sound: An overview," *Pulmonary Pharmacol. Therapeutics*, vol. 9, no. 5-6, pp. 261–268, 1996.

[27] B. McFee, "librosa: Audio and music signal analysis in python," in *Proc. 14th Python Sci. Conf.*, vol. 8, 2015, pp. 18–25.

[28] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," in *Proc. 17th Int. Soc. Music Inf. Retrieval Conf.*, ISMIR, 2016, pp. 805–811.

[29] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. Int. Conf. Mach. Learn. Deep Learn. Workshop*, 2015, pp. 1–8.

[30] J. Wang, "Learning fine-grained image similarity with deep ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1386–1393.

[31] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[32] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *Proc. Int. Workshop Similarity-Based Pattern Recognit.*, Springer, 2015, pp. 84–92.

[33] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Comput.*, vol. 29, pp. 1–98, 2017.

[34] M. Meyer, L. Cavigelli, and L. Thiele, "Efficient convolutional neural network for audio event detection," *CoRR*, vol. abs/1709.09888, 2017. [Online]. Available: https://arxiv.org/abs/1709.09888

[35] K. Palanisamy, D. Singhania, and A. Yao, "Rethinking CNN models for audio classification," *CoRR*, vol. abs/2007.11154, 2020. [Online]. Available: https://arxiv.org/abs/2007.11154

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd Int. Conf. Learn. Representations*, 2015.

[37] C. Wu, R. Manmatha, A. J. Smola, and P. Krähenbühl, "Sampling matters in deep embedding learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2859–2867.

[38] K. Musgrave, S. Belongie, and S.-N. Lim, "A metric learning reality check," in *Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 681–699.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd Int. Conf. Learn. Representations*, 2015.

[40] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *J. Mach. Learn. Res. - Proc. Track*, vol. 9, pp. 249–256, 2010.

[41] M. Abadi, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Conf. Operating Syst. Des. Implementation*, ser. OSDI'16, USA: USENIX Association, 2016, pp. 265–283.

[42] N. D. Lane, "DeepX: A software accelerator for low-power deep learning inference on mobile devices," in *Proc. 15th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, 2016, pp. 1–12.