mmFlow: Facilitating At-Home Spirometry with 5G Smart Devices

Aakriti Adhikari; Austin Hetherington; Sanjib Sur
Department of Computer Science and Engineering; University of South Carolina, Columbia, USA
aakriti@email.sc.edu; austinth@email.sc.edu; sur@cse.sc.edu

***Abstract*—Respiratory diseases, like Asthma, COPD, have been a significant public health challenge over decades. Portable spirometers are effective in continuous monitoring of respiratory syndromes out-of-clinic. However, existing systems are either costly or provide limited information and require extra hardware. In this paper, we present *mmFlow*, a low-barrier means to perform at-home spirometry tests using 5G smart devices. *mmFlow* works like regular spirometers, where a user forcibly exhales onto a device; but instead of relying on special-purpose hardware, *mmFlow* leverages built-in millimeter-wave technology in general-purpose, ubiquitous mobile devices. *mmFlow* analyzes the tiny vibrations created by the airflow on the device surface and combines wireless signal processing with deep learning to enable a software-only spirometry solution. From empirical evaluations, we find that, when device distance is fixed, *mmFlow* can predict the spirometry indicators with performance comparable to in-clinic spirometers with $<$5% prediction errors. Besides, *mmFlow* generalizes well under different environments and human conditions, making it promising for out-of-clinic daily monitoring.**

***Keywords*: 5G; Millimeter-Wave; Spirometry; Vibratory Signal; Deep Learning; CNN-LSTM; Decoder.**

I. Introduction

The rapid and accelerated evolution of the telehealth industry has created a demand for more ubiquitous health-sensing tools in recent years [1]. One such tool is the *Spirometer*, which provides objective measures of lung function and has wide applicability in diagnosing Asthma, COPD, Cystic Fibrosis, and other pulmonary diseases [2]. They can also be used to diagnose Dyspnea, *i.e.*, shortness of breath. At-home spirometry tests allow patients and pulmonologists to continuously monitor for recovery, detect changes, and gather long-term lung function evolution. Several spirometers are available for home-use, but they are either costly or provide limited information and require extra hardware [3]–[5].

In this paper, we propose *mmFlow*, a low-barrier means to perform at-home spirometry tests using 5G smart devices. **mmFlow** ***leverages the built-in millimeter-wave technology on ubiquitous mobile devices and designs a software-only spirometry solution.*** *mmFlow* requires the user to hold the device in front of their mouth, inhale their full lung volume, and forcibly exhale until the entire volume is expelled, like regular spirometry tests [6]. It then outputs seven key spirometry indicators and a flow-volume graph [7].

mmFlow's key idea is intuitive: Strong airflow on the device surface creates tiny vibrations, and these vibrations directly affect the phase of the reflected millimeter-wave (mmWave) signal from nearby objects. For example, a 79 GHz device (*i.e.*, signal wavelength: 3.79 mm) will register a 50 μm vibration displacement as a 9.5° phase change. Stronger airflow yields larger vibration and higher phase change. So, if we can analyze the temporal phase change of the signal, we should be able to identify the correlated airflow rate, exhalation volume, and predict the lung function accurately. But *mmFlow* faces two key challenges in bringing this idea into practice.

First, to accurately correlate the airflow with the vibration, the phase change should only be derived from the airflow and needs to be tracked from a nearby static object. In the absence of a static object, we could leverage the user's body as the reflector, but sway motion of the hand or body will introduce spurious phase change, corrupting the airflow-only information. Besides, all spirometers measure the flow rate of air leaving the user's mouth [6]. Yet, a user could hold the device at various distances from the mouth, affecting the vibration due to weaker airflow at a greater distance. So, the challenges here are to compensate for the flow rate loss due to device distance and track the phase change under sway motion.

Second, the relationship between the vibration signal and the spirometry indicators or the flow-volume graph has never been explored before. Intuitively, peak vibration amplitude and frequency could determine the peak airflow rate; but a mathematical model for mapping the vibration signal to all spirometry indicators and flow-volume characteristics would be complex and would require hand-tuning several parameters in practice. So, aiming for such a mathematical model may not only be intractable but also yield inaccurate results.

To overcome these challenges, *mmFlow* proposes two core design techniques. (1) Beamforming, reflector tracking, and distance calibrating modules which provide us accurate vibration estimation: The high-level idea is to use multiple receive antennas in locating and tracking a strong reflector for estimating device's self-vibration and apply a physics-based distance transfer function to accurately predict the vibration due to the air leaving the mouth. (2) A machine-learning model that uses the vibration signal to predict the spirometry results: The high-level idea is to use a deep convolution network that learns, from previous hundreds of examples, the hidden association between the vibration signal and airflow information. The network then can predict the key spirometry indicators and generate a flow volume graph, similar to those found in clinical or high-end home spirometers.

We have prototyped *mmFlow* on an off-the-shelf mmWave device and conducted experiments to verify its performance. Due to the lack of publicly available ground-truth data, we have built a real-world data collection platform integrating a 77–81 GHz device [8] and a clinical flow-volume calibrator

978-1-6654-4108-7/21/$31.00

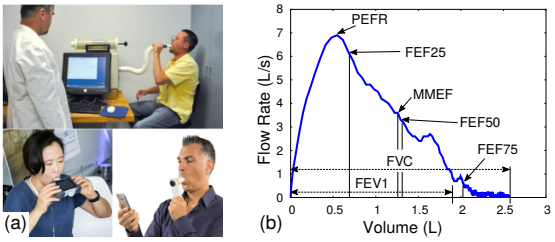

Figure 1: *(a) Current spirometry tests: In-clinic (top); and Portable (bottom). (b) Spirometry indicators and flow-volume graph.*

[9] to collect samples from real, hand-held spirometry tests. Based on our dataset of 1217 spirometry samples, we find that *mmFlow* can predict the spirometry indicators, PEFR, FEV1, and FVC, with median errors of only 0.40 L/s, 0.05 L, and 0.03 L, respectively, under a fixed device distance. Even with the variable distance, variable angle, hand-held sway motion, and background noise from other airflow sources, *mmFlow*'s median prediction errors for PEFR, FEV1, and FVC are only 0.96 L/s, 0.20 L, and 0.10 L. *mmFlow* can generate accurate flow-volume graphs with a median similarity score of 0.96 *w.r.t.* the ground-truth (1 is a perfect match). Besides, *mmFlow*'s prediction accuracy is comparable to a widely used commercial, at-home spirometer [10].

In summary, we make the following contributions: (1) We design a framework that can accurately estimate a 5G smart device's surface vibration characteristics due to the external airflow. (2) We design customized deep convolution networks that use the vibration characteristics to predict the spirometry indicators and flow-volume graphs. Our results demonstrate that *mmFlow* is generalizable under different environments and human conditions. To the best of our knowledge, *mmFlow* is the first system that enables at-home spirometry using mmWave technology on mobile devices. To catalyze the hand-held spirometry research using mmWave, we will open-source the measured dataset through our project repository.

II. Background and Fundamentals

Spirometry Indicators: A spirometer estimates the instantaneous flow (flow rate) of the air leaving the mouth (Figure 1[a]; [6]), and this rate can be integrated over time to get a flow-volume graph, like Figure 1(b). From the graph, a pulmonologist identifies the lung condition as normal, obstructive, or restrictive, and measures the following seven indicators:

► ***Peak Expiratory Flow Rate (PEFR)***: The maximum flow rate during exhalation; healthy adults have a PEFR between 8–10 Liters/second (L/s), and Asthma patients, for example, have an average PEFR of 5 L/s or lower [11], [12].

► ***Forced Vital Capacity (FVC)***: Total air volume exhaled; healthy adults have a capacity between 3.75–5.25 Liters (L), and COPD patients have an average of 2.6 L or lower [13].

► ***Forced Expiratory Volume in 1 second (FEV1)***: Air volume exhaled in the first second of the test; healthy adults exhale 75–80% of their capacity within 1 second, but obstructions in the airways would substantially reduce it [13].

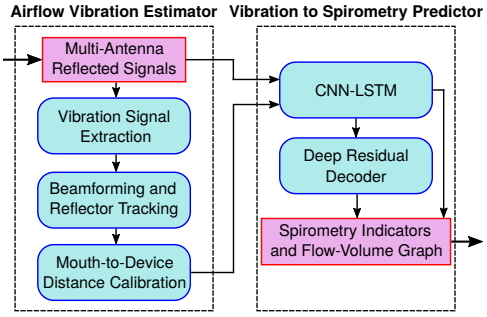

Figure 2: *System overview of* mmFlow.

► ***Forced Expiratory Flows (FEF25, FEF50, FEF75)***: The measure of flow rates at 25%, 50%, and 75% of total capacity and are used in airways narrowing assessments [14].

► ***Maximal Mid Expiratory Flow (MMEF)***: Maximum mid-airflow rate during exhalation and can be used to diagnose minute airways dysfunctions [15].

Millimeter-Wave Devices and Standards: mmWave devices operate at a very high frequency and ultra-wide bandwidth. Currently, there are two most popular mmWave standards: IEEE 802.11ad [16] and 5G NR [17]. Due to the very small signal wavelength (∼3.9 mm for 77 GHz, for example), the signals are extremely sensitive to the device movements, even in the order of few microns. While such high sensitivity poses a challenge to design a robust mmWave network, it creates an opportunity to build systems that can detect tiny vibrations, even due to the external airflow. *mmFlow* leverages this opportunity to estimate human lung function with a hand-held mmWave device that not only provides a low-cost solution but also enables continuous, at-home spirometry tests.

III. *mmFlow* Design

A. Overview

mmFlow ***aims to bring at-home spirometry to ubiquitous mobile mmWave devices by addressing the practical challenges in the hand-held setting.*** It relies on the reflected mmWave signals when a user blows air at the device. Then by processing the signals, *mmFlow* estimates the tiny vibration. To map the vibration to airflow information, it uses a Convolutional Neural Network with Long Short-Term Memory (CNN-LSTM) and a Deep Residual Decoder; the networks predict not only the seven key spirometry indicators but also the flow-volume graph. Figure 2 shows an overview of the *mmFlow* system.

The reflected signals from a nearby reflector, like the wall or user's body, are measured using multiple antennas and are used to extract the device's vibration characteristics. Since under a hand-held setting, the device's relative location *w.r.t.* reflector could change, *mmFlow* adopts a strategy to steer the signal towards the same reflector and track it continuously. Then, it uses a mouth-to-device distance calibration to ensure that the estimated vibration is only due to the air leaving the mouth.

Since there does not exist any model to map the vibration to spirometry results, *mmFlow* designs a data-driven approach that learns the association between the vibration signal and spirometry indicators. Then, the indicators are fed to the deep

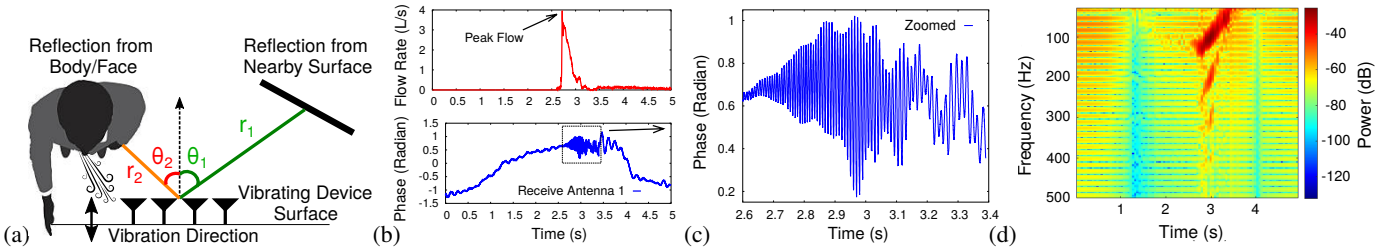

Figure 3: *(a) Millimeter-wave device receives reflections from different objects, and airflow affects the reflected signals. (b) (Top) Flow rate over time of an example exhalation with a peak rate of 3.96 L/s and (bottom) corresponding phase of the reflected signal; (c) Zoomed phase; (d) Spectrogram (time-frequency) plot shows distinct vibration signature due to the airflow.*

residual decoder that outputs the full flow-volume graph. Next, we describe these design components in detail.

B. Airflow Vibration Estimator

Vibration Signal Extraction: Assume that a hand-held mmWave device is continuously transmitting a wireless signal towards a reflector and receiving its reflections (Figure 3[a]). When the user holds the device in a static position, the phase of the reflected signal remains static. However, when airflow strikes the device's surface, the device starts to vibrate, causing a time-variant change in the distance of the device to the reflector. This change in distance results in a change in phase: $\Delta\phi(t) = \{4\pi\Delta d(t)\}/\lambda$, where $\Delta d(t)$ is the temporal change in distance, and λ is the signal wavelength. Figures 3(b–c) show an example of such phase change due to exhalation with 3.96 L/s peak flow rate: We observe a distinct vibratory signature on the reflected signal. Figure 3(d) further analyzes the phase change by plotting the spectrogram, which identifies not only the vibration start and stop time but also the temporal distribution of the vibration amplitude and frequency.

However, the challenge is to identify a strong reflector that the device could use as a reference to track the phase change. In practice, a transmitted wireless signal will reflect off of various objects and the user's body to arrive at the receiver. Estimating the phase change from combined reflections will be erroneous since multiple reflections could add constructively or destructively. So, to separate the reflections from multiple objects, *mmFlow* leverages the signal arrival time measured from a wide-bandwidth mmWave device, similar to [18]. The system uses a radar technique called frequency-modulated continuous-wave (FMCW) to extract the phase of the reflected signal and to isolate signals reflected off different objects. Since wireless signal travels at the speed of light, reflections from objects at a different distance would arrive at slightly different times. Based on the arrival time, *mmFlow* can choose the reflector with the highest reflective strength. As long as the device's relative location *w.r.t.* the reflector remains the same, *mmFlow* can extract the vibration signal accurately.

Beamforming and Reflector Tracking: However, the relative location could change due to the hand or body's swaying motion during the spirometry test. Tracking the phase change from a moving reference yields incorrect estimation of vibration and a wrong spirometry result. To overcome this challenge, *mmFlow* leverages the opportunity that mmWave devices comprise of multiple receive antennas (4 in *mmFlow*) that can measure reflections simultaneously. Due to the antenna separation, the measured signals could be used to localize a reflector in space. Under the device's sway motion, the signals could be steered continuously towards the same reflector to estimate the vibration characteristics accurately.

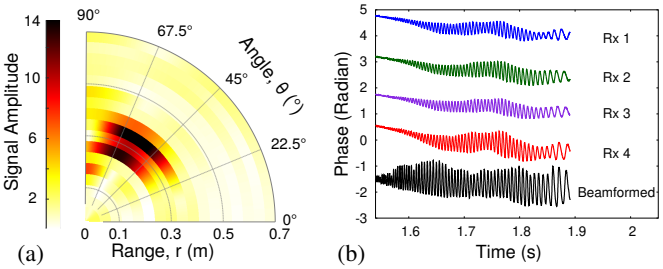

Figure 4: *(a) Range-angle heatmap shows two strong reflectors. (b) Example of vibration signal estimated from 4 receive antennas and beamformed signal towards the strongest reflector.*

To ensure that *mmFlow* could steer the signal towards an optimal location, the signals from all 4 receive antennas are combined using the following range-angle (r-θ) equation [19]:

$$BF(r,\theta) = \sum_{n=1}^{4} v_n(r) \times exp\Big[\frac{-j2\pi}{\lambda}(n-1)dcos(\theta)\Big] \quad (1)$$

where $v_n(r)$ is the reflected signal received at n^{th} antenna from range r, λ is the signal wavelength, and d is the inter-distance between the antennas ($d = \lambda/2$, in *mmFlow*). Figure 4(a) shows an example of a range-angle heatmap estimated by *mmFlow* in an environment with two reflectors, one at 35 cm and 55°, and another at 25 cm and 60° from the device. To estimate the phase change from a single strongest reflector throughout the spirometry test, *mmFlow* uses a reflector tracking scheme leveraging the range-angle estimation.

First, when the user places the device in front of their mouth, and before she starts blowing the air, *mmFlow* uses the measured reflected signals from multiple receive antennas and computationally scans the nearby region to find the strongest reflector. Said differently, it finds the r and θ from Equation (1) for which $|BF(r,\theta)|$ is maximum ($|\cdot|$ is the signal strength):

$$\{r^*, \theta^*\} = argmax\{|BF(r,\theta)|\} \quad (2)$$

Then, every time it receives new reflected signals, *mmFlow* re-applies Equation (1) to estimate a new range-angle heatmap. When air is blown at the device, the reflector's relative location *w.r.t.* the device may shift, and the earlier strongest reflector may no longer remain the strongest. Still, to track the phase

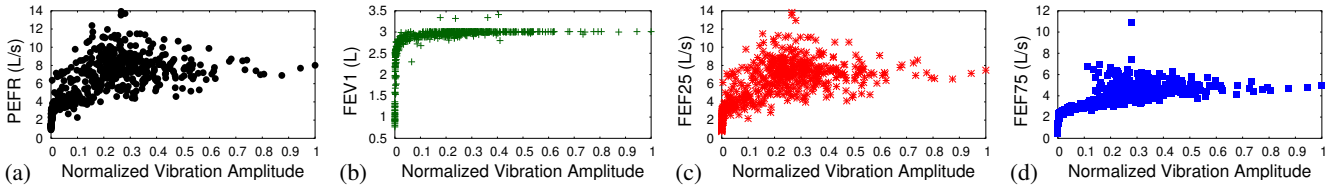

Figure 5: *Relationship between different spirometry indicators and vibration amplitudes across 589 exhalation experiments.*

change from the earlier reflector, *mmFlow* finds few sets of (r_i, θ_i) by sorting the new $|BF(r_i, \theta_i)|$ in descending order and choosing the (r_i, θ_i) based on the shortest Euclidean distance from the earlier reflector's range-angle, (r^*, θ^*). This ensures *mmFlow* could track the phase from a single reflector:

$$\{r_i^*, \theta_i^*\} = \underset{i}{argmin}\{\sqrt{(r^* - r_i)^2 + (\theta^* - \theta_i)^2}\} \quad (3)$$

Finally, *mmFlow* uses the new estimation, (r_i^*, θ_i^*), and applies the receive beamforming at the correct range and angle [19]. Figure 4(b) shows an example of vibration detected by the individual receive antennas and the beamformed signal. Clearly, the beamformed signal improves the quality of vibration signature since beamforming also increases the signal-to-noise ratio. Therefore, the beamformed signal along with the signals from all 4-receive antennas provides additional useful information to analyze vibratory signatures. Besides, *mmFlow* applies a highpass filter to remove the residual low-frequency sway movement of the hand. These ensure an accurate estimation of the vibration signal, even under the user's sway movement or relative location change of the reflector during the spirometry tests.

Mouth-to-Device Distance Calibration: The vibration characteristics are also affected by the distance of the device from the mouth. Larger device distance creates lower vibration since it reduces the air pressure and the corresponding flow rate. To compensate for the flow rate loss, *mmFlow* uses an inverse flow radiation model, similar to [20]. The high-level idea is to apply a transfer function on the measured vibration signal approximated by the laminar flow of air. But in contrast to [20] which leverages audio signals that could reverberate around the user's head, *mmFlow* only applies the transfer to compensate for the mouth-to-device distance. This model works well for a device distance of approximately half of a typical arm-length.

C. Vibration to Spirometry Predictor

Correlation between Vibration and Spirometry Indicators: To map the estimated vibration to the spirometry indicators, *mmFlow* first explores the correlation between them. Intuitively, stronger airflow (*i.e.*, higher values of indicators) yields larger vibrations. To verify this intuition, we use the mmWave device [8] and the flow-volume calibrator that mimics human exhalation [9]. We place the device at an 11 cm distance from the calibrator nozzle on a static table and use the wall behind the setup as our reflector (see Figure 9). We perform 589 spirometry tests emulating exhalation from healthy individuals and different pulmonary disease patients. Thus, our experiments span a wide range of lung functions: PEFR (0.93–13.9 L/s), FEV1 (0.77–3.41 L), *etc.* For each test, we estimate the vibration signal and record the spirometry indicators.

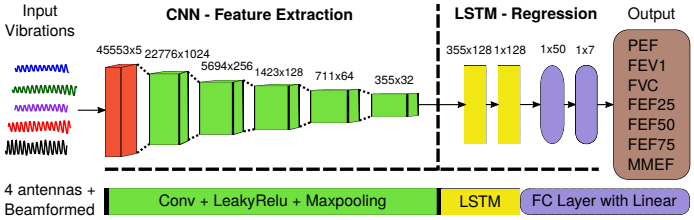

Figure 6: *CNN-LSTM architecture of the* mmFlow *system.*

Figures 5(a–d) show the relationship between four of the spirometry indicators with the normalized vibration amplitude. While we observe that there is a trend in the relationship between volume and rate of airflow and vibration signal, it is hard to capture this relationship using straightforward models. We can also infer that the spirometry indicators are not linearly correlated with the vibration signal. Thus, *mmFlow* aims to learn the correlation using a non-linear model.

Learning Correlation using CNN-LSTM: Instead of a parametric non-linear regression model, which may need hand-crafted tuning of several parameters, *mmFlow* captures the correlation using a data-driven approach. The high-level idea is intuitive: *mmFlow* trains a CNN-LSTM framework by showing hundreds of examples of the mmWave vibration signal and the corresponding ground-truth spirometry indicators. The framework uses CNN to extract the short-term features and LSTM to identify the long-term evolution of vibration so that the network could learn the association between the time-domain vibration signal and the indicators. In what follows, we first describe the CNN-LSTM fundamentals briefly and then discuss the network components in detail.

CNN-LSTM Fundamentals: CNN is the classical machine learning approach that applies deep architecture and a combination of sparser and denser connectivity between layers to find complex, hidden features in the input and learn the association between input and output [21]. While CNN is very useful in identifying the local features in spatial data, it fails to capture the global features in temporal data [22]. LSTM can learn long-term inter-dependencies in the temporal data and can extract the global features [23]. So, the CNN-LSTM combination allows *mmFlow* to map local and global features from the time-domain vibration signal to spirometry indicators.

Figure 6 shows the network architecture in *mmFlow*. *mmFlow* uses one-dimensional CNN as a feature extractor and LSTM with dense layers as a long-term regressor. The CNN-LSTM network takes the vibration signal estimated from the 4 receive antennas and beamformed signal and predicts the

	1DC1	1DC2	1DC3	1DC4	1DC5	LSTM1	LSTM2	FC1	FC2
Filter #	1024	256	128	64	32	128	128	50	7
Filter Size	2	2	2	2	2				
Maxpooling	2	4	4	2	2				
Act. Fcn.	LReLU	LReLU	LReLU	LReLU	LReLU	Tanh	Tanh	ReLU	Linear

Table I: *CNN-LSTM network parameters. 1DC: 1D Convolution; FC: Fully Connected; Act. Fcn.: Activation Function; LRelu: LeakyRelu.*

seven spirometry indicators (Section II).

1. CNN as the Feature Extractor: ***The core purpose of the CNN is to learn relevant features that can map input vibration to output indicators through series of convolutions***. To this end, *mmFlow*'s CNN passes the 1D vibration signal from each receive antenna and beamformed signal through 5 1D convolution layers; the layers apply convolutional filters with a set of weights that slides over the input. Filters are receptive fields, and with a smaller filter size, each convolution layer compresses the vibration signal towards its abstract local features [24]. *mmFlow*'s CNN also employs maxpooling [21] that helps not only to reduce the spatial size of convolved features but also to suppress noisy activations and extract the dominant features that are position invariant in the vibration. Each convolution layer employs a LeakyReLU (LReLU) activation to help tune the network weights better [25]. At the end of the CNN layers, *mmFlow* is able to extract the local feature maps in the input vibration signal. These features are broken down into sequential components and are fed into recurring LSTM units for temporal analysis and indicators' prediction.

2. LSTM as the Regressor: ***The core purpose of the LSTM is to capture long-term temporal dependencies between local features through recurrent connections between units.*** LSTM employs forget gates to selectively forget or remember input features. By backpropagating and adapting the weights through the gates, the LSTM module can retain higher importance temporal features and discard irrelevant ones [23]. To be compatible with the CNN feature size and to train the network better, *mmFlow* uses 2 LSTM layers with unit sizes equal to 4× the CNN's output size. Each LSTM layer employs Tanh activation and is fully connected with the adjacent layers. Finally, the output neurons go through a regression layer with linear activation to predict the actual values of the indicators. Table I summarizes the CNN-LSTM network parameters.

3. Network Loss Function: CNN-LSTM relies on a loss function to appropriately tune the convolution weights. We use Mean Squared Error (MSE) as the loss function that calculates the average squared difference between the predicted and ground-truth indicators, and it accelerates the learning by penalizing weights causing larger errors. *mmFlow* fine-tunes the network by exploring different hyper-parameters of optimization, learning rate, batch size, and epoch. We will discuss the hyper-parameters tuning in Section IV.

Mapping Indicators to Flow-Volume Graph: In addition to the spirometry indicators, clinical and high-end spirometers can also provide flow-volume graphs, such as Figure 1. It can help visually diagnose a patient's condition: Figure 7 shows a few examples of flow-volume graphs for patients under different conditions. Flow-volume graph also indicates if the patient has performed the spirometry test correctly [26]. So, pulmonologists use the graph to not only observe the patient's test performance but also diagnose the patient's lung condition.

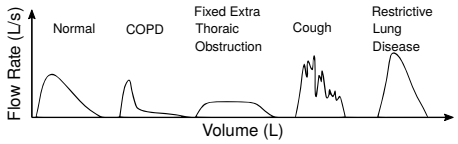

Figure 7: *Flow-Volume graph under different lung conditions.*

To emulate the clinical/high-end spirometers, *mmFlow* aims to produce an accurate flow-volume graph. A straightforward approach could be to use regression or polynomial curve fitting over the seven indicators. But this approach does not work since the volumes (x-axis) corresponding to the predicted flow rates (y-axis) are unknown. Instead, **mmFlow** ***designs a deep residual decoder network for curve learning based on an open-source CDC database on lung function measures [27].*** Then, at run-time, the network takes key indicators as the input and generates a complete flow-volume graph as the output.

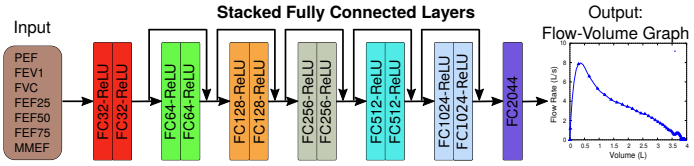

Figure 8: *Deep residual decoder of the* mmFlow *system.*

Figure 8 shows the decoder architecture in *mmFlow*. Each stage of the decoder employs a fully-connected dense layer that expands the input towards the full flow-volume graph. Since the input size is very small (only 7) compared to the output size (*e.g.*, 1000, for a flow-volume graph with resolution 0.01 L and maximum 10 L), *mmFlow*'s decoder relies on 13 fully-connected layers stacked on each other. Each layer takes input from its previous layer to expand the compressed input towards the abstract features of the full flow-volume graph.

Such a large number of stacked layers are necessary to expand the decoder's input to output, but it poses a challenge for the backpropagation signal to penetrate the stacks, reach the previous layers, and adapt weights efficiently during training. So, the network could learn suboptimal mapping and produce incorrect flow-volume graphs. To overcome this challenge, **mmFlow** ***employs skip connections between the stacked layers that enable not only easier backpropagation but also easier learning of the residual mapping between input and output***, ensuring accurate flow-volume graph prediction. The decoder uses ReLU activation for each stacked layer since the spirometry indicators do not have any negative values. Similar to CNN-LSTM, the decoder also relies on a network loss function to appropriately tune the fully-connected layers' weights. To ensure the outliers in the ground-truth data in the CDC database [27] do not affect the network performance significantly, *mmFlow* uses the Mean Absolute Error as the decoder's loss function. We have further discussed the network optimization and hyper-parameters tuning of the decoder network in Section IV.

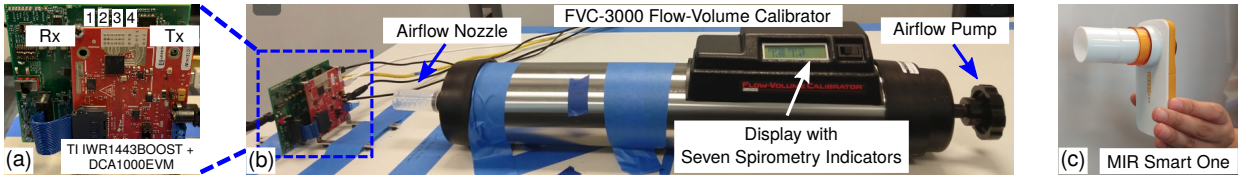

Figure 9: *Ground-truth data collection setup: (a) 77–81 GHz mmWave device with 1 Tx and 4 Rx antennas. (b) FVC-3000, a gold standard flow-volume calibrator, blowing air onto the mmWave device. (c) MIR Smart One device used for at-home spirometry.*

IV. Implementation

Hardware Platform: We implement and evaluate *mmFlow* using real data collected from a 77–81 GHz mmWave device, TI IWR1443BOOST [8], and a Flow-Volume calibrator, Jones Medical FVC-3000 [9] (Figure 9). The mmWave device is equipped with 4 receive antennas that can collect reflected signals independently. It operates on a 4 GHz of bandwidth and can resolve signal arrival times with 0.15 nanoseconds precision. We apply traditional FMCW processing [8] to extract the phase of the reflected signal and use the following FMCW parameters during experiments: start frequency, 77.33 GHz; baseband sampling rate, 5 MSps; frequency ramp slope, 70.3 MHz/μS; number of ADC samples, 256; sweep duration, 56.9 μS; and pulse repetition rate, 1 kHz. We implement *mmFlow* in Matlab and Python running on a host PC and Google Colab Pro environment, which uses the reflected signals as input and predicts the spirometry indicators and graphs as output.

Real Data Collection: Since a spirometry test with a ground-truth spirometer restricts outside airflow, it is challenging to obtain the vibration signal and the ground-truth indicators at the same time. To overcome this challenge, we use the FVC-3000 calibrator to pump air onto the mmWave device and measure the vibration and the ground-truth indicators simultaneously. FVC-3000 generates airflow like human exhalation and is considered the gold standard in the spirometry industry to calibrate all clinical and at-home spirometers [9]. It can emulate a wide variety of lung functionalities with different air pump strokes and measure the flow rate and volume with 1.5 milliliters accuracy [9]. We collect 500 spirometry samples by placing the mmWave device at a fixed distance from FVC-3000, and use them to train *mmFlow*'s CNN-LSTM network. We collect additional 717 spirometry samples in the wild by varying the device distance, angle, background airflow as noise, and in natural hand-held settings: These samples are used only for testing and benchmarking our design.

For the deep residual decoder, we rely on a larger, open-source spirometry database released by the CDC [27]. This database consists of real but anonymized patients' data collected over one year. It contains a rich deposit of spirometry flow-volume and indicators measured from tests on diseased and healthy individuals from different demographics aged 6 to 79 years. We trained our decoder on 155,000 CDC data, and during the run-time, we feed in the CNN-LSTM predicted indicators to estimate the complete flow-volume graph.

Network Training: We train CNN-LSTM and decoder network separately due to unavailability of means to compute ground truth for decoder from FVC-3000 calibrator. We explore different combinations of hyper-parameters by training the networks multiple times: with optimizers, like Stochastic Gradient Descent, Adam, Momentum, and RMSprop; with learning rates, like 10^{-1}, 10^{-2}, 10^{-3}, ...; with mini-batch sizes, like 4, 8, 16, ...; with different in-network activation functions, like ReLU, LReLU, sigmoid, Tanh, ... [28]. We find that CNN-LSTM performed much better with RMSprop optimizer, learning rate of 10^{-4}, mini-batch size of 4, LReLU activations, and adaptive connection drop-out of 25% each of the LSTM and two fully connected layers. For the decoder, we find that network is optimal with the same optimizer, learning rate, but with a mini-batch size of 32, and ReLU activations. Finally, all network architectures are implemented in Python using Keras with TensorFlow as the backend. Both networks take $\sim$20 minutes each to train themselves. They can generate output within 1 minute, which makes *mmFlow* promising for real-time spirometry tests.

V. Performance Evalution

We evaluate *mmFlow* with two metrics: (1) Error in predicting the spirometry indicators by measuring the absolute difference in prediction and ground-truth; (2) Similarity scores (scale: 0 to 1) of flow-volume graphs by measuring the normalized cross-correlation between prediction and ground-truth.

Evaluation Summary: (1) *mmFlow* is highly accurate at a fixed distance with median errors in PEFR, FEV1, and FVC only 0.40 L/s, 0.05 L, and 0.03 L; this is comparable to an in-clinic spirometer. Changing the device distance or angle affects the performance, but we still find that the median errors are 0.58 L/s, 0.4 L, and 0.15 L with $\pm 5^\circ$ relative angle and half of a typical arm-length distance. (2) Even with hand-held sway motions and external noises, *mmFlow* can still predict PEFR with a median error below 1 L/s. *mmFlow*'s decoder consistently outputs accurate flow-volume graphs for healthy and diseased patients and achieves a median similarity score of 0.96. (3) Finally, in comparison to an at-home spirometer, *mmFlow* achieves similar FEV1 and PEFR performance with only 0.03 L and 0.04 L/s median differences, indicating that the system is generalizable under real conditions and achieves prediction results similar to a widely used at-home spirometer.

Indicators' Prediction at a Fixed Device Distance: To evaluate *mmFlow*'s effectiveness in predicting the indicators, we use the test samples at 11 cm device distance (Figure 9[b]). Our experiments are conducted in a quiet office environment without any external airflow sources. We also compare the results with an Artificial Neural Network (ANN) recently proposed in [29] to predict spirometry indicators using audio signals. The ANN is trained on identical vibration datasets.

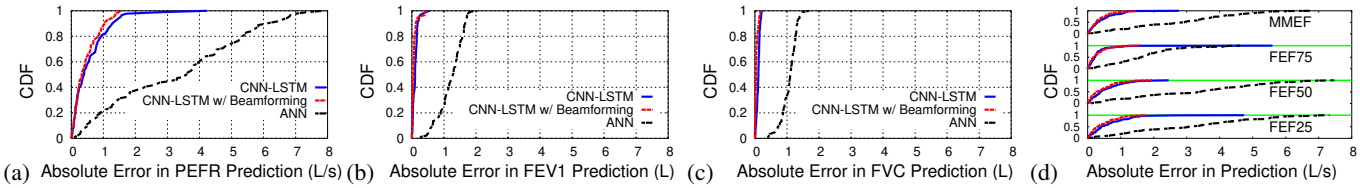

Figure 10: *CDF of prediction errors in seven spirometry indicators.*

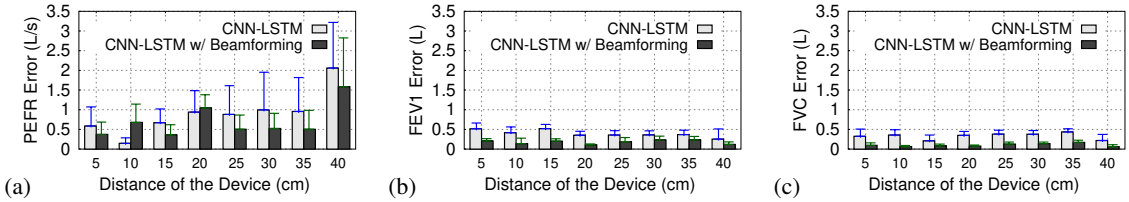

Figure 11: *Impact of device distance on spirometry indicators' prediction.*

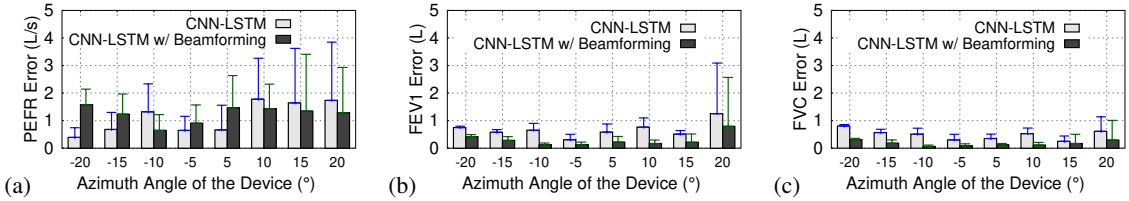

Figure 12: *Impact of device's relative angle on spirometry indicators' prediction.*

Then, we collect the ground-truth indicators and find the absolute difference between the prediction and ground-truth.

Figures 10(a–c) show the CDF of prediction errors for three key indicators, PEFR, FEV1, and FVC. Under ANN, the median and 90^{th} percentile errors for PEFR are 3.60 L/s and 6.06 L/s, respectively. In contrast, under *mmFlow*'s CNN-LSTM, the median and 90^{th} percentile errors are 0.40 L/s and 1.28 L/s only. This improvement highlights the efficacy of a deeper network combining the local and global features in vibration signal. Prediction with additional beamformed signal improves the performance slightly, because the device and reflector are static. Besides, the median prediction error in *mmFlow* for FEV1 and FVC are very low, 0.05 L and 0.03 L only. We also observe similar performance gain in 4 other indicators too in Figure 10(d). These results indicate that *mmFlow* is highly accurate when the device distance is fixed.

Impact of Device Distance: In practice, the device distance could vary during a real spirometry test. So, to systematically study the effect of distance, we place the device at varying distances and collect 20 spirometry test samples at each distance. Recall that, *mmFlow* applies distance calibrations to compensate for the flow rate loss (Section III-B).

Figure 11(a) shows that *mmFlow* achieves the lowest prediction error in PEFR at a 10 cm distance; this is expected since the network is trained with samples from 11 cm distance. We also observe that higher distance causes more error; the error is highest at the 40 cm device distance. This is because, the flow rate compensation model assumes laminar airflow; but at a higher distance, airflow tends to become turbulent affecting the predictions. Still, we see that even at 35 cm (half of typical arm-length), PEFR median prediction error is below 0.5 L/s. Both the FEV1 and FVC predictions have low errors even at different distances, with median error always below 0.5 L.

Impact of Device Angle: Intuitively, vibration due to airflow is the strongest when the device is at 0° angle *w.r.t.* the mouth. In practice, the relative angle could vary depending on the way a user holds the device. To test this effect, we vary the relative azimuth angle: ±5°, ±10°, ±15°, and ±20°. Beyond ±20°, the airflow is too weak to generate any vibration signatures.

Figures 12(a–c) show that prediction error is the lowest when the device has a relative angle of ±5°; the median errors for PEFR, FEV1, and FVC are 0.65 L/s, 0.4 L, and 0.15 L, respectively. Higher relative angle introduces more error, but we still find that the median errors for PEFR, FEV1, and FVC under ±20° cases are within 1.35 L/s, 0.6 L, and 0.3 L.

Impact of Hand-held Sway Motion: We now evaluate the effect of hand-held natural sway motion, by hand-holding the device at different distances. For each test, *mmFlow* runs the beamforming and tracking (Section III-B) to estimate the vibration and uses the CNN-LSTM to predict the indicators.

Figure 13(a) shows the prediction errors of PEFR, FEV1, and FVC with CDF plots. We observe that even when the device is under natural sway movement, the median PEFR prediction error is still close to 1 L/s only. Moreover, the prediction for both the FEV1 and FVC are still accurate: Median errors are within 0.04 L and 0.05 L only. These results demonstrate that *mmFlow*'s beamforming, tracking, and CNN-LSTM can generalize well under practical hand-held settings.

Impact of Environmental Noise: While clinical spirometry tests could be performed in a noise-free environment, it is hard to ensure that for at-home tests. To evaluate the impact of noises, we introduce worst case noise sources in our experimental setup: A loudspeaker playing music nearby

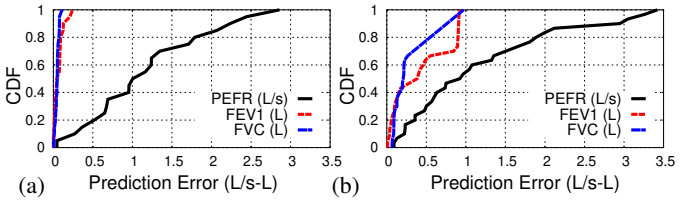

Figure 13: *Impacts of other sources: (a) Hand-held sway movement. (b) Background noise from a loud speaker and spinning fan.*

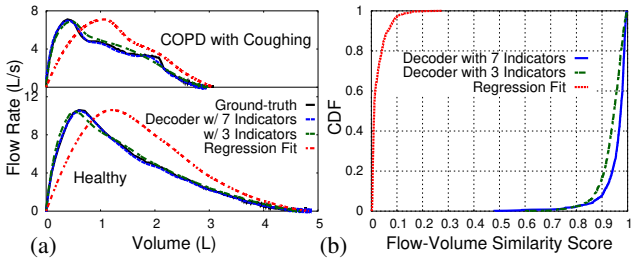

Figure 14: *(a) Two flow-volume graphs approximated by* mmFlow*'s decoders and regression fit method. (b) Similarity score of flow-volume graph prediction across 871 test samples.*

generating background vibrations, and a spinning table fan at 11 cm distance from device generating background airflow.

Figure 13(b) shows that spurious vibrations introduce higher prediction errors. While the median errors for PEFR, FEV1, and FVC are still very small, 0.93 L/s, 0.38 L, and 0.21 L, respectively, the worst case error could be up to 3.42 L/s, 0.92 L, and 0.97 L. We note that these experiments stress test *mmFlow* by placing external vibration sources very close to the setup. As we move the noise sources away, we do not observe any significant performance changes.

Accuracy of Flow-Volume Graph Prediction: Recall that *mmFlow*'s decoder uses the indicators to predict a flow-volume graph. To understand the benefit of the decoder, we also implement a parametric regression based flow-volume graph prediction. Then, we find the similarity between predicted and ground-truth flow-volume graphs by measuring normalized cross-correlation on the scale of 0 to 1, where 1 means the predicted graph matches with the ground-truth perfectly.

Figure 14(a) shows two examples of ground-truth flow-volume graphs and predictions from decoder and regression fit. Clearly, regression fit fails to identify the correct flow-volume in both the cases. But the decoder predicts accurate flow-volume for the healthy and diseased cases and works with less than seven indicators too, so that it can be used with cheap off-the-shelf spirometers with limited indicators. Figure 14(b) further shows that the regression fit achieves a similarity score of only 0.06, even at the 90^{th} percentile, indicating that the method produces incorrect flow-volume graphs. In contrast, *mmFlow* can predict accurate flow-volume graphs with a median similarity score of 0.97. Even with limited indicators, the median similarity score is still 0.96, indicating the effectiveness of a deep stacked residual learning model.

Comparison with a Commerical At-Home Spirometer: Finally, we compare *mmFlow* with an existing home spirometer, MIR Smart One (Figure 9[c]; [10]). This device is recommended by pulmonologists for continuous, at home spirometry tests [10]. However, it can only measure two of the seven key indicators, PEFR and FEV1; so, we only compare these results from *mmFlow*. Figures 15(a–b) show that for both PEFR and FEV1, *mmFlow*'s prediction error is similar to the MIR Smart One. The median FEV1 prediction error for *mmFlow* with a beamformed signal is 0.05 L, and the 90^{th} percentile is 0.13 L only. The median PEFR prediction error for *mmFlow* is also small, 0.33 L/s. It is only 0.04 L/s higher than the MIR Smart One. These results demonstrate that *mmFlow* is well generalizable under real conditions with different devices and environmental conditions, and its indicators' prediction results are similar to a widely used at-home spirometer.

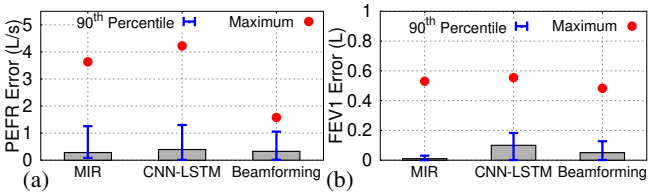

Figure 15: *Comparison with MIR Smart One home spirometer.*

VI. Related Work

Sensing Vital Signs from Wireless Signals: Recent works have been able to wirelessly monitor human vital signs, like heart rate, breathing, and blood pressure [30]–[33]. Wireless signal has also been used to identify behavioral changes, and monitor mental health, and sleep [34], [35]. Recent works have also developed contactless lung function monitoring systems, but they are either expensive or require extra hardware [4], [36]. So, researchers have proposed low-cost means to measure lung function using audio signals [20], [26], [29]. However, these systems are not only highly susceptible to noise and motion but also provide limited spirometry indicators. In contrast, *mmFlow* utilizes mmwave sensor to enable a contactless method of lung function monitoring that provides a low-cost solution, is not affected by noise and motion, and provides all key spirometry indicators and complete flow-volume characteristics of a spirometry test.

Learning Biomarkers: Researchers have been leveraging recent advancements in deep learning to infer hidden biomarkers in ECG, PPG, SCG, *etc.* [22], [37], [38]. Learning has been applied in the areas of image and video processing to estimate human pose, age, and general activities [39], [40]. Recently [41] proposed a CNN-LSTM model for detecting COVID-19 from X-ray images automatically. Besides, deep learning has been used in identifying and monitoring behavior at home [42], learning sleep stages or monitoring sleep quality [43], [44], predicting lung functions [29], learning heart mechanics [45], extracting multi-person breathing signals [46], detecting falls [47], and recognizing emotions [48]. *mmFlow* leverages the advancements in CNN-LSTM and stacked network architectures to facilitate at-home spirometry with cheap and ubiquitous mobile mmWave devices.

VII. Conclusion

This work demonstrates that *mmFlow* can be a promising software-only solution in bringing at-home spirometry to ubiquitous mobile devices. *mmFlow* analyzes the vibration caused by forcible airflow and predicts spirometry indicators from the vibration. It combines wireless signal processing with deep learning and provides not only the key indicators found in clinical/at-home spirometers but also the flow-volume graphs used by pulmonologists to evaluate the patient's condition visually. Our experimental results show that *mmFlow* performs close to existing at-home spirometers and generalizes well in different environments and human conditions. We believe *mmFlow* can be a key solution to transform 5G smartphones into reliable at-home spirometers in the post-COVID era.

VIII. Acknowledgement

We sincerely thank the reviewers for their comments and feedback. The authors were partially supported by the NSF under grant CNS-1910853 during this work.

References

[1] Darren Roblyer, "Perspective on the Increasing Role of Optical Wearables and Remote Patient Monitoring in the COVID-19 Era and Beyond," *Journal of Biomedical Optics*, vol. 25, no. 10, 2020.

[2] Gundersen Health System, "Diagnosis and Treatments for Respiratory and Lung Disorders," 2021. [Online]. Available: https://bit.ly/3umLTbk

[3] EasyOne Air, "EasyOne Air," 2020. [Online]. Available: https://nddmed.com/products/spirometers/easyone-air

[4] Microlife, "Asthma Monitor," 2020. [Online]. Available: https://bit.ly/3sg2vjg

[5] MFI Medical, "Spirometry, Oximetry and Mobile-Health," 2020. [Online]. Available: https://www.spirometry.com/

[6] CDC, "Spirometry Procedures Manual," 2011. [Online]. Available: https://bit.ly/37rcc6t

[7] Ranu Harpreet, et al., "Pulmonary Function Tests," *Ulster Medical Journal*, vol. 80, no. 2, 2011.

[8] Texas Instruments, "IWR1443 Single-Chip 76-GHz to 81-GHz MmWave Sensor Evaluation Module," 2020. [Online]. Available: https://www.ti.com/tool/IWR1443BOOST

[9] Jones Medical, "Flow-Volume Calibrator (FVC-3000)," 2017. [Online]. Available: https://www.jonesmedical.com/product/flow-volume-calibrator-fvc-3000

[10] MFI Medical, "MIR Smart One Spirometer," 2020. [Online]. Available: https://mfimedical.com/products/mir-smart-one-spirometer

[11] Ian Gregg, et al., "Peak Expiratory Flow in Normal Subjects," *British Medical Journal*, vol. 3, no. 5874, 1973.

[12] William M Tierney, et al., "Assessing Symptoms and Peak Expiratory Flow Rate as Predictors of Asthma Exacerbations," *Journal of General Internal Medicine*, vol. 19, no. 3, 2004.

[13] Nuvoair, "Do You Know How to Interpret the Results of Your Spirometry Test?" 2021. [Online]. Available: https://bit.ly/2ZHHIZz

[14] Raji Hanieh, et al., "Forced Expiratory Flow at 25-75% as a Marker for Airway Hyper Responsiveness in Adult Patients with Asthma-like Symptoms," *Tanaffos*, vol. 17, no. 2, 2018.

[15] Majid Mirsadraee, et al., "Diagnosis of Chronic Obstructive Pulmonary Disease Earlier than Current Global Initiative for Obstructive Lung Disease Guidelines Using a Feasible Spirometry Parameter," *Chronic Respiratory Disease*, vol. 10, no. 4, 2013.

[16] IEEE Standards Association, "IEEE Standards 802.11ad-2012, Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band," goo.gl/r2JeYd, 2012.

[17] 3GPP: A Global Initiative, "The Mobile Broadband Standard: Release 15," 2017. [Online]. Available: http://www.3gpp.org/release-15

[18] Fadel Adib, et al., "Smart Homes that Monitor Breathing and Heart Rate," in *ACM CHI*, 2015.

[19] Jingwei Xu, et al., "Joint Range and Angle Estimation Using MIMO Radar With Frequency Diverse Array," *IEEE Transactions On Signal Processing*, vol. 63, no. 13, 2015.

[20] Eric C Larson, et al., "SpiroSmart: Using a Microphone to Measure Lung Function on a Mobile Phone," in *ACM UbiComp*, 2012.

[21] Yann LeCun, et al., *Convolutional Neural Networks for Images, Speech and Time-Series*, 4th ed. AT&T Bell Laboratories, 1995.

[22] Dwaipayan Biswas, et al., "CorNET: Deep Learning Framework for PPG-Based Heart Rate Estimation and Biometric Identification in Ambulant Environment," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 13, no. 2, 2019.

[23] Xiaobin Zhang, et al., "A Combination of RNN and CNN for Attention-based Relation Classification," *Proc. Computer Science*, vol. 131, 2018.

[24] Amin Khorram, et al., "End-to-End CNN+LSTM Deep Learning Approach for Bearing Fault Diagnosis," *Applied Intelligence*, vol. 51, 2020.

[25] Bing Xu, et al., "Empirical Evaluation of Rectified Activations in Convolutional Network," 2015. [Online]. Available: https://arxiv.org/abs/1505.00853

[26] Mayank Goel, et al., "SpiroCall: Measuring Lung Function over a Phone Call," in *ACM CHI*, 2016.

[27] Center for Disease Control and Prevention, "NHANES Comprehensive Data List," 2014. [Online]. Available: https://wwwn.cdc.gov/nchs/nhanes/search/datapage.aspx

[28] Tong Yu, et al., "Hyper-Parameter Optimization: A Review of Algorithms and Applications," 2020. [Online]. Available: https://arxiv.org/abs/2003.05689

[29] Xingzhe Song, et al., "SpiroSonic: Monitoring Human Lung Function via Acoustic Sensing on Commodity Smartphones," in *MobiCom*, 2020.

[30] Alizadeh Mostafa, et al., "Remote Monitoring of Human Vital Signs Using mm-Wave FMCW Radar," *IEEE Access*, vol. 7, 2019.

[31] Dong Shuqin et al., "A Millimeter-Wave Doppler Sensor for Bio-Signals Detection," in *2018 IEEE MTT-S International Wireless Symposium (IWS)*, 2018.

[32] Ahmad Adeel et al., "Vital Signs Monitoring of Multiple People Using a FMCW Millimeter-Wave Sensor," in *2018 IEEE Radar Conference (RadarConf18)*, 2018.

[33] Zhao Peijun et al., "Heart Rate Sensing with a Robot Mounted mmWave Radar," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

[34] Zhicheng Yang, et al., "Vital Sign and Sleep Monitoring Using Millimeter Wave," *ACM Trans. on Sensor Networks*, vol. 13, no. 2, 2017.

[35] Bettina S. Husebo, et al., "Sensing Technology to Monitor Behavioral and Psychological Symptoms and to Assess Treatment Response in People With Dementia. A Systematic Review," *Frontiers in Pharmacology*, vol. 10, no. 1699, 2020.

[36] Lorenzo Scalise, et al., "Wireless Sensing for the Respiratory Activity of Human Beings: Measurements and Wide-band Numerical Analysis," *Journal of Antennas and Propagation*, vol. 2013, no. 396459, 2013.

[37] Attila Reiss, et al., "Deep PPG: Large-Scale Heart Rate Estimation with Convolutional Neural Networks," *Sensors*, vol. 19, no. 14, 2019.

[38] Eoin Brophy, et al., "Optimised Convolutional Neural Networks for Heart Rate Estimation and Human Activity Recognition in Wrist Worn Sensing Applications," 2020. [Online]. Available: https://arxiv.org/abs/2004.00505

[39] Yu Xiang, et al., "PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes," 2018. [Online]. Available: https://arxiv.org/abs/1711.00199

[40] Chenjing Yan, et al., "Age Estimation Based on Convolutional Neural Network," in *Advances in Multimedia Information Processing*, 2014.

[41] Md. Zabirul Islam, et al., "A Combined Deep CNN-LSTM Network for the Detection of Novel Coronavirus (COVID-19) using X-ray Images," *Informatics in Medicine Unlocked*, vol. 20, no. 100412, 2020.

[42] Chen-Yu Hsu, et al., "Enabling Identification and Behavioral Sensing in Homes using Radio Reflections," in *ACM CHI*, 2019.

[43] Chen Yu Hsu, et al., "Zero-Effort In-Home Sleep and Insomnia Monitoring using Radio Signals," in *ACM IMWUT*, 2017.

[44] Mingmin Zhao, et al., "Learning Sleep Stages from Radio Signals: A Conditional Adversarial Architecture," in *ICML*, 2017.

[45] Unsoo Ha, et al., "Contactless Seismocardiography via Deep Learning Radars," in *ACM MobiCom*, 2020.

[46] Shichao Yue, et al., "Extracting Multi-Person Respiration from Entangled RF Signals," in *ACM IMWUT*, 2018.

[47] Yonglong Tian, et al., "RF-Based Fall Monitoring Using Convolutional Neural Networks," in *ACM IMWUT*, 2018.

[48] Mingming Zhao, et al., "Emotion Recognition using Wireless Signals," in *ACM MobiCom*, 2016.