IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 12 March 2024, accepted 1 April 2024, date of publication 4 April 2024, date of current version 12 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3384983

RESEARCH ARTICLE

Breathing Rate Classification Using Piezoresistive Sensor Utilizing Continuous Wavelet Transform and Lightweight CNN

KHUSHI GUPTA, SREENIVASA REDDY YEDURI, (Member, IEEE), AND LINGA REDDY CENKERAMADDI, (Senior Member, IEEE)
ACPS Research Group, Department of Information and Communication Technology, University of Agder, 4879 Grimstad, Norway

Corresponding author: Linga Reddy Cenkeramaddi (linga.cenkeramaddi@uia.no)

This work was supported by Indo-Norwegian Collaboration in Autonomous Cyber-Physical Systems (INCAPS) of the International Partnerships for Excellent Education, Research and Innovation (INTPART) Program from the Research Council of Norway under Project 287918.

ABSTRACT The breath rate can now be monitored remotely due to the advancements in digital stethoscope sensor technology, signal processing, and machine learning. Automatic breathing rate classification, on the other hand, provides additional benefits in medical diagnostics. In this paper, a lightweight convolutional neural network is proposed for automatic breathing rate classification utilizing the piezoresistive sensor data. In the proposed work, the raw signals from the piezoresistive sensor are pre-processed using a continuous wavelet transform to generate the corresponding images. These images are then fed into a lightweight convolutional neural network, which efficiently classifies the breathing rate into six classes based on the number of breaths per minute. Through extensive results, we show that the proposed model results in a classification accuracy of 96.40% which is higher than all the benchmark models considered in this paper. We also evaluate the performance of the proposed model using edge computing devices such as Raspberry Pi, Nvidia AGX Xavier, and Nvidia Jetson Nano.

INDEX TERMS Breathing rate classification, breathing sensor, continuous wavelet transform, deep convolutional neural network, machine learning, piezoresistive sensor.

I. INTRODUCTION

Breathing rate, heart rate, temperature, and blood pressure constitute the four most important vital signs to check the healthy functioning of a human body [1], [2]. Respiration rate is a very important vital sign that is sensitive to various underlying physical (cardiac events, fatigue, pneumonia, cold, etc.) and emotional (stress) conditions of the body [3]. Here, respiration rate defines the rate at which a person breathes in a minute. Respiratory rate is more sensitive with respect to other vital signs [4] but still is not monitored on a regular basis. An abnormal breathing rate, especially when the person is idle, may indicate a severe health issue that may need proper treatment on time.

The associate editor coordinating the review of this manuscript and approving it for publication was Lorenzo Mucchi.

Respiration rate is one of the vital signs that changes at the earliest when compared to others [4]. It helps in detecting various underlying diseases at the initial phase, hence, its continuous monitoring is essential. Generally, people only monitor these vital signs when they have any symptoms but if these signs are monitored on a routine basis a lot of diseases can be detected in their early stage and their treatment can be started on [5]. Respiration rate is a better indicator than temperature, blood pressure, and pulse rate, especially in detecting cardio-pulmonary collapse. This is due to the fact that respiration rate changes a lot in such conditions and hence the change can be easily recorded a clear distinction can be placed between healthy and abnormal patients and proper medical care can be provided accordingly [6]. But still, this vital sign is neglected and till today in many places, it's still counted manually [7] which is an inaccurate and time-consuming method.

A breathing rate between 12-25 breaths per minute (BPM) is considered normal, less than 12 BPM as low, and higher than 25 BPM as high. Breathing rate is an important indicator to represent the level of health and fitness of a person. Motivated by this, many approaches have been proposed in the literature for breathing rate monitoring which are described below.

A. SURVEY

A study on various existing methods for continuous monitoring of respiratory rate and gas exchange has been carried out in [8]. Afterward, every method is described with respect to pros, cons, and market use. In [9], various non-contact and contact methods of respiration have been reviewed and analyzed on the basis of hygiene maintenance and comfort of the subject while recording as well as the accuracy. In [10], a study over contact-based respiratory systems has been presented based on factors such as sound, air temperature, humidity, chest wall movements, etc. Most of the literature states that contact methods are uncomfortable for the subject, and may become subjective over the setup but give accurate measurements due to the contact, whereas non-contact methods have no such discriminatory factors, subjects are more comfortable and can be equal or less accurate than the existing contact-based methods. A literature survey on the breathing rate (BR) estimation methods using ECG and PPG signals has been presented in [11]. In that survey, the authors started with a description of the structure of the BR algorithms followed by their performance and finally, the possible future scope was presented. Performance comparison of 134 algorithms has been assessed in [12] for the respiration rate estimation from both ECG and PPG signals. It has been concluded that the usage of PPG data is better than ECG data. Further, the time domain analysis is suggested to be better than the frequency domain analysis. Finally, the authors have suggested the usage of fusion methods for performance improvement.

B. CONTACT

A novel method based on multiple autoregressive models has been proposed in [13] for respiration rate monitoring from PPG signals. The method has been tested on two independent datasets including recordings from adult and pediatric patients in a hospital. It has been demonstrated that the proposed method in [13] achieved comparable performance to the existing methods with a mean absolute error of 4.0 and 1.5 on the adult and pediatric patient datasets, respectively.

In [14], the heart rate and breathing rate signals have been utilized for the stress level detection of car drivers. Here, an adrenergic sensor coupled to a chest strap has been utilized to collect the heart rate and breathing rate readings. The effectiveness of heart rate and breathing rate is evaluated and when they were used together resulted in a mean average accuracy of 70%. Given that the monitoring of heart activity is less complex than the monitoring of breathing, a variety of algorithms have been developed to estimate breathing activity from heart activity. An increase in the usage of smartwatches with in-built Photoplethysmogram (PPG) has led to the continuous monitoring of respiration rate. Motion artifact correction along with machine learning has been utilized in [15] for respiration rate monitoring from the features extracted from the PPG signal. It has been analyzed through the hyperparameter optimization that Gaussian Process Regression (GPR) with Fit a GPR model feature extraction algorithm results in improved performance when compared to other combinations [15].

C. NON-CONTACT

Over recent developments of COVID-19, breathing rate detection and monitoring have become more essential leading to more research in respiration rate monitoring [16], [17]. In [18], the authors have focused on the drawbacks of contact-based systems and presented a contactless approach to measure the breathing rate using a video to analyze the signals. The face detection algorithm is used to focus on the chest region and then peak-to-peak measurement is used to calculate the breathing rate from the signal giving an accuracy of about 84.66%. In [19], a software-defined radio (SDR) based platform has been proposed for analyzing breathing, hand movement, and coughing using the variations in the OFDM subcarriers. This platform used zero-cross detection to detect the normal breathing rate of 20 BPM, peak detection to detect the slow breathing rate of 10 BPM, and Fourier transformation to detect the fast breathing rate of 28 BPM. Feature extraction has been used in [20] to assess the energy distribution of the signal which belongs to three breathing state classes along with nonlinear time series feature extraction on the reflected signal to the process and obtain an accuracy of nearly 97%. In [21], a single-channel continuous wave radar with confidence intervals has been incorporated to provide a valid breathing rate estimation. Retrieval of Doppler frequency in the range of breathing rate and averaging the dominating Doppler frequencies throughout time was used to estimate the breathing rate with the use of mathematical modeling and simulations. To avoid the problem of noise interference, the bootstrap method is used which results in 95% confidence interval for breathing rate estimation [21]. In [22], a video-based respiration and activity monitoring system has been proposed to monitor a primary subject during its sleeping time. In the proposed system, a video camera and intelligent algorithms have been utilized to monitor the activity level and extract the breathing signal [22]. Initially, these breathing signals passed through three main processing phases which are the reading and plotting phase, the feature extraction phase using Gammatore Frequency Cepstral Coefficients (GFCC), and the classifying phase using Gaussian Mixture Model (GMM). Then, the classification results can be used to give an alert to the E-Health system. In [23], TR-BREATH

has been introduced, a contact-free breathing monitoring system using time reversal (TR). The authors have exploited the channel state information from Wi-Fi signals to monitor the breathing rate of either a single person or multiple persons present either in LoS or NLoS. Further, the Root-MUSIC algorithm has been utilized to extract the features from TR resonating strength (TRRS) projected with CSI. Through experiments, the authors have shown to achieve an accuracy of 99% for single person NLoS scenario and a mean accuracy of 98.65% for a dozen people in the LoS scenario.

Focusing on the limitations of contact-based methods and stating them as uncomfortable, the authors in [24] have proposed a camera-based approach to monitoring the respiration rate by analyzing the facial videos to track the source. In that work, CEEMDAN scheme has been utilized for signal decomposition to extract the Intrinsic Mode Functions which are then processed using ML. It has been demonstrated through results that an RMSE of 2.30 BPM is obtained with the proposed method. A vector network analyzer-based continuous wave radar system has been proposed in [25] for contactless monitoring of the breathing rate. Thereafter, different feature extraction methods and machine learning models are investigated on the collected radar sensor data. Moreover, time-frequency feature extraction methods such as short-term Fourier transform and continuous wavelet transform have been implemented. The proposed system in [25] has investigated 31 people with low/normal/high breathing rates. In [26], the authors have analyzed the speech breathing rate obtained from the speech recordings for the early detection of the disease and emotions. Further, the Cepstogram matrix is used as the feature matrix of speech frames corresponding to breath or non-breath. Finally, the Support Vector Machine with a Radial kernel is used as the classifier for classifying the breath and non-breathing signals. The model yields an F1 score of 89% and a root mean square error (RMSE) of 4.5 breaths per minute.

Most of the works presented in the literature are complex sensing systems but CNN reduces the complexity and trains on fewer parameters giving equal or higher accuracy as mentioned ahead. Further, convolutional neural network (CNN) has been utilized in many applications such as coal mining face point cloud segmentation [27], public management [28], Electrocardiogram (ECG) signal processing [29], etc. Motivated by the above discussion, in this paper, a deep convolutional neural network is proposed that works on a Piezoresistive sensor dataset that contains the breathing rate signals. Using breathing rate signals as images and feeding them to a lightweight convolutional neural network over the Piezoresistive sensor dataset is a new attempt in the field of vital signs detection with high overall classification accuracy. The following are the major contributions of this work:

- Time-frequency analysis using a continuous wavelet transform is performed on the raw voltage signals from a Piezoresistive chest sensor.

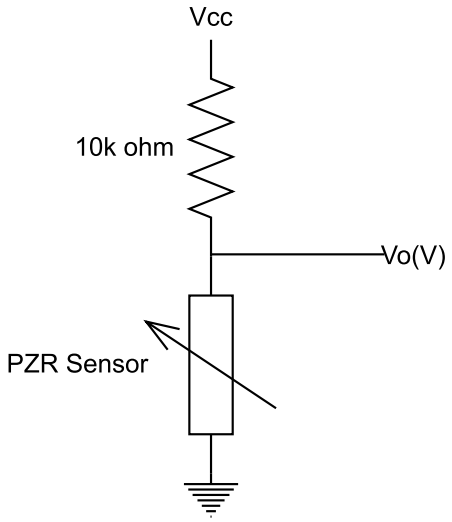

FIGURE 1. Voltage Divider circuit to obtain voltage values from the Piezoresistive sensor.

- A deep convolutional neural network is proposed, along with appropriate data split, for accurate classification of breathing rate into six different classes.
- The proposed model is then evaluated in terms of size, validation accuracy over 5 folds, training accuracy, and the total number of parameters (trainable and nontrainable parameters).
- The proposed model is also deployed in edge computing devices such as Raspberry Pi, AGX Xavier, Nvidia Jetson Nano, and Intel Core Processor IBRS.
- Finally, when deployed on each edge computing device, the proposed method is compared to existing methods in terms of inference time.

The rest of the paper is organized as follows. Section II provides the dataset details and pre-processing, the architecture of the convolutional neural network, and other benchmark models considered in this work. Section IV presents the comparison results of the proposed model with the benchmark models in terms of accuracy and inference time. Finally, the conclusion and possible future works are discussed in Section V.

II. DATASET DETAILS AND PRE-PROCESSING

We consider the publicly available Piezoresistive dataset [30] for the classification of breathing rate. The proposed method of classification consists of data pre-processing and training using the CNN Model. The pre-processing involves balancing the data, arranging the data into data frames, and later feeding it in a CWT function to obtain frequency time plots for better results. Later, the proposed CNN model is described layer-wise. Finally, the architecture of benchmark models is described.

A. PIEZORESISTIVE BREATHING SENSOR DATASET

The Piezoresistive sensor used in [30] (Flexiforce) gives an output resistance that varies based on the force applied

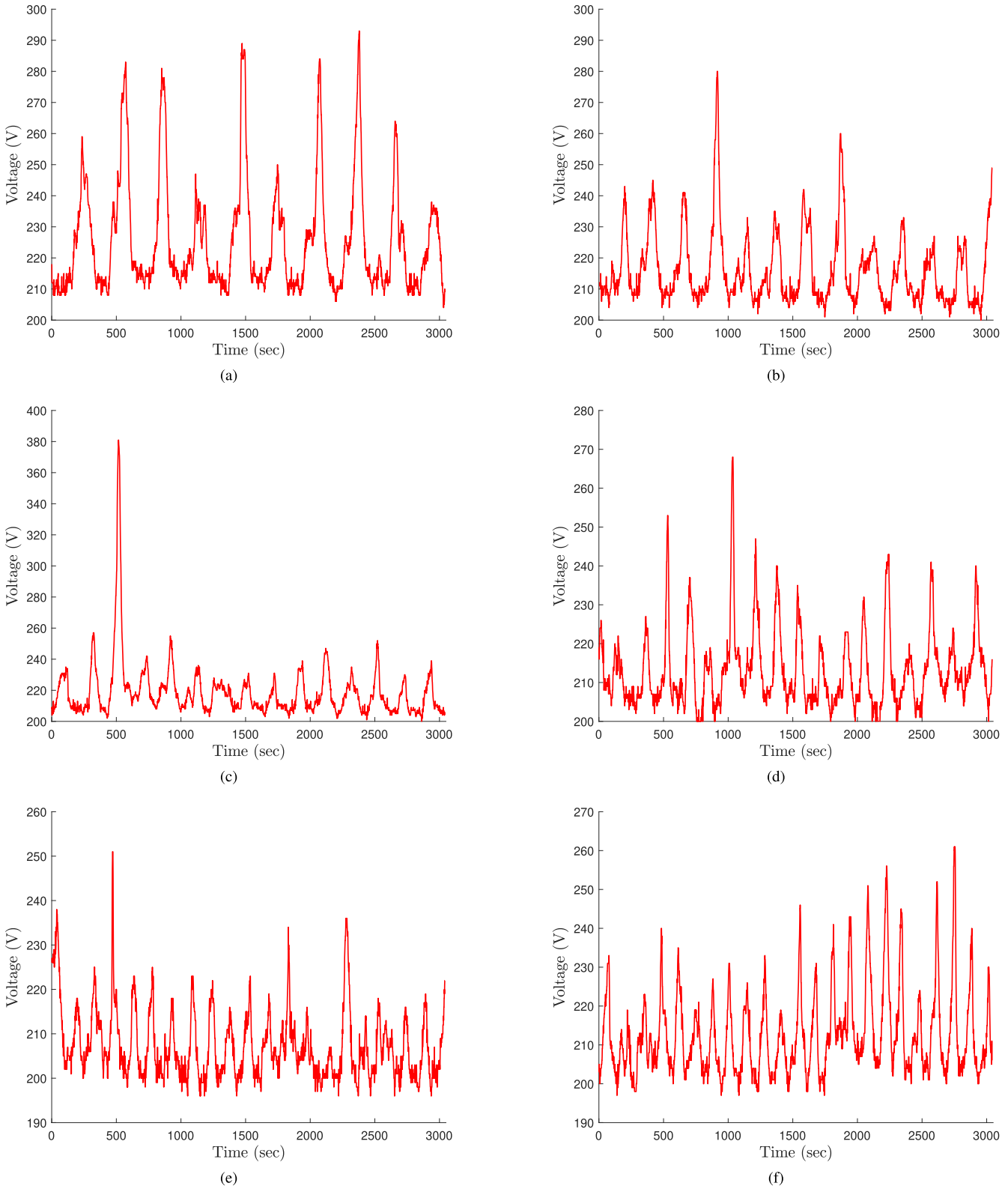

FIGURE 2. Raw signals from Piezoresistive sensor corresponding to class (a) 10 BPM, (b) 12.5 BPM, (c) 15 BPM, (d) 17.5 BPM, (e) 20 BPM, and (f) 22.5 BPM.

across it. The applied force is obtained as

$$\text{Force (g)} = \sqrt[0.827]{\frac{399.88}{R(\text{k}\Omega)}} \tag{1}$$

where R is the resistance with a range from 2KΩ to 10KΩ. In (1), the values are obtained by observing the resistance-force curve provided by the manufacturer which provides the relation between the resistance drop and the

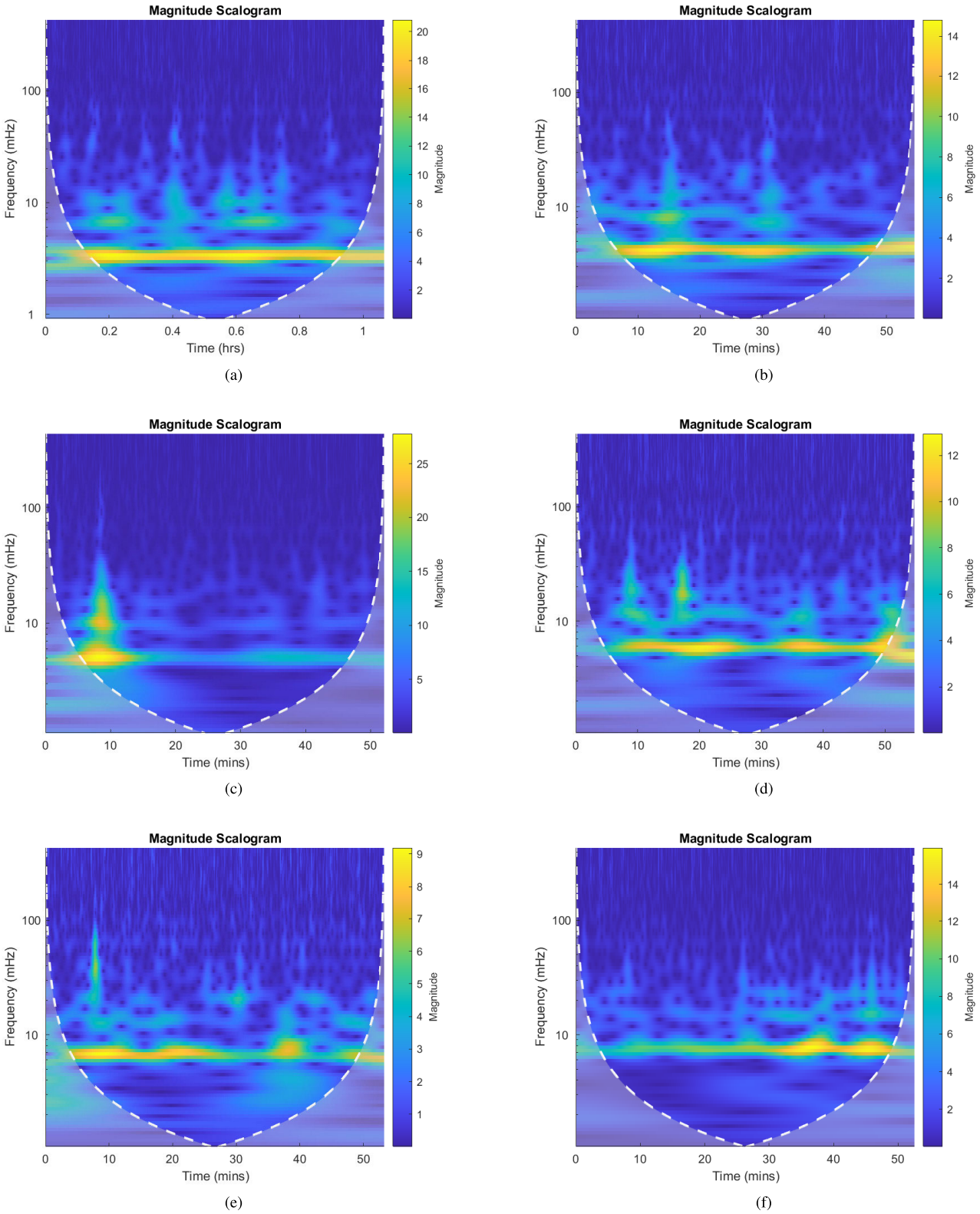

FIGURE 3. The CWT output image corresponding to class (a) 10 BPM, (b) 12.5 BPM, (c) 15 BPM, (d) 17.5 BPM, (e) 20 BPM, and (f) 22.5 BPM.

force applied [30]. This obtained resistance gives a voltage drop which contributes to a voltage divider circuit as shown in Fig. 1 to obtain a voltage output at Analog to Digital Converter (ADC).

TABLE 1. Dataset details.

dB	Position	No. text files	No. of Spectrograms
10	Sitting	21	21
	Sitting and Moving	21	21
	Standing	21	21
	Standing and Moving	21	21
	Walking	21	21
12.5	Sitting	21	21
	Sitting and Moving	21	21
	Standing	21	21
	Standing and Moving	21	21
	Walking	21	21
15	Sitting	21	21
	Sitting and Moving	21	21
	Standing	21	21
	Standing and Moving	21	21
	Walking	21	21
17.5	Sitting	21	21
	Sitting and Moving	21	21
	Standing	21	21
	Standing and Moving	21	21
	Walking	21	21
20	Sitting	21	21
	Sitting and Moving	21	21
	Standing	21	21
	Standing and Moving	21	21
	Walking	21	21
22.5	Sitting	21	21
	Sitting and Moving	21	21
	Standing	21	21
	Standing and Moving	21	21
	Walking	21	21

The dataset obtained from the sensor [30] consists of data from 21 people who wore the developed piezoresistive wearable around their chest using a strap. The subjects which are chosen have physical variations i.e., different ages, weights, and chest diameters. Further, 16 subjects were perfectly healthy and the rest had respiratory problems like asthma. The dataset comprises 6 different classes based on the breathing rate values 10, 12.5, 15, 17.5, 20, and 22.5 BPM as shown in Fig. 2. These BPM values are captured by asking these subjects to breathe over a minute and with a total of 30 minutes per subject for dataset collection in five different metabolic conditions such as sitting, standing, walking, standing and moving, and standing and walking. Moreover, a gap of one minute is given between two readings. The dataset comprises of 21 folders corresponding to 21 subjects with 30 files (6 BPM for five different conditions) of voltage values obtained across ADC, making a total of 630 files for the dataset.

B. DATA PRE-PROCESSING

1) DATA BALANCING

Each case within the dataset has different data points. To feed the dataset to the CNN model, the data need to be uniform hence data need to be balanced using pandas data frame. The data point in each case is approximately in the range of 3040-3500 causing variability, and variable row vectors cannot be fit into the CWT function and transformed. Hence, the introduction of constant row sizes is essential. The lowest data point range is obtained as 3042 data points and the remaining data is removed to avoid any edge case such as 'infinity' or 'NaN' which can further cause the loss function of the CNN model to show incorrect or faulty values. The data which is separated as text files and classified into various cases are assembled into tables with its 6 classes using pandas data frame.

2) CONTINUOUS WAVELET TRANSFORM

For applications in which the signal is extremely transient and changes its form a lot in a short period of time, wavelet transformations are more appropriate. In this type of transformation, a wavelet (a small part of the signal) slides across the entire signal by varying the amplitude and frequency of the wavelet. Finally, at each time step the wavelet and signal are then multiplied [31], [32]. The change in wavelet is obtained by varying the wavelet scale whose coefficient is the product of the multiplication of wavelet and signal. Due to this algorithm, wavelet transforms can extract local and temporal information together and also provide a range of wavelets to choose from to slide over.

The use of CWT and image input in the convolutional neural network is necessary as image signals give information about the signal at any time as the data frame becomes continuous instead of inputting raw discrete data points. Secondly, CWT converts the signal images to the frequency domain, which is ideal for this paper as signal transformation does not change the nature of the signal in the frequency domain but gets changed in the time domain, which will result in misclassification by the CNN. The CWT operation is computationally not heavy as it took nearly 7 minutes to convert 720 images into CWT transformed images on an Intel i7 10th generation processor and also reduces the size of the image due to data compression.

Figs. 2a, 2b, 2c, 2d, 2e, and 2f show the raw voltage signals corresponding to 10 BPM, 12.5 BPM, 15 BPM, 17.5 BPM, 20 BPM, and 22.5 BPM, respectively. Figs. 3a, 3b, 3c, 3d, 3e, and 3f show the CWT output corresponding to 10 BPM, 12.5 BPM, 15 BPM, 17.5 BPM, 20 BPM, and 22.5 BPM, respectively. The transformed image is obtained by passing the data points through the CWT function in MATLAB along with the variable 't' as an argument. Variable 't' was in the range of 0 to 60.84 s. The dataset is sampled at 50 Hz frequency with 3042 data points in each vector (filtered out). This will result in a time of $3042/50 = 60.84$ s per signal.

III. DEEP CONVOLUTIONAL NEURAL NETWORK

The CNN model works accurately with the signal images. The model is self-sufficient in doing feature extraction and learns on its own. This is quite flexible and has the ability to extract features from raw data without any human supervision eliminating the traditional image processing techniques [33].

A deep convolutional neural network takes input as an image and assigns weights to each feature in the image. Based on the weighted difference, features can be distinguished from one another. With the use of filters in the network spatial, and temporal features can be extracted. CNN

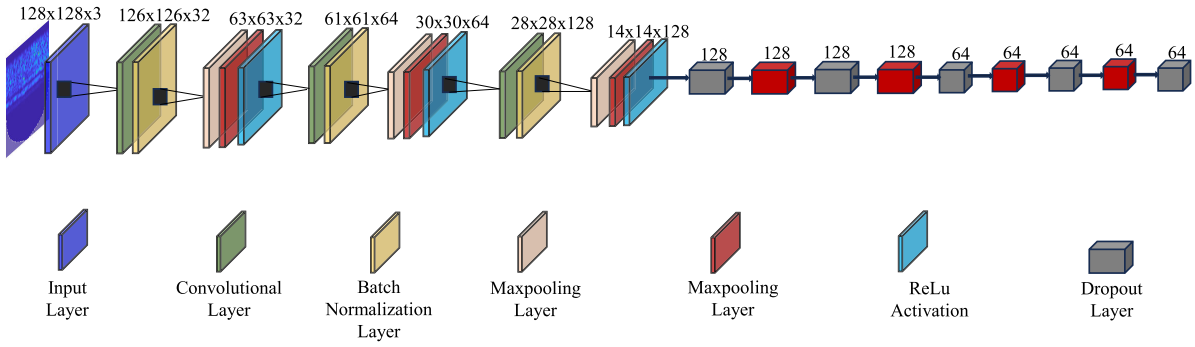

FIGURE 4. The architecture of the proposed CNN model.

TABLE 2. Summary of the number of parameters in each layer and its output shape for the proposed model.

Layer (type)	Number of Parameters	Output Shape
Input Layer	0	(None, 128, 128, 3)
conv2d (2D convolutional layer)	896	(None, 126, 126, 32)
batch_norm (Batch normalization layer)	128	(None, 126, 126, 32)
max_pooling2d (2D Max Pooling layer)	0	(None, 63, 63, 32)
dropout (Dropout Layer)	0	(None, 63, 63, 32)
re_lu (ReLU activation function)	0	(None, 63, 63, 32)
conv2d_1 (2D convolutional layer)	18496	(None, 61, 61, 64)
batch_norm_1 (Batch normalization layer)	256	(None, 61, 61, 64)
max_pooling2d_1 (2D Max Pooling layer)	0	(None, 30, 30, 64)
dropout_1 (Dropout layer)	0	(None,30, 30, 64)
re_lu_1 (ReLU activation function)	0	(None,30, 30, 64)
conv2d_2 (2D convolutional layer)	73856	(None, 28, 28, 128)
batch_norm_2 (Batch normalization layer)	512	(None, 28, 28, 128)
max_pooling2d_2 (2D Max Pooling layer)	0	(None, 14, 14, 128)
dropout_2 (Dropout layer)	0	(None,14, 14, 128)
re_lu_2(ReLU activation function)	0	(None,14, 14, 128)
flatten (Flatten layer)	0	(None, 25088)
dense (Dense layer)	3211392	(None, 128)
dropout_3 (Dropout layer)	0	(None,128)
dense_1 (Dense layer)	16512	(None, 128)
dropout_4 (Dropout layer)	0	(None,128)
dense_2 (Dense layer)	8256	(None, 64)
dropout_5 (Dropout layer)	0	(None,64)
dense_3 (Dense layer)	4160	(None, 64)
dropout_6 (Dropout layer)	0	(None,64)
dense_4 (Dense layer)	390	(None, 6)
Total parameters: 3,334,854		
Trainable parameters: 3.334.406		
Non-trainable parameters: 448		

architecture fits the image dataset as it reduces the parameters involved and reuses the weights.

Fig. 4 shows the architecture of the proposed 2D CNN model which is lightweight and consists of convolution layers followed by pooling layers with the ReLu activation function. These layers perform 3×3 convolutions and are later connected to the output layer via dropout and dense layers. The first layer is the input layer taking the input images of dimension $128 \times 128 \times 3$ followed by the convolution layer of filter size 3×3 and the batch normalization layer is added to stabilize the learning process as the inputs of the layer are normalized. The fourth layer is again the max pooling layer with size 2×2 and the fifth layer is the dropout layer with a dropout of 0.2 followed by the ReLu activation function and flatten layer. The next layers are a sequence of dense layers (ReLU activation) in combination with appropriate dropout layers to classify the image accurately. Finally, the output layer is used which consists of a dense layer with Softmax activation as it helps in proper identification in multi-class scenarios. The number of parameters and output shape of each layer is tabulated in Table 2.

1) CONVOLUTIONAL LAYER

The convolutional layer involves extracting the useful features from the image with the help of a kernel (a small matrix) passing over the image which transforms based on

the type of filter over it. Our model has two convolutional layers with a filter size of 3×3 [33].

2) BATCH NORMALIZATION LAYER

This layer acts as an input pre-processing layer before training the data over a neural network. Usually, intermediate layers result in a problem called the internal covariant shift. This will slow down the process as each layer learns to adapt after every epoch. In order to avoid such a problem, we normalize every layer reducing sensitivity and accelerating the training [34].

3) MAX POOLING LAYER

Max pooling is the type of pooling in which the maximum pixel value is returned by the kernel which rolls over the image. We chose max pooling because of its ability to suppress noise. Hence, three such layers are employed in the architecture. It can de-noise and also reduce the dimension altogether unlike average pooling which just reduces the dimension in order to suppress the noise [35].

4) DROPOUT LAYER

Training of bigger networks is difficult as they result in overfitting in most cases. But the dropout layer is used to work against this problem by putting a fraction of weights into training instead of all weights being put in a training step [36]. Seven dropout layers are used in our model till the correct classification of breathing rate.

5) RELU ACTIVATION LAYER

Activation functions are used to transform the input into neurons. It decreases the non-linear variables in the network to develop a clear relationship between the input and output of the network.

6) FLATTEN LAYER

This layer ensures the conversion of the input data in the layer to a 1-D array referred to as 'flattening' into a single feature vector. This layer connects to the dense layer (fully connected layer). The pixel data is put in one line and fed to a layer that fully connects to the final level [37].

7) DENSE LAYER

This layer ensures the connection between every neuron in the network [38]. It is always used at the end of the network for the classification. Every neuron of the previous layer is connected to every neuron of the dense layer. If the preceding layer gives the output as $M \times N$ matrix, the number of neurons in the dense layers should sum up to N. Five dense layers are added to our model architecture to classify the breathing rate with higher accuracy.

A. TRAINING OF CNN

The model training involves a batch size of 16 and is optimized using Adam (works well with sparse or noisy gradients and is adaptive) with a learning rate of 0.00001 and 'sparse-categorical cross entropy' over the Google Compute Engine Backend. As we have multiple classes (i.e., 6) sparse categorical cross entropy works and suits the best.

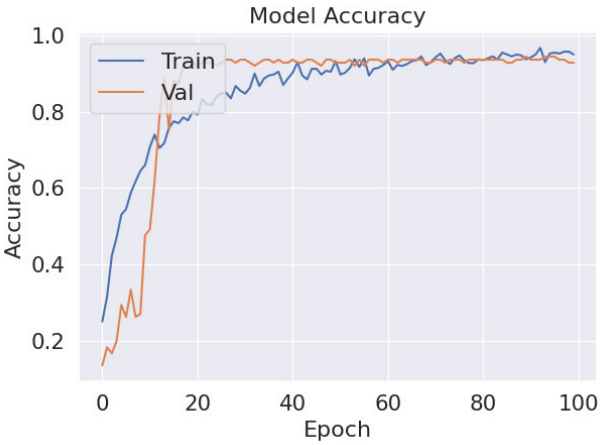

FIGURE 5. The training and validation accuracy of the proposed deep CNN model.

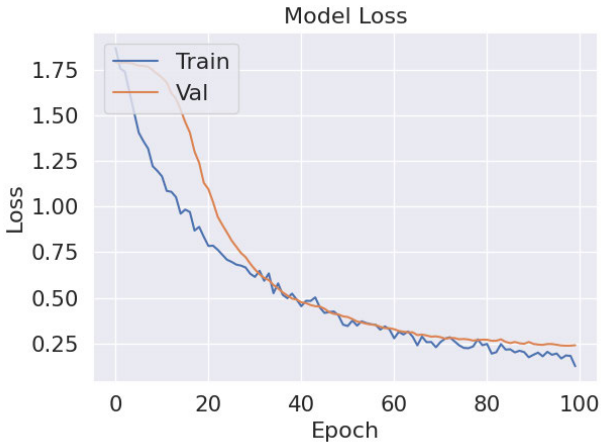

FIGURE 6. The training and validation loss of the proposed deep CNN model.

Among the available datasets, training, and testing were split by using the k-fold cross-validation method which is described as follows.

1) K-FOLD CROSS VALIDATION

When the data sample is limited then a cross-validation procedure is used to re-sample the available dataset and help the machine learning models to train themselves. The validation method has a parameter k which defines the number of data split groups or folds and also the number of ways the data is split into train and test samples [39]. The cross-validation method is employed to estimate how good the model is on unseen data. We have used the value of k as 5 based on trial and error to find the best accuracy and least overfit for that particular value.

TABLE 3. Variation of validation accuracy and number of parameters for all the models considered in this paper.

Model	Average Validation Accuracy (5-fold in %)	Accuracy(in %)					Total Parameters	Trainable Parameters	Non-trainable Parameters	Size
		Train 1	Train 2	Train 3	Train 4	Train 5				
Our Model	96.40	96.67	98	97.11	96.44	96	3,334,854	3,334,406	448	13.34 MB
Resnet152V2	82.382	84	83.11	80.44	79.56	77.56	58,360,326	5,539,846	52,820,480	226.8 MB
Resnet101V2	84.13	84	80.89	84.67	82.89	79.33	42,655,238	5,539,846	37,115,392	165 MB
EfficientnetV2B0	75.39	85.00	90.50	82.25	76.00	84.00	5,937,238	4,026,582	1,910,656	22.75 MB
EfficientB1V2	72.21	78.50	66.25	68.50	74.25	77.00	6,949,050	4,026,582	2,922,468	26.63 MB
EfficientB2V2	72.86	54.25	94.25	91.50	62.75	74.75	8,789,092	4,718,482	4,070,610	33.72 MB
Resnet50	37.46	81.25	93.25	93.00	86.75	87.25	23,616,390	14,470,662	9,145,728	92.5 MB
MobilenetV2	73.00	70.00	72.43	76.00	72.72	71.54	4,104,774	4,067,078	37,696	57.68 MB
Densenet201	47.15	43.00	38.25	30.25	35.75	53.97	20,826,694	6,080,966	14,745,728	80.13 MB
Resnet152	55.56	44.67	60.67	62.22	54.67	62.44	58,399,622	5,540,870	52,858,752	26 MB
VGG19	40.71	75.33	75.56	80.67	98.00	62.22	20,031,558	20,029,510	2,048	78.24 MB
Resnet101	31.75	70.75	74.25	69.25	65.75	69	42,686,854	5,540,870	37,145,984	15.6 MB

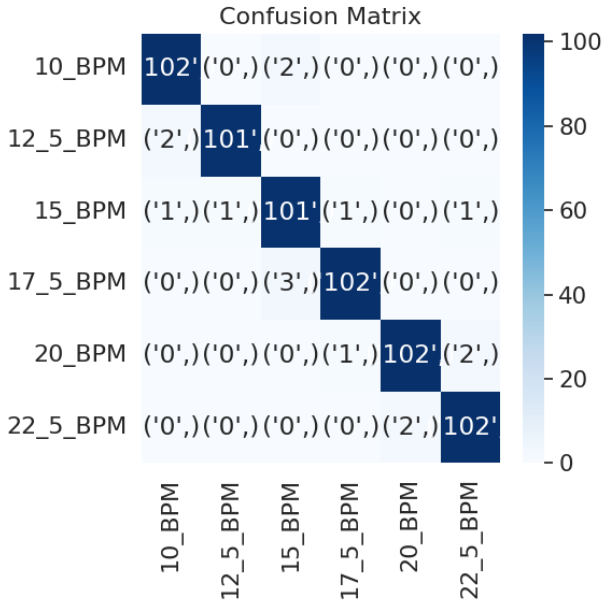

FIGURE 7. Confusion matrix of the proposed deep CNN model.

B. BENCHMARK MODELS

We compare the proposed CNN model with several benchmark models described below.

1) DENSENET

Dense Convolutional Network (DenseNet) [40], unlike convolutional networks, connects every layer to every other layer in the network by feed forwarding. DenseNets has worked on various problems presented by other benchmark models and has improved vanishing gradient problems, introduced feature reuse, and reduced the number of parameters. The model is pre-trained on Imagenet and its variants such as DenseNet201 are used in the paper for comparative study and analysis. DenseNet201 was modified with the addition of a global average pooling layer, two batch normalization layers, two dropouts, and three dense layers in this work. Further, its last 100 layers were kept as trainable layers and the rest were frozen to obtain trainable parameters in alignment with the number of trainable parameters of our model.

2) MOBILENETV2

MobileNetV2 is a type of convolutional neural network that has connections between the bottleneck layers and uses deep convolutions as filters for feature extraction. The architecture consists of a fully connected convolutional layer with 32 filters and 19 bottleneck layers [41].

3) RESNET

Residual networks or ResNet is an optimized model architecture that trains itself by learning on residual layers instead of non-residual ones also known as residual mapping. The residual blocks are stacked over each other depending on the number of layers stacked, and a variant of the ResNet is formed. For example, ResNet-50 has 50 such layers. This paper uses ResNet50, ResNet101, ResNet101V2, ResNet152, and ResNet152V2 for comparative study and analysis [42]. Resnet101, ResNet50, Resnet152, Resnet101V2, and Resnet152V2 are modified by adding one global average pooling layer, two batch normalization layers, two dropouts, and a dense layer while keeping its last 15 layers as trainable and 30 for ResNet50.

4) VGG19

VGG [43] is a 3 × 3 filter convolutional neural network architecture that is introduced to increase the depth of the existing neural network. The architecture is the same as the CNN except it uses a pooling and fully connected layer in addition. The VGG19 variant of the VGG model, pre-trained on the Imagenet dataset, is considered in this work. VGG19's last 50 layers are set as trainable and the rest of the architecture is frozen for a comparable parameter and accuracy study with the proposed model. It is modified by adding one global average pooling layer, two batch normalization layers, two dropouts, and a dense layer.

5) EFFICIENTNET

EfficientNet [44] is another convolutional neural network architecture that uses a fixed compound coefficient to scale all the dimensions of width, depth, and resolution uniformly. When the input image size increases the network

needs more depth, channels, and receptive fields to work on the bigger image which can be achieved by scaling them all together. In this work, we consider EfficientNetB0V2, EfficientNetB1V2, and EfficientNetB2V2 models pre-trained on the ImageNet dataset. All three models used in the paper were modified by adding one global average pooling layer, two batch normalization layers, two dropouts, and a dense layer while keeping its last 100 layers trainable.

IV. RESULTS AND DISCUSSION

The proposed deep convolutional neural network is trained on a publicly available dataset of the breathing rate of six classes in terms of 10, 12.5, 15, 17.5, 20, and 22.5 BPM. Figs. 5 and 6 show the validation accuracy and loss with increasing epochs (maximum 100). It is observed from Fig. 5 that the validation accuracy increases fast with respect to training accuracy and both settle down near 70 epochs. The validation reaches a maximum 97% and training reaches 96% over 100 epochs. From Fig 6, it is observed that loss is exponentially smooth and stabilizes after nearly 70 epochs, reaching a minimum value of around 0.2 over the 100th epoch.

Fig. 7 shows the confusion matrix of the proposed CNN model. From Fig. 7, it is observed that the proposed model classifies the 10 and 22.5 BPM with 98% accuracy, 12.5 with 97.14% accuracy, 15 with 96.19% accuracy, 17.5 with 95.23% accuracy and the least accuracy of 93.34% in 20 BPM. The confusion matrix highlights how accurate and efficient the proposed model is in classifying the breathing rate even with a small dataset, reaching an accuracy of 98%.

We also trained some of the pre-trained models such as ResNet, DenseNet, MoblieNet, VGG, and EfficientNet with the considered dataset. The proposed model achieves training and test accuracy in the range of 96%-98% and 92% to 98%, respectively. Table 3 shows the comparison of the proposed model with the considered pre-trained models in terms of accuracy, parameters, and size of the dataset. Further, the proposed model achieves an average validation accuracy of 96.4% whereas the pre-trained models achieve validation accuracy in the range of 31.75% and 84.13%. It can be observed from Table 3 that the proposed model is the most lightweight in terms of the number of parameters (nearly three million) and gives the highest accuracy compared to all the other models over the five folds.

Table 4 presents the inference time of all the models over six different hardware such as Raspberry Pi 4, Colab GPU and TPU, Intel IBRS, Nvidia Jetson Nano, and Nvidia AGX Xavier. Inference time defines the time taken by the proposed model to process the given input image and classify it when run on an edge computing device. It also defines the latency of the network, which is an important aspect to deal with when the model is used in real-time. The inference time is calculated using tf.lite.Interpreter() function by importing the time library. The time library computed the difference

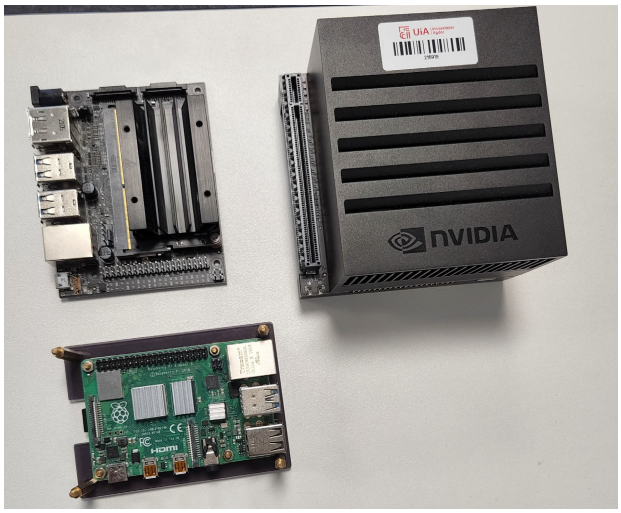

FIGURE 8. Edge computing devices used to calculate inference times for model performance. In Figure a) Nvidia Jetson Nano (top left), b) Raspberry Pi4 (bottom left), and c) Nvidia AGX Xavier (top right).

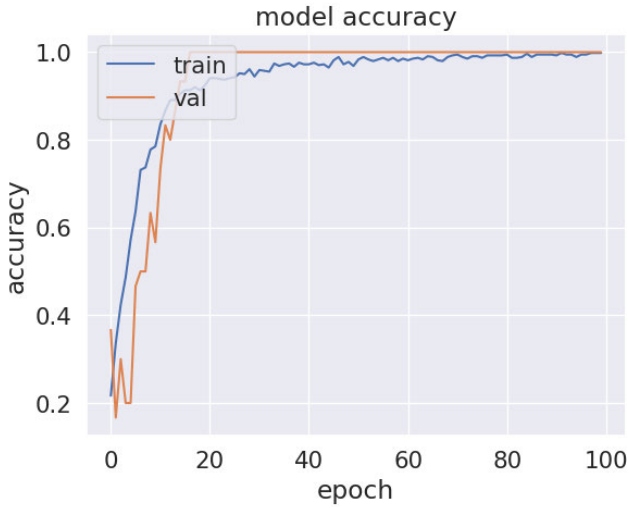

FIGURE 9. The training and validation accuracy of the proposed deep CNN model by holding the individual people out.

between the start and stop time of the model classification by using 300 randomly generated samples. Fig.8 shows the hardware modules corresponding to Raspberry Pi 4, Nvidia Jetson Nano, and Nvidia AGX Xavier. It is observed from Table 4 that the inference time of our model is much less compared to other models. The lowest inference time is on Intel IBRS of about 9.01 ms obtained by our model with the highest being 47.42 ms over Raspberry Pi4. Closest to our model, ResNet101V2 results in an inference time of around 11 ms but has its highest at 152 ms on Raspberry Pi 4. The highest inference time recorded is by VGG19 with 1722 ms on Jetson Nano which is over 78 times more than the inference time of our model over Jetson Nano. This clearly states that the proposed model is fast on the Piezoresistive dataset compared to other benchmark models.

TABLE 4. Inference time for all the models considered in this paper.

Models	Inference time(ms)					
	Raspberry Pi 4	Google Colab GPU	Google Colab TPU	Intel Core Processor IBRS	NVIDIA JETSON NANO	NVIDIA AGX XAVIER
Our Model	47.42	9.57	9.67	9.01	22.97	44.07
DenseNet201	466.52	100.69	101.65	103.89	448.05	201.67
EfficientNetV2B0	137.84	26.99	32.53	25.78	122.13	50.56
EfficientNetV2B1	181.6	34.90	39.41	32.77	157.87	68.4
EfficientNetV2B2	214.65	47.33	53.87	49.07	186.27	80.13
MobileNetV2	888.38	303.82	308.12	296.88	1325.25	766.05
ResNet152V2	1214.7	249.23	236.99	235.62	1012.16	591.78
ResNet152	1234.26	237.35	239.93	240.24	1032.18	606.94
ResNet101V2	152.1	28.94	32.47	11.47	117.09	40.33
Resnet101	824.29	162.15	165.18	163.07	713.54	411.87
Resnet50	427.54	89.84	90.64	85.98	369.59	211.24
VGG19	1159.27	387.04	381.54	375.02	1722.89	967.89

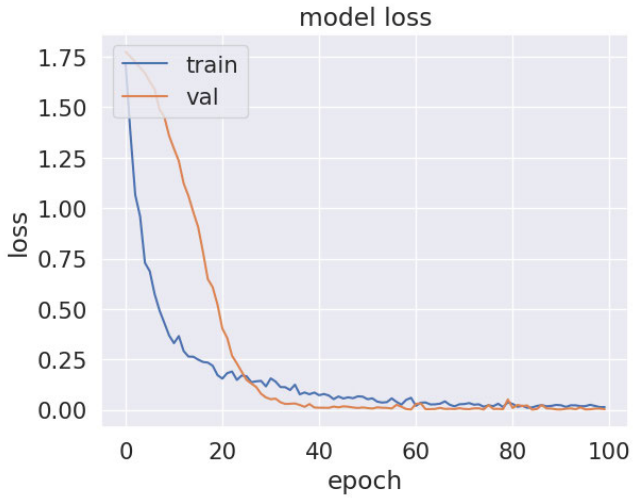

FIGURE 10. The training and validation loss of the proposed deep CNN model by holding the individual people out.

TABLE 5. Comparing Precision (P), Recall (R), and F1 scores for all models over the dataset.

Model	Precision(P)	Recall(R)	F1
Our Model	0.98	0.98	0.98
VGG19	0.92	0.89	0.89
Resnet101	0.46	0.84	0.59
Resnet50	0.35	0.64	0.45
Resnet101V2	0.84	0.84	0.84
Resnet152	0.83	0.74	0.78
Resnet152V2	0.90	0.89	0.89
MobilenetV2	0.46	0.47	0.36
Densenet201	0.60	0.47	0.48
EfficientnetV2B0	0.90	0.89	0.89
EfficientB1V2	0.85	0.83	0.83
EfficientB2V2	0.91	0.90	0.90

Table 5 shows the Precision, Recall, and F1 scores of all the models considered in this paper. It is observed from Table 5 that the proposed CNN model has the highest precision, recall, and F1 score indicating higher accuracy over the dataset.

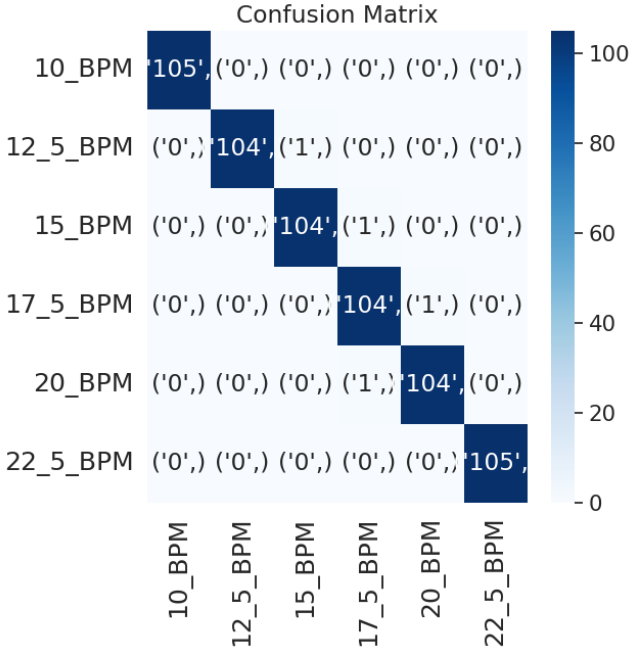

FIGURE 11. Confusion matrix of the proposed deep CNN model by holding the individual people out.

A. CROSS-VALIDATION RESULTS BY HOLDING THE INDIVIDUAL PEOPLE OUT OF THE CROSS-VALIDATION FOLDS

In this section, we trained the CNN model with 20 people's data and a single person's data kept for testing. Fig. 9 shows the variation of training and validation accuracy and Fig. 10 shows the variation of training and validation loss. It is observed from Fig. 9 that training and validation accuracy increase exponentially with the epoch and achieve a maximum accuracy of 96.67%. Further, it is observed that the training and validation loss is decaying exponentially. Fig. 11 shows the confusion matrix of the proposed model. It is observed from Fig. 11 that the proposed model correctly classifies all images corresponding to 10 and 22.5 BPM. However, it classifies 104 out of 105 images correctly

TABLE 6. Comparison of the proposed method with state-of-the-art.

Proposed method	Year	Dataset details	Measure	Value
SVM [14]	2018	Publicly available dataset of HR and BR signals	Accuracy	70%
GPR with Fitrgp feature selection algorithm [15]	2021	VORTAL dataset	RMSE	2.63 BPM
Contact-free breathing rate measurement [18]	2019	Video dataset from 10 people	Accuracy	84.66%
Nonlinear time series feature extraction and Bayes classifier [20]	2019	Continuous wave (CW) radar data with 3 subjects	Accuracy	97%
Bootstrap method [21]	2019	CW radar dataset	Average of absolute error	1.88 BPM
A video-based respiration and activity monitoring system [22]	2015	Video recordings from CMOS camera	Accuracy	88%
TR-BREATH [23]	2017	Channel state information from Wi-Fi signals	Accuracy	99%
CEEMDAN scheme with ML [24]	2020	Video dataset with 10 subjects	RMSE	2.30 BPM
A vector network analyzer-based continuous wave radar system [25]	2019	A dataset from 31 people with low/normal/high breathing rates	Accuracy	99.89%
Auto-regressive (AR) based technique [45]	2019	PPG signals	MAE	5.53
Empirical Mode Decomposition [46]	2021	the BIDMC and MIT-BIH Polysomnographic databases	Mean absolute percentage error	5%
Proposed CNN model	2023	Piezoresistive dataset	Accuracy	96.40%

corresponding to 12.5, 15, 17.5, and 20 BPM classes. Further, an image corresponding to 12.5, 15, 17.5, and 20 BPM are classified as 15, 17.5, 20, and 17.5 BPM, respectively.

V. CONCLUSION

In this paper, a lightweight convolutional neural network (CNN) model for breathing rate classification has been proposed using a publicly available dataset captured by a Piezoresistive sensor. The data has been pre-processed by arranging and balancing it so that it can be processed with the continuous wavelet transform. These CWT images were fed into a lightweight CNN model, which efficiently classified breathing rate into six classes based on breaths per minute. When compared to other pre-trained models, extensive results show that the proposed CNN model has achieved the highest accuracy with the smallest model size. When compared to other pre-trained models, it has also resulted in less inference time when run on all edge computing devices. The proposed model's accuracy has demonstrated its success in estimating breathing rate classification from a contact-based Piezoresistive sensor, and it can thus be integrated in conjunction with other sensors to obtain other vital signs. The major limitations of using a Piezoresistive Sensor are noise introduction due to any movement of the sensor, improper contact with the skin, temperature sensitivity, higher power consumption, and limited frequency range. In the future, we plan to integrate more than one sensor, such as a piezoresistive and an accelerometer to collect the data for high reliability. We also plan to look into the model that can process the multi-modal sensor data. In addition, we plan to collect a multi-sensor dataset of respiration rates for different tasks, performed by the person, and then propose a methodology to classify the tasks in addition to breathing rates.

REFERENCES

[1] E. Grooby, C. Sitaula, D. Fattahi, R. Sameni, K. Tan, L. Zhou, A. King, A. Ramanathan, A. Malhotra, G. A. Dumont, and F. Marzbanrad, "Real-time multi-level neonatal heart and lung sound quality assessment for telehealth applications," *IEEE Access*, vol. 10, pp. 10934–10948, 2022.

[2] A. Lazaro, M. Lazaro, R. Villarino, and D. Girbau, "Seat-occupancy detection system and breathing rate monitoring based on a low-cost mm-wave radar at 60 GHz," *IEEE Access*, vol. 9, pp. 115403–115414, 2021.

[3] A. Nicoló, C. Massaroni, E. Schena, and M. Sacchetti, "The importance of respiratory rate monitoring: From healthcare to sport and exercise," *Sensors*, vol. 20, no. 21, p. 6396, Nov. 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/21/6396

[4] P. C. Loughlin, F. Sebat, and J. G. Kellett, "Respiratory rate: The forgotten vital sign—Make it count!" *Joint Commission J. Quality Patient Saf.*, vol. 44, no. 8, pp. 494–499, Aug. 2018.

[5] D. O. Lemos and C. A. Siebra, "Methods for breathing rate measurement through mobile platform: A review," in *Proc. Anais do 19th Simp. Brasileiro de Comput. Aplicada á Saúde*, 2019, pp. 288–293.

[6] H. Liu, J. Allen, D. Zheng, and F. Chen, "Recent development of respiratory rate measurement technologies," *Physiological Meas.*, vol. 40, no. 7, Jul. 2019, Art. no. 07TR01.

[7] J. Kellett, M. Li, S. Rasool, G. C. Green, and A. Seely, "Comparison of the heart and breathing rate of acutely ill medical patients recorded by nursing staff with those measured over 5 min by a piezoelectric belt and ecg monitor at the time of admission to hospital," *Resuscitation*, vol. 82, no. 11, pp. 1381–1386, 2011.

[8] M. Folke, L. Cernerud, M. Ekström, and B. Hök, "Critical review of non-invasive respiratory monitoring in medical care," *Med. Biol. Eng. Comput.*, vol. 41, no. 4, pp. 377–383, Jul. 2003.

[9] F. Q. Al-Khalidi, R. Saatchi, D. Burke, H. Elphick, and S. Tan, "Respiration rate monitoring methods: A review," *Pediatric Pulmonol.*, vol. 46, no. 6, pp. 523–529, Jun. 2011.

[10] C. Massaroni, A. Nicolò, D. Lo Presti, M. Sacchetti, S. Silvestri, and E. Schena, "Contact-based methods for measuring respiratory rate," *Sensors*, vol. 19, no. 4, p. 908, Feb. 2019.

[11] P. H. Charlton, D. A. Birrenkott, T. Bonnici, M. A. F. Pimentel, A. E. W. Johnson, J. Alastruey, L. Tarassenko, P. J. Watkinson, R. Beale, and D. A. Clifton, "Breathing rate estimation from the electrocardiogram and photoplethysmogram: A review," *IEEE Rev. Biomed. Eng.*, vol. 11, pp. 2–20, 2018.

[12] P. H. Charlton, T. Bonnici, L. Tarassenko, D. A. Clifton, R. Beale, and P. J. Watkinson, "An assessment of algorithms to estimate respiratory rate from the electrocardiogram and photoplethysmogram," *Physiol. Meas.*, vol. 37, no. 4, p. 610, 2016.

[13] M. A. F. Pimentel, A. E. W. Johnson, P. H. Charlton, D. Birrenkott, P. J. Watkinson, L. Tarassenko, and D. A. Clifton, "Toward a robust estimation of respiratory rate from pulse oximeters," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 8, pp. 1914–1923, Aug. 2017.

[14] P. Napoletano and S. Rossi, "Combining heart and breathing rate for car driver stress recognition," in *Proc. IEEE 8th Int. Conf. Consum. Electron.*, Sep. 2018, pp. 1–5.

[15] M. N. I. Shuzan, M. H. Chowdhury, M. S. Hossain, M. E. H. Chowdhury, M. B. I. Reaz, M. M. Uddin, A. Khandakar, Z. B. Mahbub, and S. H. M. Ali, "A novel non-invasive estimation of respiration rate from motion corrupted photoplethysmograph signal using machine learning model," *IEEE Access*, vol. 9, pp. 96775–96790, 2021.

[16] V. Chamola, V. Hassija, V. Gupta, and M. Guizani, "A comprehensive review of the COVID-19 pandemic and the role of IoT, drones, AI, blockchain, and 5G in managing its impact," *IEEE Access*, vol. 8, pp. 90225–90265, 2020.

[17] A. Bella, R. Latif, A. Saddik, and L. Jamad, "Review and evaluation of heart rate monitoring based vital signs, a case study: COVID-19 pandemic," in *Proc. 6th IEEE Congr. Inf. Sci. Technol. (CiSt)*, Jun. 2020, pp. 79–83.

[18] G. O. Ganfure, "Using video stream for continuous monitoring of breathing rate for general setting," *Signal, Image Video Process.*, vol. 13, no. 7, pp. 1395–1403, Oct. 2019.

[19] M. Rehman, R. A. Shah, M. B. Khan, N. A. A. Ali, A. A. Alotaibi, T. Althobaiti, N. Ramzan, S. A. Shah, X. Yang, A. Alomainy, M. A. Imran, and Q. H. Abbasi, "Contactless small-scale movement monitoring system using software defined radio for early diagnosis of COVID-19," *IEEE Sensors J.*, vol. 21, no. 15, pp. 17180–17188, Aug. 2021.

[20] I. Nejadgholi, H. Sadreazami, S. Rajan, and M. Bolic, "Classification of Doppler radar reflections as preprocessing for breathing rate monitoring," *IET Signal Processing*, vol. 13, no. 1, pp. 21–28, 2019.

[21] I. Nejadgholi, H. Sadreazami, Z. Baird, S. Rajan, and M. Bolic, "Estimation of breathing rate with confidence interval using single-channel CW radar," *J. Healthcare Eng.*, vol. 2019, pp. 1–14, Mar. 2019.

[22] A. Heinrich, F. van Heesch, B. Puvvula, and M. Rocque, "Video based actigraphy and breathing monitoring from the bedside table of shared beds," *J. Ambient Intell. Humanized Comput.*, vol. 6, no. 1, pp. 107–120, Feb. 2015.

[23] C. Chen, Y. Han, Y. Chen, H.-Q. Lai, F. Zhang, B. Wang, and K. J. R. Liu, "TR-BREATH: Time-reversal breathing rate estimation and detection," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 3, pp. 489–501, Mar. 2018.

[24] M. Ghodratigohar, H. Ghanadian, and H. Al Osman, "A remote respiration rate measurement method for non-stationary subjects using CEEMDAN and machine learning," *IEEE Sensors J.*, vol. 20, no. 3, pp. 1400–1410, Feb. 2020.

[25] N. T. P. Van, L. Tang, A. Singh, N. D. Minh, S. C. Mukhopadhyay, and S. F. Hasan, "Self-identification respiratory disorder based on continuous wave radar sensor system," *IEEE Access*, vol. 7, pp. 40019–40026, 2019.

[26] K. M. I. Y. Arafath and A. Routray, "Automatic measurement of speech breathing rate," in *Proc. 27th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2019, pp. 1–5.

[27] Z. Xing, S. Zhao, W. Guo, F. Meng, X. Guo, S. Wang, and H. He, "Coal resources under carbon peak: Segmentation of massive laser point clouds for coal mining in underground dusty environments using integrated graph deep learning model," *Energy*, vol. 285, Dec. 2023, Art. no. 128771. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544223021655

[28] O.-F. Yolvi, V.-F. Luis Alex, V.-F. M. Angel, and R.-L. R. Alberto, "Artificial intelligence (AI) applied to public management," *Metaverse*, vol. 2, no. 1, p. 8, Jan. 2021.

[29] P. Bing, W. Liu, and Z. Zhang, "DeepCEDNet: An efficient deep convolutional encoder-decoder networks for ECG signal enhancement," *IEEE Access*, vol. 9, pp. 56699–56708, 2021.

[30] E. Vanegas, R. Igual, and I. Plaza, "Piezoresistive breathing sensing system with 3D printed wearable casing," *J. Sensors*, vol. 2019, pp. 1–19, Dec. 2019.

[31] L. Debnath and F. A. Shah, *Wavelet Transforms and Their Applications*. Berlin, Germany: Springer, 2002.

[32] N. Lopac, F. Hržic, I. P. Vuksanovic, and J. Lerga, "Detection of non-stationary GW signals in high noise from Cohen's class of time–frequency representations using deep learning," *IEEE Access*, vol. 10, pp. 2408–2428, 2022.

[33] T. Bezdan and N. Bacanin, "Convolutional neural network layers and architectures," in *Proc. Int. Sci. Conf. Inf. Technol. Data Related Res.*, 2019, pp. 445–451.

[34] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[35] J. Nagi, F. Ducatelle, G. A. Di Caro, D. Ciresan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. M. Gambardella, "Max-pooling convolutional neural networks for vision-based hand gesture recognition," in *Proc. IEEE Int. Conf. Signal Image Process. Appl. (ICSIPA)*, Nov. 2011, pp. 342–347.

[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 56, pp. 1929–1958, 2014.

[37] J. Jin, A. Dundar, and E. Culurciello, "Flattened convolutional neural networks for feedforward acceleration," 2014, *arXiv:1412.5474*.

[38] A. M. Javid, S. Das, M. Skoglund, and S. Chatterjee, "A ReLU dense layer to improve the performance of neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 2810–2814.

[39] Y. Wen, M. Kalander, C. Su, and L. Pan, "An ensemble noise-robust K-fold cross-validation selection method for noisy labels," 2021, *arXiv:2107.02347*.

[40] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.

[41] A. Howard, A. Zhmoginov, L.-C. Chen, M. Sandler, and M. Zhu, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," in *Proc. CVPR*, 2018, pp. 4510–4520.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, vol. 16, 2016, pp. 770–778.

[43] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[44] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[45] D. Jarchi, P. Charlton, M. Pimentel, A. Casson, L. Tarassenko, and D. A. Clifton, "Estimation of respiratory rate from motion contaminated photoplethysmography signals incorporating accelerometry," *Healthcare Technol. Lett.*, vol. 6, no. 1, pp. 19–26, Feb. 2019.

[46] A. Adami, R. Boostani, F. Marzbanrad, and P. H. Charlton, "A new framework to estimate breathing rate from electrocardiogram, photoplethysmogram, and blood pressure signals," *IEEE Access*, vol. 9, pp. 45832–45844, 2021.

KHUSHI GUPTA is currently pursuing the bachelor's degree in electrical and electronics (EEE) with the Birla Institute of Technology and Science (BITS), Pilani, Hyderabad Campus. She is a visiting student with ACPS Research Group, Department of Information and Communication Technology, Grimstad. Her research interests include machine learning, deep learning, the Internet of Things (IoT), computer architecture, digital image processing, computer vision, and embedded systems.

SREENIVASA REDDY YEDURI (Member, IEEE) received the bachelor's degree in electronics and communication engineering from Andhra University, Visakhapatnam, India, in 2013, the master's degree in computer science engineering (advanced networks) from the ABV-Indian Institute of Information Technology, Gwalior, India, in 2016, and the Ph.D. degree in electronics and communication engineering from the National Institute of Technology, Goa, India, in 2021. He has carried out his Ph.D. work with the Department of Electrical Engineering, Indian Institute of Technology Hyderabad, Hyderabad, India. He is currently a Postdoctoral Research Fellow with the Autonomous and Cyber-Physical Systems (ACPS) Research Group, Department of ICT, University of Agder, Grimstad, Norway. His research interests include AI/ML in communication/networking, cyber-physical systems, machine-type communications, the Internet of Things, LTE MAC, 5G MAC, optimization in communication, wireless networks, power line communications, visible light communications, hybrid communication systems, spectrum cartography, spectrum sensing, UAV-assisted wireless networks, wireless sensor networks, UAV path planning, and computer vision.

LINGA REDDY CENKERAMADDI (Senior Member, IEEE) received the master's degree in electrical engineering from Indian Institute of Technology Delhi (IIT Delhi), New Delhi, India, in 2004, and the Ph.D. degree in electrical engineering from the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, in 2011. He was with Texas Instruments on mixed-signal circuit design, before joining the Ph.D. Program with NTNU. After finishing the Ph.D. degree, he worked on radiation imaging for an atmosphere space interaction monitor (ASIM mission to the International Space Station) with the University of Bergen, Bergen, Norway, from 2010 to 2012. He is currently the Leader of the Autonomous and Cyber-Physical Systems (ACPS) Research Group and a Professor with the University of Agder, Grimstad, Norway. He has coauthored over 190 research publications that have been published in prestigious international journals and standard conferences in the research areas of the Internet of Things (IoT), cyber-physical systems, autonomous systems, robotics and automation involving advanced sensor systems, computer vision, thermal imaging, LiDAR imaging, radar imaging, wireless sensor networks, smart electronic systems, advanced machine learning techniques, and connected autonomous systems, including drones/unmanned aerial vehicles (UAVs), unmanned ground vehicles (UGVs), unmanned underwater systems (UUSs), 5G- (and beyond) enabled autonomous vehicles, and socio-technical systems like urban transportation systems, smart agriculture, and smart cities. He is also quite active in medical imaging. Several of his master's students won the best master thesis award in information and communication technology (ICT). He is also a member of ACM and the editorial boards of various international journals and the technical program committees of several IEEE conferences. He serves as a reviewer for several reputed international conferences and IEEE journals. He is the Principal Investigator and a Co-Principal Investigator of many research grants from Norwegian Research Council.