Cough Detection for Prevention Against the COVID-19 Pandemic

Btissam Bouzammour, Ghita Zaz, Malika Alami Marktani, Ali Ahaitouf, and Mohammed Jorio

Abstract This paper proposes a novel and robust technique for remote cough recognition for COVID-19 detection. This technique is based on sound and image analysis. The objective is to create a real-time system combining artificial intelligence (AI) algorithms, embedded systems, and network of sensors to detect COVID-19-specific cough and identify the person who coughed. Remote acquisition and analysis of sounds and images allow the system to perform both detection and classification of the detected cough using AI algorithms and image processing to identify the coughing person. This will give the ability to distinguish between a normal person and a person carrying the COVID-19 virus.

Keywords COVID-19 · AI algorithms · Cough detection · Embedded systems · Sensor fusion · Real-time processing

B. Bouzammour (✉) · G. Zaz · A. Ahaitouf · M. Jorio
SIGER Laboratory, Faculty of Sciences and Technology, Sidi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: btissam.bouzammour@usmba.ac.ma

G. Zaz
e-mail: ghita.zaz@usmba.ac.ma

A. Ahaitouf
e-mail: ali.ahaitouf@usmba.ac.ma

M. Jorio
e-mail: mohammed.jorio@usmba.ac.ma

M. Alami Marktani
SIGER Laboratory, National School of Applied Sciences, Sidi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: malika.alamimarktani@usmba.ac.ma

1   Introduction

The world has suffered greatly from the consequences of the coronavirus on our lives. The rapid spread of this pandemic continues to disrupt the balance of the world, making any attempt to limit the spread of the virus one of the most important priorities to be taken seriously.

Many research groups are trying to find a solution to this pandemic by using AI to recognize the cases of COVID-19 quickly. The main goal is to distinguish this virus from other similar pathologies by listening to a person's voice when coughing.

Imran et al. [1] conducted a preliminary study to detect COVID-19-related coughs collected with smartphone applications, where a combination of deep patterns was trained from 48 patients who tested positive. Chloe et al. [2] used Web-based applications to download the population's coughing sounds along with their demographic data and medical history to develop a machine learning algorithm based on voice, breath, and cough sounds. Gökcen et al. [3] develop an AI-based mobile application to COVID-19 by real-time cough measurement. A public data set was used, features (MFCC features, status, gender, respiratory condition, fever muscle pain, and status) were selected, and they applied deep learning algorithm for classification. The model provided an accuracy of 79%. Erdoğan et al. [4] propose a study to develop a system able to detect COVID-19(+) patients from the acoustic data of cough. Data has been selected from free access site. The feature extraction has been done by a traditional approach using the empirical mode decomposition and the discrete wavelet transform, and the feature selection was applied with the ReliefF algorithm. An accuracy of 97.8% was obtained. Tena et al. [5] have developed a model based on the automatic diagnosis of COVID-19 from automatic extraction of cough characteristics. Autoencoder was implemented for extraction features, and supervised machine learning algorithm was applied. The model provided an accuracy close to 90%.

This paper presents a novel technique to detect the presence of COVID-19 cough through smart technologies using visual and sound methods to be able to detect and identify any person carrying this virus. The rest of the paper is organized as follows. In Sect. 2, coughing detection audio estimation is described. Section 3 presents the coughing detection pose estimation technique. Section 4 gives experimental results of the proposed solution. Finally, the paper is concluded in Sect. 5.

2   Coughing Detection Audio Estimation

The proposed algorithm is illustrated in Fig. 1. The system consists of four main components: data segmentation, features extraction, classification, and data separation.
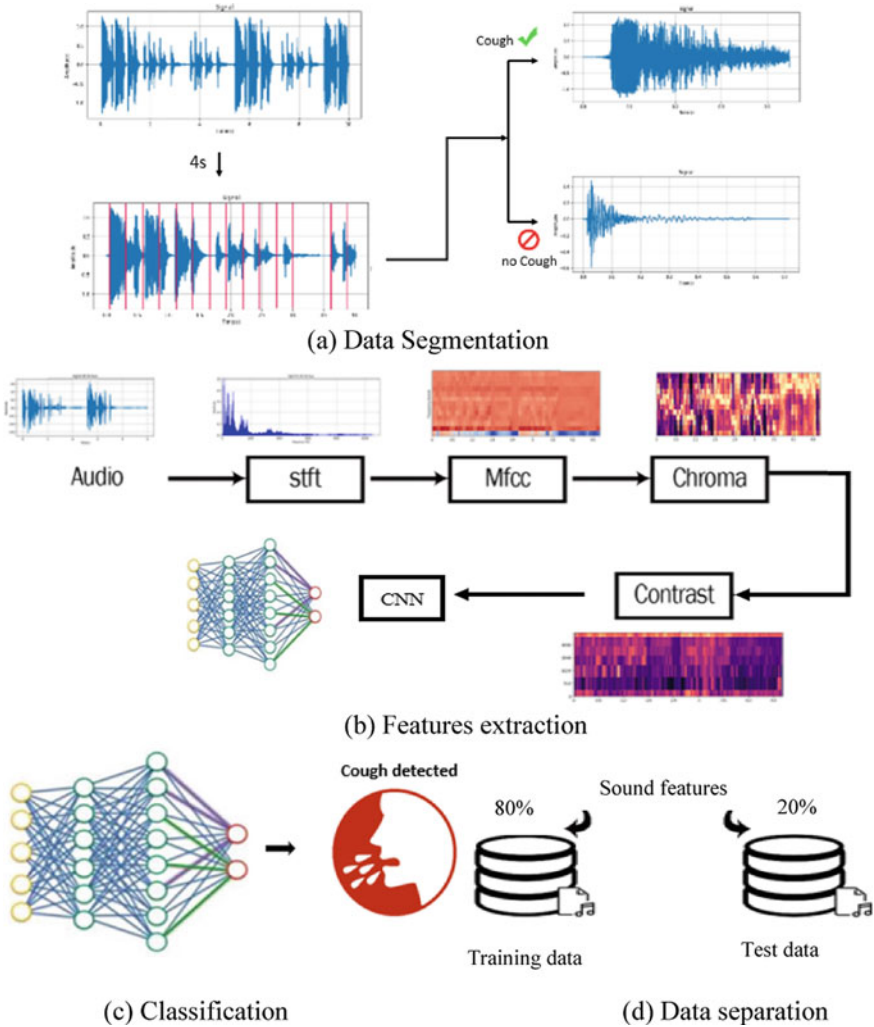
(a) Data Segmentation

(b) Features extraction

(c) Classification

(d) Data separation

Fig. 1 Cough sound detection process

2.1 Data Segmentation

The dataset is labeled as cough and no-cough with a duration of 4 s that was chosen experimentally. The cough class contains the sound of pure cough, and the no-cough includes any sound except cough (environment, clear noise, speech …).

2.2 Feature Extraction

Using the librosa python library, four features of the audio files were extracted. These features are Mel frequency cepstral coefficients (MFCC), Short-Time Fourier Transform (STFT), Chroma, and Contrast.

- **Mel frequency cepstral coefficients (MFCC)**: It is a widely used feature in automatic sound recognition. It is the result of the real short-term log-cosines transformation of the energy spectrum, expressed by the Mel frequency scale [6]. The original sound is pre-processed by a pre-emphasis filter and a bandpass filter. Then, the pre-processed signal is segmented into frames and a window is added.
- **Short-Time Fourier Transform (STFT)**: It is performed on each frame, and the spectrum is squared. Then, the result is filtered by a bank of filters Mel filters to obtain an energy adapted to human frequencies (Mel Energy). The logarithm of the Mel energy is used. Then, a Discrete Cosine Transform (DCT) is performed. Finally, the MFCC are obtained.
- **Chroma**: Chroma or chrominance vector is a 12-element feature vector indicating the amount of energy in each pitch class, which identifies the property that allows the classification of the sound on a frequency-related scale, and the energy of each frequency is represented by a color [7]. The blue color corresponds to a low amplitude, and the more vivid colors (such as red) corresponds to amplitudes progressively stronger.
- **Contrast**: The contrast characterizes the light distribution of an image [8]. Visually, it can be interpreted as a spread of the histogram of brightness of the image. A high contrast image has a good dynamic distribution of gray values over the entire range of possible values, with clear whites or deep blacks. On the contrary, a low contrast image has a low dynamic range, with most pixels having very close gray values.

2.3 Classification

Convolutional Neural Network (CNN) is a machine learning technique inspired by the structure of the brain. It comprises a network of learning units called neurons. These neurons learn to convert input signals (in our case the spectrogram image of the cough) into corresponding output signals (the label "cough"), forming the basis for automated recognition.

A CNN architecture is made of a succession of processing blocks allowing to extract the features that discriminate the image class from the others. A treatment block consists of [9, 10]:

- Convolution layer (CONV), which processes the data from a receiver field, it is used to extract the different characteristics of the input images.

- Activation layer (ReLU), it is a non-linear activation function, which replaces all negative values received as inputs by zeros.
- Pooling layer (POOL), it allows to compress the information by reducing the size of the intermediate image, to improve network efficiency and avoid overlearning.
- Grouping layer (Flatten), which allows the grouping of feature maps in vector columns.
- Fully connected layer (FC), which allows to classify the input image of the network. It returns a vector where each element indicates the probability for the input image belongs to a class.

2.4 Data Separation

The general principle of audio classification systems includes two stages [11]:

1. A learning stage which can be seen as a development phase leading to the implementation of a classification strategy.
2. A testing stage by which the performance of the classification system is evaluated.

In general, a system is ready for real use only after a succession of learning and testing steps that allow the implementation of an efficient classification strategy.

2.5 Performance Evaluation

After training the model, the results obtained are observed and the training characteristics are varied in order to increase the accuracy rate and decrease the error rate.

The model should perform as well on the training data as on the validation data. This is the ideal case; it means that the model is efficient and recognizes the images it knows as well as those it has never seen.

3 Coughing Detection Pose Estimation

To indicate the movements of a person with the camera, this paper proposes to use the "multi-person pose estimation" model that detects the main points in the human body, knowing that in general a person who coughs places his hand or his elbow facing his mouth [12]. The developed algorithm calculates two indexes indicating whether a person is coughing or not. All the details are given bellow.

Fig. 2 Presentation of the "multi-person pose estimation" model indices [12]
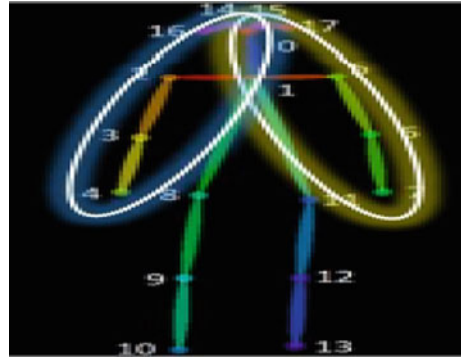
3.1 Multi-person Pose Estimation

The multi-person pose estimation model is a model that estimates the position of the 18 points (x, y) of $P0–P17$ in a 2D plane [13], where we can distinguish different points in the body such as elbows, knees, neck, shoulders, hips, and chest.

A person who coughs makes specific gestures and movements and then makes a coughing sound. The first reaction is to move the hand toward the mouth (right or left hand) and sometimes the whole arm toward the same sound outlet (the mouth) (Fig. 2).

We defined two main indices:

$$\text{Index\_R for the right side: } \frac{d_{5,6} + d_{6,7}}{2 * d_{0,7}} \tag{1}$$

$$\text{Index\_L for the left side: } \frac{d_{2,3} + d_{3,4}}{2 * d_{0,4}} \tag{2}$$

Each index uses the distances between different points. These are calculated from their coordinates (x, y), in order to propose an equation that allows us to define a cough threshold.

3.2 Threshold Validation

The idea was taken from a project on wink detection by Soukupova and Cech [14] who were able to validate the results by using the SVM model to have a threshold. The latter was also based on distances between specific points in the eye.

The same steps are used to validate the final index threshold in the next section.

To find a threshold that indicates the existence of cough, we first collect data to build a dataset, then apply a classification to obtain the threshold of detection of cough that we will integrate in our program, and finally make tests.

3.3 Dataset

To collect a real database, the first step was to use a video of a person coughing and another video where the person is not coughing, and then we store the values of the two indices (Index_R and Index_L), as shown in Figs. 3 and 4. For this purpose, the stored videos were cut in 60 frames per second in order to process each frame (Frame by Frame), as we aim to increase the accuracy of the results. The results are stored in an Excel sheet (.xlsx) to facilitate classification.

From Figs. 3 and 4, we can visualize the margin or the person coughs, and for both indices, the next step is to apply a classification algorithm to properly indicate and validate the chosen threshold, based on the Support Vector Machine.

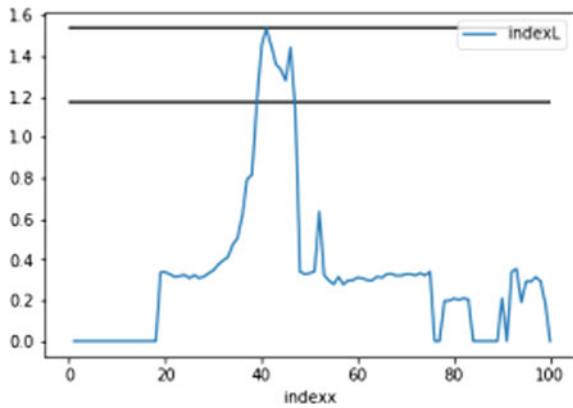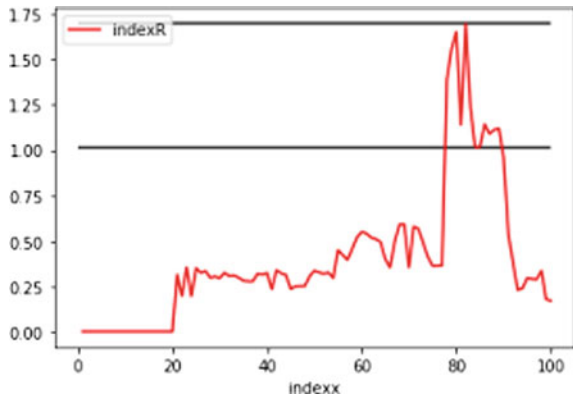**Fig. 3** Presentation of Index_L

Fig. 4 Presentation of Index_R

3.4 Support Vector Machine

Support Vector Machines (SVM) are a class of learning algorithms initially defined for discrimination. They have been then generalized to the prediction of a quantitative variable. In the case of discrimination of a dichotomous variable, they are based on the search of the optimal margin hyperplane which, when possible, correctly classifies or separates the data while being as far as possible from all the observations. The principle is therefore to find a classifier, or a discrimination function, with the highest possible generalization capacity (predictive quality) [15]. The choice of this model is motivated by non-negligible technical constraints, and SVM present in practice very good performances, it is able to provide good classification performances from a reduced number of learning examples while acting in very high dimensional spaces. We apply the SVM algorithm and repeat it until the final accuracy increases to 1.0, where the two index thresholds are: Index_L = 1.23 and Index_R = 1.23.

3.5 Performance Evaluation

To detect the person who coughs, the program must indicate the values of the two indexes (Left, Right):

- If Index_L ≥ 1.23 OR Index_R ≥ 1.23 → it's a coughing person.
- Else if Index_L < 1.23 and Index_R < 1.23 → it's a non-coughing person.

4   Results

4.1 Audio Detection

The proposed architecture consists of four convolution layers, followed by pooling layers. The activation function (ReLU) [10] is performed with the convolution and finally a fully neural network layer for classification. The accuracy of learning and validation increases with the number of epochs, i.e., the number of times an algorithm uses the dataset, this reflects that at each epoch the model learns more information. Similarly, the learning and validation error decreases with the number of epochs. There is no evidence of over-training or under-training, so the model has done the training well and can generalize on audio files that it has never seen.

   The result of our model is as follows:

- A "waiting for detection" message is displayed when the program is executed.
- A "cough detected" message is displayed when the cough is identified.
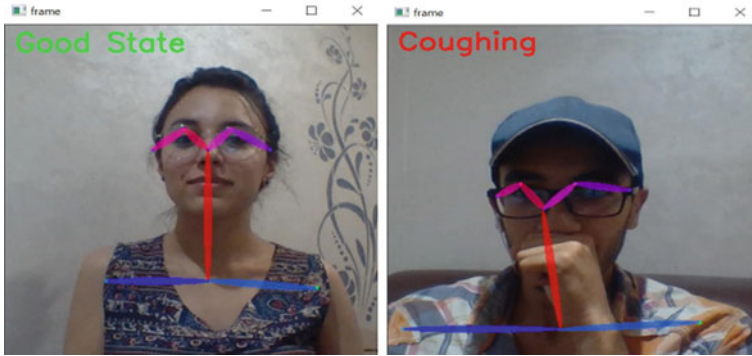
Fig. 5 Real-time results

This model has been developed using a TOSHIBA PROTÉGÉ laptop. It is characterized by: Windows 10 Professional x64, Intel(R) Core (TM) i7 CPU 2.70–2.90 GHz, 16 GO, and 237 Go SSD.

The model achieved an accuracy of 95.90% in one hour, 31 min, 34 s, with a learning error rate of 3.9472%.

4.2 Image Detection

To visualize the program results, the desktop camera is used. And the result is displayed in the video in real time, either Coughing or Good State as shown in Fig. 5.

The estimated time during the processing of a single image was between 0.3 and 0.5 s depending on the number of persons on the image. The estimated time during processing of each frame of the video was between 0.7 and 0.85 s depending on the condition of the person as well as the quality of PPI (Pixel Per Inch) and the resolution of the video (most of the time the tests are done by our computer camera).

5 Conclusion

This paper proposes an intelligent system capable of identifying one of the most common symptoms of COVID-19 (cough). The design of this system was carried out in several stages based on 2 main components: The first one allows to detect the sound of the cough, and the second one allows to locate the person who coughs. During this study, significant results are obtained. These results were presented and interpreted to show the effectiveness of the proposed methods. This progress gives the possibility to integrate this system in another more powerful system which includes

the detection of other symptoms of COVID such as body temperature and respiratory rate detection in order to give a more accurate diagnosis for carrying the COVID-19 virus.

References

1. Imran A, Posokhova I, Qureshi HN, Masood U, Riaz MS, Ali K, Nabeel M et al (2020) AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app. Inform Med Unlocked 20:100378
2. Brown C, Chauhan J, Grammenos A, Han J, Hasthanasombat A, Spathis D, Mascolo C et al (2020) Exploring automatic diagnosis of COVID-19 from crowdsourced respiratory sound data. arXiv preprint arXiv:2006.05919
3. Gökcen A, Karadağ B, Riva C, Boyacı A (2021) Artificial intelligence-based COVID-19 detection using cough records. Electrica 21(2):203–208
4. Erdoğan YE, Narin A (2021) COVID-19 detection with traditional and deep features on cough acoustic signals. Comput Biol Med 136:104765
5. Tena A, Clarià F, Solsona F (2022) Automated detection of COVID-19 cough. Biomed Signal Process Control 71:103175
6. McFee B, Raffel C, Liang D, Ellis D, McVicar M, Battenberg E, Nieto O (2015) Librosa: audio and music signal analysis in python. In: Presented at the python in science conference. Austin, Texas, pp 18–24. https://doi.org/10.25080/Majora-7b98e3ed-003
7. Srivastava A, Jain S, Miranda R, Patil S, Pandya S, Kotecha K (2021) Deep learning based respiratory sound analysis for detection of chronic obstructive pulmonary disease. PeerJ Comput Sci 7:e369. https://doi.org/10.7717/peerj-cs.369
8. Lee D, Lee J, Ko J, Yoon J, Ryu K, Nam Y (2019) Deep learning in MR image processing. Investig Magn Reson Imag 23:81. https://doi.org/10.13104/imri.2019.23.2.81
9. Ghimire A, Thapa S, Jha AK, Kumar A, Kumar A, Adhikari S (2020) AI and IoT solutions for tackling COVID-19 pandemic. In: Presented at the 4th international conference on electronics, communication and aerospace technology (ICECA). IEEE, Coimbatore, India, pp 1083–1092. https://doi.org/10.1109/ICECA49313.2020.9297454
10. OpenClassrooms (2021) Découvrez les différentes couches d'un CNN—Classez et segmentez des données visuelles. Last accessed 12 June 2021
11. Affonso C, Rossi ALD, Vieira FHA, de Leon Ferreira ACP (2017) Deep learning for biological image classification. Exp Syst Appl 85:114–122
12. Chen S, Demachi K (2020) A vision-based approach for ensuring proper use of personal protective equipment (ppe) in decommissioning of fukushima Daiichi nuclear power station. Appl Sci, 10(15), 5129
13. Faber M (2019) https://github.com/michalfaber/keras_Realtime_Multi-Person_Pose_Estimation.github. Last accessed 09 July 2021
14. Soukupova T, Cech J (2016) Eye blink detection using facial landmarks. In: 21st computer vision winter workshop. Rimske Toplice, Slovenia
15. Drugman T, Urbain J, Dutoit T (2011) Assessment of audio features for automatic cough detection. In: 19th European signal processing conference. IEEE, pp 1289–1293