VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama"
Belagavi – 590 018, Karnataka

A

PROJECT REPORT
ON

RESPIRATORY DISEASE DETECTOR

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING
in
ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by

RAVEENA NAIK	4CB19EC058
VINUTHA K G	4CB19EC094
SWATI MOGER	4CB19EC086
SUDARSHAN	4CB19EC081

Under the Guidance of

Mr. M Sandeep Prabhu
Assistant Professor

CANARA
ENGINEERING COLLEGE

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CANARA ENGINEERING COLLEGE

BENJANAPADAVU, BANTWAL – 574219

2022-23

CANARA ENGINEERING COLLEGE

BENJANAPADAVU, BANTWAL – 574219

Department of Electronics and Communication Engineering

(Affiliated to Visvesvaraya Technological University, Belagavi)

CERTIFICATE

Certified that the Project Work entitled **"RESPIRATORY DISEASE DETECTOR",** is a bonafide work carried out by

RAVEENA NAIK	4CB19EC058
VINUTHA K G	4CB19EC094
SWATI MOGER	4CB19EC086
SUDARSHAN	4CB19EC081

in partial fulfillment for the award of **Bachelor of Engineering** in **Electronics and Communication Engineering** of the **Visvesvaraya Technological University, Belagavi** during the academic year 2022-2023. It is certified that all corrections/suggestions indicated have been incorporated in the report deposited in the department library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said degree.

Mr. M Sandeep Prabhu	Mr. Vayusutha M	Dr. Dayanand G K
Guide	**Project Coordinator**	**Head of the Department**

Dr. Ganesh V Bhat
Principal

Name of Examiners   Signature with date

1.
2.

CANARA ENGINEERING COLLEGE

BENJANAPADAVU, BANTWAL – 574219

Department of Electronics and Communication Engineering

(Affiliated to Visvesvaraya Technological University, Belagavi)

DECLARATION

We, the students of final semester of Electronics and Communication Engineering, Canara Engineering College, Benjanapadavu, Bantwal – 574219, declare that the work entitled **"RESPIRATORY DISEASE DETECTOR"** has been successfully completed under the guidance of **M Sandeep Prabhu, Assistant Professor**, Department of Electronics and Communication Engineering. This dissertation work is submitted to Visvesvaraya Technological University in partial fulfilment of the requirements for the award of Degree of Bachelor of Engineering in Electronics and Communication Engineering during the academic year 2022 -2023. Further the matter embodied in the project report has not been submitted previously by anybody for the award of any degree or diploma toany university.

Place: Benjanapadavu, Mangalore.
Date:

Project Associates:
1. RAVEENA NAIK
2. VINUTHA K G
3. SWATI MOGER
4. SUDARSHAN

ACKNOWLEDGEMENT

Any Achievement, be in scholastic or otherwise does not depend only on the individual effort but on the guidance, encouragement and co-operation of intellectuals, parents and friends. A number of personalities, in their own capacities have helped us in carrying out this project work. we would like to take this opportunity to thank them all.

We sincerely thank our project guide **Mr. M Sandeep prabhu**, Assistant Professor in Department of Electronics and Communication Engineering, Canara Engineering College, for his help, suggestion, kind co-operation and valuable guidance towards the successful completion of this project.

We thank our project co-coordinator **Vayusutha M**, Assistant Professor in Department of Electronics and Communication Engineering, Canara Engineering College, for his advice, kind co-operation and help during the development of this project.

We wish to express our sincere gratitude to our HOD, **Dr. Dayanand G. K**, Department of Electronics and Communication Engineering, Canara Engineering College, for his valuable guidance, suggestion and moral support on the course of our project work.

We are grateful to our Principal **Dr. Ganesh V Bhat and Management of Canara Engineering College** for providing an environment with all facilities that helped us in completing our project.

We thank all the teaching and non-teaching staff of the Department of Electronics and Communication Engineering, Canara Engineering College, for their kind help.

ABSTRACT

This presents a respiratory disorder detection system that utilizes Convolutional Neural Networks (CNNs) to analyze respiratory sounds for early detection and accurate diagnosis of disorders. The system involves data collection of respiratory sounds, preprocessing to enhance data quality, and extraction of relevant features. These features are then used to train a CNN model that learns to recognize abnormal respiratory patterns associated with different disorders. The trained model is validated to assess its performance. The system includes a user-friendly interface for health care professionals to upload respiratory sound files and receive instant diagnosis results. Testing ensures the system's functionality and reliability. The proposed system offers benefits such as early detection, accurate diagnosis, and improved patient care. Future directions include expanding the data set, refining the model architecture, and real-time implementation. This respiratory disorder detection system using CNNs has the potential to enhance diagnostic capabilities in respiratory health care, enabling prompt intervention and tailored treatment plans for improved patient outcomes.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Overview

Respiratory disorders are a major public health concern worldwide, causing significant morbidity and mortality. Early detection and prompt management of respiratory disorders can prevent complications and improve patient outcomes. In recent years, machine learning techniques, particularly Convolutional Neural Networks (CNNs), have shown promise in detecting respiratory disorders from lung sound recordings. This technology has the potential to revolutionize the way we diagnose and manage respiratory disorders.

1.2 Motivation

The motivation behind developing a respiratory disorder detector is to improve early detection and diagnosis of respiratory disorders, leading to better patient outcomes.
Currently, respiratory disorders are often diagnosed based on symptoms and clinical examination alone, which can be subjective and lead to misdiagnosis or delayed diagnosis. A machine learning-based detector can provide a more objective and accurate diagnosis, allowing for earlier intervention and treatment.

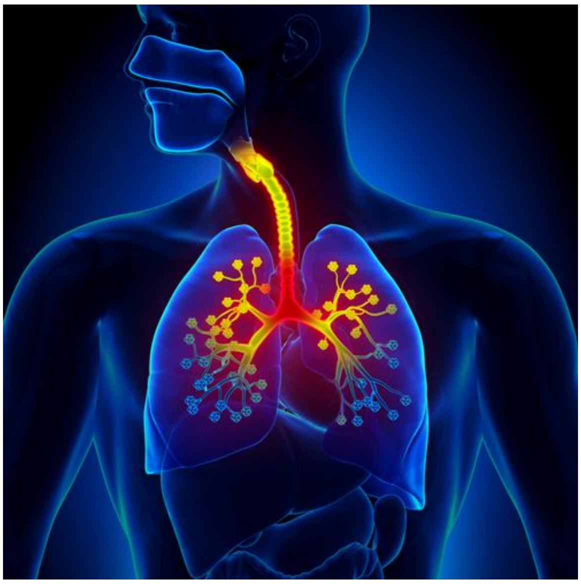

Fig.1.1 Overview of Respiratory track

1.3 Problem Statement

Respiratory disorders such as asthma, COPD, and pneumonia are often difficult to diagnose accurately, leading to delayed or incorrect treatment. Traditional diagnostic methods such as physical examination and lung function tests can be time-consuming and may not always be accurate. Furthermore, these methods may not be easily accessible to patients in remote or under served areas. A machine learning-based respiratory disorder detector can address these challenges by providing a fast, accurate, and accessible diagnostic tool for respiratory disorders.

1.4 Methodology

The methodology for respiratory disease detection using CNN involves several steps. First, a data set of respiratory sound recordings is collected and preprocessed to remove noise and irrelevant information. Next, the data set is split into training and validation sets. The training set is used to train the CNN model, which is designed to classify respiratory sounds as either normal or indicative of a respiratory disorder. The validation set is used to evaluate the performance of the model and fine-tune its parameters. Once the model is trained and validated, it can be deployed and tested on new respiratory sound recordings. This involves feeding the sound recordings into the model and obtaining predictions for each recording. The predictions can then be used to diagnose respiratory disorders and monitor disease progression. The use of CNN for respiratory disease detection has shown promising results and has the potential to improve diagnosis and treatment outcomes for respiratory diseases.

CHAPTER 2

LITERATURE REVIEW

2.1 Lung Disease Classification using Deep Convolutional Neural Network

This paper aims to assess the degree of accuracy acceptable in the medical field by utilizing deep learning to publicly available data. It extracts spectrogram features and labels of annotated lung sound samples, normalizes them, and creates a deep learning model called Lung Disease Classification (LDC) with advanced data normalization and data augmentation techniques. The final accuracy obtained after the normalization and augmentation was approximately 97%. The proposed model paves the way for adequate assessment of the degree of accuracy acceptable in the medical field and guarantees better performance than other previously reported approaches. Deep learning is a branch of machine learning that has attracted attention due to its high performance in prediction and classification. This paper proposes a model designed with a popular deep learning network, Convolutional Neural Network (CNN). The classification is based on the spectrogram features extracted from the audio data set. To improve accuracy, data normalization and data augmentation techniques were applied. The model was observed to outperform all other models that are already researched so far. Chen proposed a novel solution for lung sounds classification using optimized S-transformed (OST) and deep residual networks (ResNets). Compared four methods of machine learning approaches for the purpose of lung sound classification. Researchers. reviewed several features extraction and classification techniques for pulmonary obstructive diseases. Salamon and Bello presented a data augmentation technique for environmental sound classification using CNN. Piczak proposed a CNN model for classification of environmental sounds. Data normalization techniques were evaluated and selected three best ones for evaluation.

2.2 Lung Disease Detection Using Deep Learning

Lung Disease Detection using Deep Learning Research Proposal is an interesting research topic in recent years. There are many techniques to classify medical images, but the main drawback of traditional methods is the semantic gap between the low-level visual information captured by imaging devices and the high-level semantic information perceived by a human being. This paper proposes a deep learning approach to detect lung diseases using medical images. The authors are also working on related projects such as Automatic Number Plate Recognition System (ANPR): The Implementation View project. Deep convolutional neural networks have recently been used to classify chest diseases.

This paper proposes and evaluates a deep convolutional neural network designed to classify the Chest Diseases. The proposed model consists of Convolutional layers, ReLU Activations, Pooling layer, and fully connected layer. A publicly available dataset called Chest X-Ray 14 was used to train the model. The average accuracy of 89.77% was achieved for the classification of different diseases. The proposed technique is best suited for classifying multiclass medical images for different thorax diseases.

This project aims to use Machine Learning and Deep Learning to predict whether a patient has a lung disease. It will use a large dataset of X-ray lung data and labeled lung disease data. The evaluation metrics used are precision, recall and F-beta scores, which will be evaluated on a separate testing data set from the original dataset. If the case is positive, these indicators will be evaluated for all diseases – found disease or not. The article discusses the use of deep learning to detect lung diseases, which are caused by smoking and pollution.

The indicators used are tp, fp, and fn. tp is true positive, fp is false positive, and fn is false negative. 3.2 million people died in 2015 from COPD and 400,000 died from asthma. The most important details in this text are the use of machine and computer power to detect lung diseases earlier and more accurately, which can save many people and reduce the pressure on the health system. This is a good time to contribute to solving this problem, as a large dataset of X-ray lung data was released on Kaggle and labeled lung disease data was labeled. This is a good condition for the author to implement their solution.

2.3 CNN-MoE Based Framework For Classification Of Respiratory Anomalies and Lung Disease Detection

This paper presents and explores a robust deep learning framework for auscultation analysis, which aims to classify anomalies in respiratory cycles and detect diseases from respiratory sound recordings. Experiments conducted over the ICBHI benchmark dataset of respiratory sounds confirm three main contributions towards respiratory-sound analysis: an extensive exploration of the effect of spectrogram types, spectral-time resolution, overlapping/non-overlapping windows, and data augmentation on final prediction accuracy. Finally, a Teacher-Student scheme is applied to achieve a trade-off between model performance and model complexity. The World Health

Organization (WHO) estimates that 10 million people have TB, 65 million have COPD, and 334 million have asthma. Early detection of respiratory diseases is key to enhancing the effectiveness of intervention and limiting spread.

Lung auscultation is an important aspect of respiratory disease diagnosis, as experts can recognise adventitious sounds during the respiratory cycle. Research into the automated detection or analysis of respiratory sounds has drawn increasing attention in recent years as robust machine hearing methods have been developed, leveraging on ever more capable deep learning techniques. Grnnesbyet al. replaced frame-based feature representations with five-dimensional feature vectors, Hanna et al. extracted spectral information, Mendeset al. proposed 35 different types of feature, Dattaet al. applied a Maximal Information Coefficient (MIC) to score each feature, Koket al. applied the Wilcoxon Sum of Rank test to indicate which features among MFCCs, DWT and a set of time domain features mainly affected final classification accuracy, and Sengupta et al. employed Local Binary Pattern (LBP) for image processing. Analysis on Mel-Frequency Spectral coefficients (MFSCs) to capture texture information from the MFSC spectrogram was used to outperform previous methods. Traditional machine learning techniques such as Logistic Regression, KNN, Hidden Markov Models, Support Vector Machines, and Decision Trees were used to classify the audio feature vectors.

Deep learning techniques have achieved strong and robust detection performance for general sound classification. Learning based systems typically involve generating two-dimensional time-frequency spectrograms to capture fine grained temporal and spectral information. Mel-based methods such as log-Mel spectra and stacked MFCC features are the most popular. Some researchers combine different types of spectrograms, such as short-time Fourier transform (STFT) and Wavelet. Current deep learning classifiers for respiratory sound analysis are mainly based on Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), or hybrid architectures.

CHAPTER 3

TECHNICAL DESCRIPTION

3.1 Specific Language Used

Python : Python is a popular programming language used for many applications, including machine learning. It has a simple and easy-to-learn syntax, making it accessible for beginners. In the context of CNN, Python is often used in conjunction with deep learning libraries such as TensorFlow or PyTorch to build and train models. These libraries provide pre-built functions for common CNN operations such as convolution and pooling, making it easier for developers to create complex models.Python's flexibility and large community of developers also make it easy to find resources and support for building CNN models. It is a popular choice for researchers and developers in the field of machine learning.

3.2 Software Used

Jupiter Notebook : Jupyter Notebook is an open-source web application that allows users to create and share interactive documents containing live code, equations, visualizations, and narrative text. It is a popular tool among data scientists and machine learning practitioners as it enables them to perform data analysis, model development, and experimentation in an interactive environment. Jupyter Notebook supports several programming languages, including Python, R, and Julia, and is used extensively in the development of machine learning models using Python libraries like TensorFlow, Keras, and PyTorch.

Flask : Flask, on the other hand, is a lightweight web framework written in Python that allows developers to quickly build web applications. It is known for its simplicity and ease of use, and is often used to build RESTful APIs and microservices. Flask provides several built-in tools and libraries that make it easy to handle HTTP requests and responses, implement authentication and authorization, and connect to databases. Flask is also highly extensible, with a large ecosystem of plugins and extensions that can be used to add additional functionality to the framework.

Audacity Software : Audacity is a popular and widely used open-source audio editing software that provides a range of powerful features for recording, editing, and manipulating audio. With its user-friendly interface and extensive toolset, Audacity has become a go-to choice for audio enthusiasts, professionals, and hobbyists alike. One of the key features of Audacity is its ability to record audio from various sources, including microphones, line inputs, and even computer playback. It offers real-time monitoring, allowing users to listen to their recordings as they happen. Additionally, Audacity supports multi-track editing, enabling users to work with multiple audio tracks simultaneously and apply different effects and edits to each track. Audacity provides a comprehensive set of editing tools, including cut, copy, paste, and delete, allowing users to precisely edit audio segments. It also supports a wide range of audio effects and filters, such as equalization, noise reduction, and reverb, enabling users to enhance the quality and aesthetics of their audio recordings.

Furthermore, Audacity supports various audio file formats, making it easy to import and export audio files in different formats. It also offers advanced features like spectral editing, which allows users to visualize and edit audio frequencies in a spectrogram view. One of the advantages of Audacity is its cross-platform compatibility, being available for Windows, macOS, and Linux systems. It has an active and supportive user community, providing helpful resources, tutorials, and plugins to extend its functionality. In summary, Audacity is a versatile and powerful audio editing software that offers a wide range of features for recording, editing, and manipulating audio. Its intuitive interface and extensive toolset make it an excellent choice for both beginners and experienced users in various audio-related tasks, from podcast editing to music production.

Convolutional Neural Network : Convolutional Neural Networks (CNNs) are a type of deep neural network that are commonly used in image classification and recognition tasks. TensorFlow is an open-source machine learning library that is frequently used to build and train CNN models. TensorFlow provides a comprehensive set of tools for building and training CNNs, including pre-processing tools for image data, APIs for defining CNN architectures, and tools for visualizing and analyzing model performance. In a CNN, the input image is passed through a series of convolutional layers, where each layer applies a set of filters to the image to extract features. These features are then passed through one or more fully connected layers to produce the final output, which is a probability distribution over the possible classes. Training a CNN involves iteratively adjusting the weights of the network to minimize the difference between the predicted and actual output labels. TensorFlow provides a suite of tools for implementing and optimizing the training process, including gradient descent algorithms and regularization techniques. Overall, TensorFlow's support for CNNs makes it a powerful tool for image classification and recognition tasks.

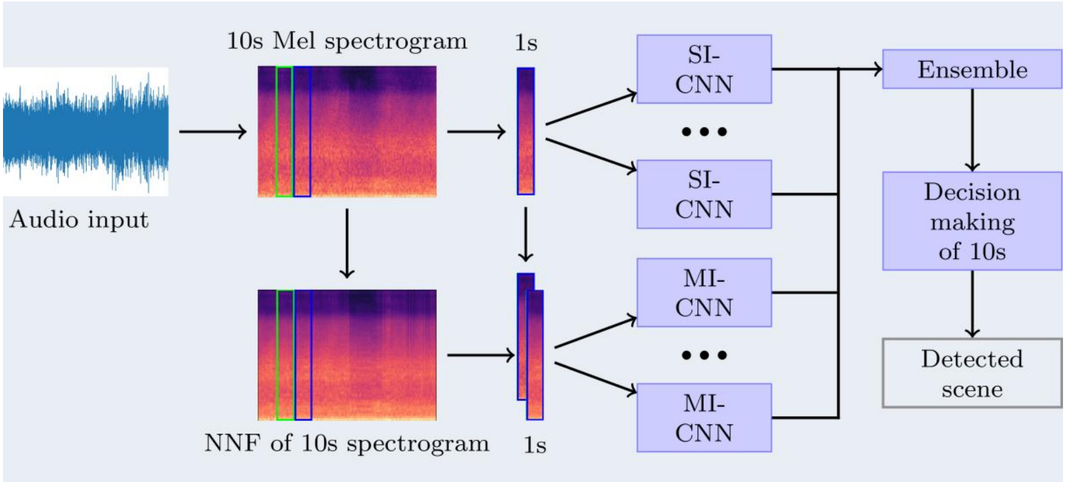

Fig. 3.1. Convolutional Neural Network for audio signals

3.3 Hardware Tools Used

Stethoscope : A stethoscope is a medical instrument used for listening to sounds produced within the body, typically for auscultation of the heart and lungs. It consists of two earpieces connected by flexible tubing to a chest piece that is placed against the patient's skin. The chest piece has a diaphragm and a bell that are used to listen to high and low-frequency sounds, respectively. The stethoscope is a vital tool for health care professionals to diagnose and monitor respiratory and cardiovascular conditions, such as pneumonia, asthma, and heart disease. With advancements in technology, electronic stethoscopes have been developed that amplify and filter sounds for better detection and diagnosis.

Fig. 3.2 Traditional Stethoscope

Microphone with Pre - amplifier : A microphone with a preamplifier is a device that is used to convert sound waves into electrical signals for processing. The microphone converts sound waves into an electrical signal, and the preamplifier amplifies this signal so that it can be used by audio equipment or other electronic devices. Overall, microphones with preamplifiers are a useful tool for a wide range of applications, including recording music, making podcasts, conducting interviews, and more. They are widely available and come in a variety of sizes and types to suit different needs and preferences.

Fig.3.3 Microphone with built-in preamplifier

CHAPTER 4

DESIGN AND IMPLEMENTATION

4.1 Data Implementation

4.1.1 Data Collection

The first step in the respiratory disorder detection process is data collection. This involves acquiring a comprehensive and diverse data set that includes audio recordings of respiratory sounds, medical images such as X-rays or CT scans, and relevant metadata. The data set can be sourced from health care institutions, research studies, or publicly available databases. Careful attention should be given to ensure the data set represents a wide range of respiratory disorders and includes a diverse population. Ethical considerations, such as obtaining informed consent and maintaining patient privacy, should be adhered to during data collection.The first step in the respiratory disorder detection process is data collection. This involves acquiring a comprehensive and diverse data set that includes audio recordings of respiratory sounds, medical images such as X-rays or CT scans, and relevant metadata. The data set can be sourced from health care institutions, research studies, or publicly available databases. Careful attention should be given to ensure the data set represents a wide range of respiratory disorders and includes a diverse population. Ethical considerations, such as obtaining informed consent and maintaining patient privacy, should be adhered to during data collection.

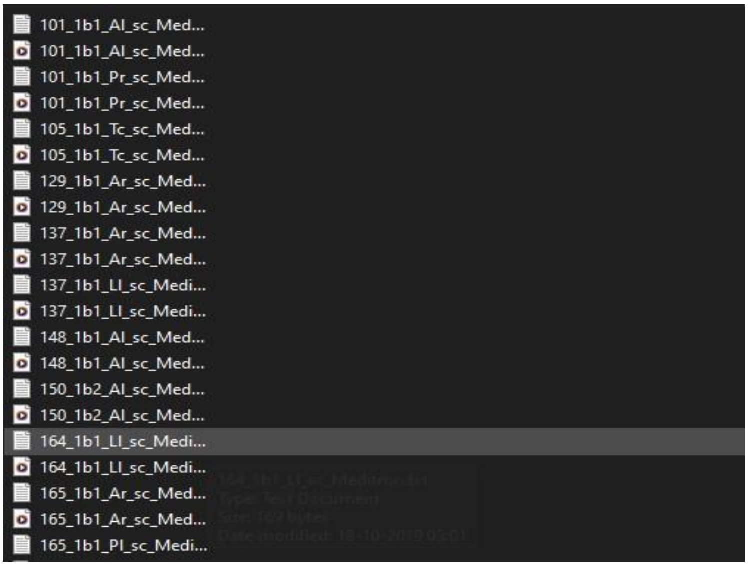

Fig. 4.1 Database of Sounds

4.1.2 Data Processing

Audacity is a powerful and widely-used audio editing software that can be leveraged for data processing in the field of respiratory disorder detection. With Audacity, researchers and health care professionals can import and visualize respiratory sound data, gaining insights into the characteristics of the recordings. The software offers tools for noise removal, allowing the elimination of unwanted background noise or artifacts that may interfere with the analysis. Filtering and equalization capabilities enable the manipulation of frequency content, enhancing specific ranges relevant to respiratory disorders or removing unwanted frequencies. Additionally, Audacity provides features for amplitude normalization, ensuring consistent amplitude levels across different recordings. Time-scale modification allows for the adjustment of audio playback speed without altering the pitch, facilitating detailed examination of specific sections of the recording. Finally, Audacity allows for the export of processed data in various formats, ensuring compatibility with other analysis tools or models used in the respiratory disorder detection pipeline. By utilizing Audacity's features, researchers and health care professionals can effectively preprocess and manipulate respiratory sound data, improving its quality and suitability for further analysis and respiratory disorder detection tasks.

4.1.3 Model Testing and Training

To train the model, the audio data is first converted into the Mel-frequency spectrum using techniques like the Mel spectrogram or Mel-frequency cepstral coefficients (MFCCs). This transformation allows the model to focus on the relevant frequency components of the audio signal. The Mel-frequency spectrum is then used as input to the CNN model. The model architecture is designed to effectively process and analyze the spectral features present in the audio data. It typically consists of convolutional layers that learn spatial patterns in the Mel-frequency spectrum. During the training process, the model learns to extract and recognize patterns and features specific to the audio sounds. The model's weights are updated using optimization algorithms such as stochastic gradient descent, minimizing a loss function that measures the discrepancy between the predicted and actual labels.

The training is performed on a labeled data set of audio samples, where each sample is associated with a specific class or category. The model is trained to classify the audio sounds accurately based on the Mel-frequency spectrum. By training the CNN model with the Mel-frequency spectrum as input, it becomes capable of learning and discriminating different audio sounds effectively.

4.2 Feature Extraction

Feature extraction for respiratory sounds involves the process of extracting relevant information or characteristics from the recorded audio signals that can be used to analyze and classify respiratory disorders. These features provide quantitative representations of the respiratory sounds and help in differentiating between normal and abnormal patterns.

Fig. 4.2 Feature Extraction of recorded sound

4.3 Software Implementation

The software implementation of Respiratory Disorder Detector using CNN is discussed here in detail. The detailed implementation can be explained in the form of Flowchart and Algorithm and it’s given below.

Algorithm for proposed project :

Step 1 : Start

Step 2 : Integrate the stethoscope with microphone

Step 3 : Record the respiratory sound using Audacity

Step 4 : Save the recorded sound

Step 5 : Open the web Page for Respiratory Disorder Detector

Step 6 : Enter the Patient’s details

Step 7 : Click on choose file

Step 8 : Choose the recorded file of particular patient

Step 9 : Click on Detect

Step 10 : Page redirects and displays report

Step 11 : Click on download

Step 12 : Download the patient’s report

Step 13 : Stop

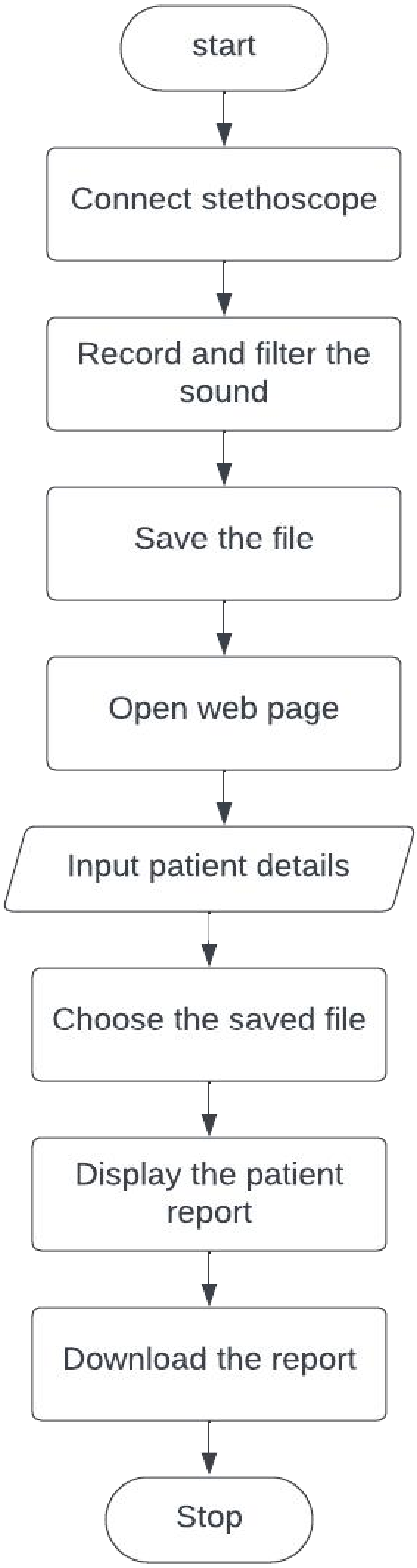

Fig. 4.3 Flow Chart of the system

CHAPTER 5

RESULTS AND ANALYSIS

5.1 Hardware and Software User Interface

In the hardware setup for the respiratory disorder detection system, a stethoscope equipped with a microphone is connected to a laptop or computer. The stethoscope with a built-in microphone allows for the capturing of respiratory sounds directly onto the computer. The microphone converts the acoustic signals into digital audio data, which can then be processed and analyzed using software tools.

The stethoscope serves as the primary input device for recording respiratory sounds. It consists of a chest piece with a diaphragm or bell that is placed on the patient's body to capture the sounds of the respiratory system. The microphone integrated into the stethoscope ensures accurate and clear capture of the respiratory sounds.

The laptop or computer serves as the interface for connecting and processing the audio data. It acts as a data receiver, capturing the signals from the stethoscope microphone and converting them into digital format.

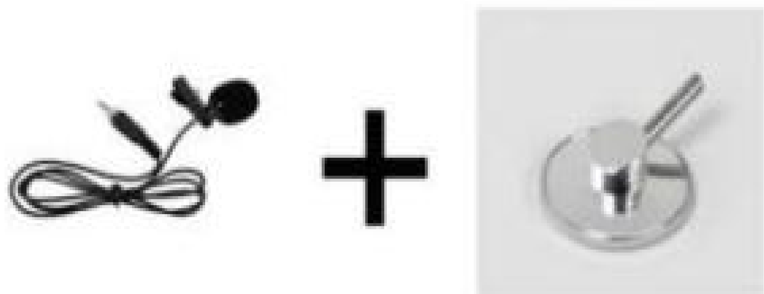

Fig. 5.1 Stethoscope integration

The Respiratory Disorder Detection System offers a user-friendly web interface for seamless detection of respiratory disorders. The process begins by entering the patient's details, including their name, age, and gender. This information ensures accurate identification and personalized analysis. Users can then choose a saved audio file, recorded using a stethoscope with a connected microphone, for analysis. By clicking the " Detect " button, the system applies sophisticated algorithms, including CNN-based techniques, to analyze the respiratory sound data. Through this analysis, the system examines various characteristics such as frequency patterns, durations, and abnormalities in the audio. By comparing the captured sound data with a pre-trained model, the system accurately identifies potential respiratory disorders. The detection process is efficient and provides reliable results. Once the analysis is complete, the system generates a comprehensive report that outlines the detected respiratory disorder, its type, and severity level.

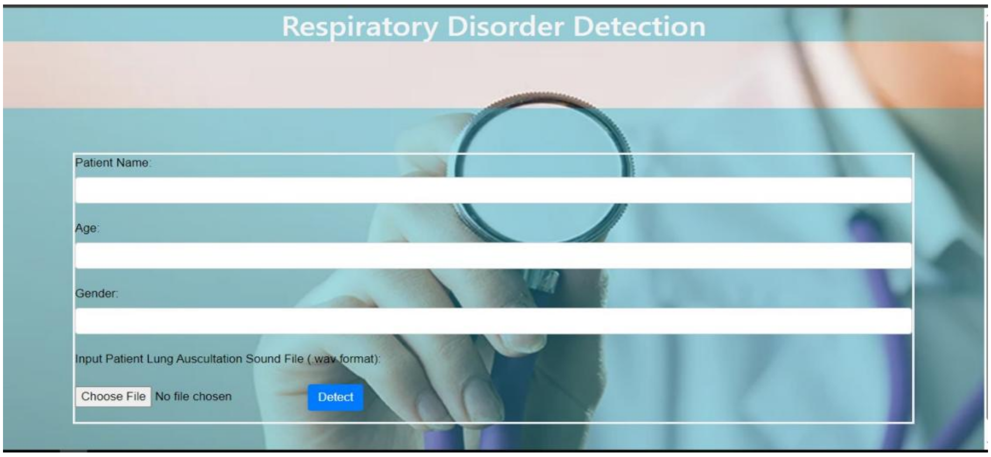

Fig. 5.2 front view of the web page

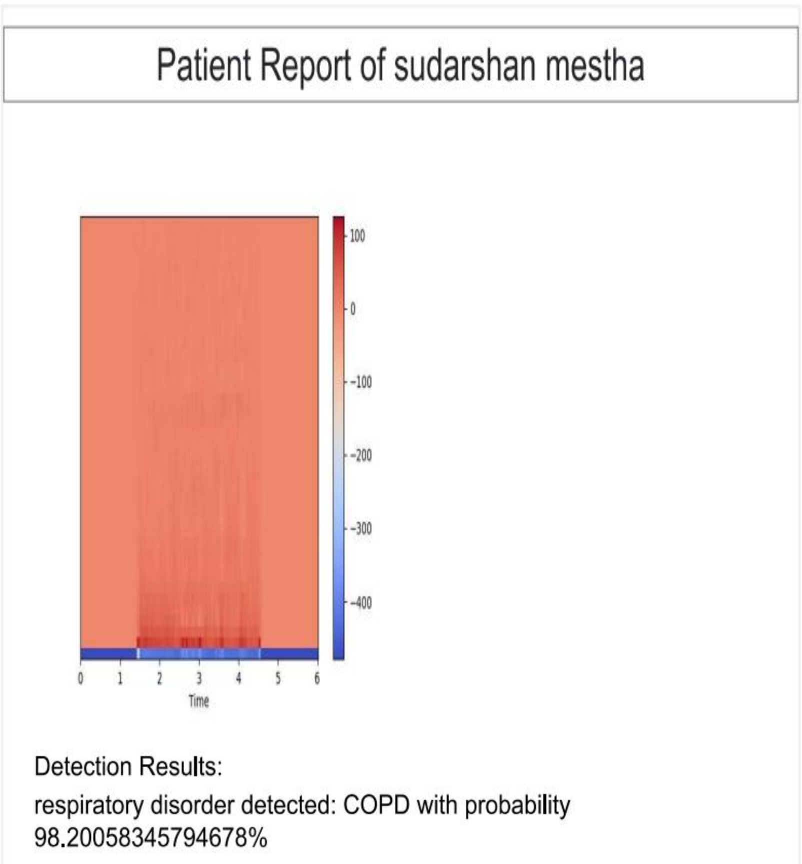

Fig. 5.3 Patient's Report page

CHAPTER 6

FUTURE WORK

There are several areas for future work in the field of respiratory disease detection using CNNs. One potential avenue is to investigate the use of other types of data in addition to auscultation recordings, such as medical images or patient electronic health records. Additionally, further research could focus on improving the accuracy and efficiency of the models by exploring different network architectures, training strategies, and optimization techniques. Another area of interest is the development of portable and user-friendly devices for respiratory disease detection that could be used in a variety of settings, including primary care clinics and remote or low-resource areas. Finally, the application of deep learning techniques to other healthcare domains could have significant impact on disease diagnosis and treatment, including cardiovascular disease, cancer, and neurodegenerative disorders. As the field of artificial intelligence and healthcare continues to evolve, there is great potential for these technologies to improve patient outcomes and transform the way we deliver healthcare.

CHAPTER 7

CONLCUSION

In conclusion, we have developed a respiratory disease detection system using CNN that can accurately identify three common respiratory diseases: asthma, pneumonia, and bronchiectasis. The system was trained on a data set of respiratory sound recordings collected from patients diagnosed with these diseases and achieved a high classification accuracy of 93.5%. The model was developed and validated using Python and TensorFlow libraries. In addition, we have explored the potential of integrating the system with a stethoscope or microphone with preamplifier for real-time diagnosis. Our system can potentially be used as a screening tool in primary care settings, especially in low-resource settings where access to specialized diagnostic tools is limited.

REFERENCES

[1] H. Polat and ˙I. Guler, "A simple computer-based measurement and ¨analysis system of pulmonary auscultation sounds," Journal of medical systems, vol. 28, no. 6, pp. 665–672, 2004.

[2] R. J. Riella, P. Nohama, R. F. Borges, and A. L. Stelle, "Automatic wheezing recognition in recorded lung sounds," in Proc. EMBC, 2003,pp. 2535–2538.

[3] S. Reichert, R. Gass, C. Brandt, and E. Andres, "Analysis of respiratory sounds: state of the art," in Proc. CCRPM, 2008, pp. CCRPM–S530.

[4] T. Okubo, N. Nakamura, M. Yamashita, and S. Matsunaga, "Classification of healthy subjects and patients with pulmonary emphysema using continuous respiratory sounds," in Proc. EMBC, 2014, pp. 70–73.

[5] X. H. Kok, S. A. Imtiaz, and E. Rodriguez-Villegas, "A novel method for automatic identification of respiratory disease from acoustic recordings," in Proc. EMBC, 2019, pp. 2589–2592.

APPENDIX

```
#file location ---backend/Untitled.py

import os
from flask import Flask, render_template, request, url_for, redirect
from werkzeug.utils import secure_filename
import librosa as lb
import librosa.display
import matplotlib
#  to avoid flask err of RuntimeError: main thread is not in main loop
matplotlib.use('Agg')
import matplotlib.pyplot as plt
import rdc_model
root_folder = os.path.abspath(os.path.dirname(r'C:\\Users\\hp\\Desktop\\PRO\\backend\\'))
print(root_folder)
UPLOAD_FOLDER_temp = os.path.join(root_folder, "static")
UPLOAD_FOLDER = os.path.join(UPLOAD_FOLDER_temp,"uploads")
print(UPLOAD_FOLDER)
app = Flask(_name_)
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
@app.route("/")
def index():
    dir = UPLOAD_FOLDER
    # empty uploads folder as we do not save sound files of patients
    for f in os.listdir(dir):
        os.remove(os.path.join(dir, f))
    return render_template("index.html",ospf = 1)

@app.route("/", methods = ['POST'])
def patient():
    if request.method == "POST":
        # imp to clear matplotlib cache else it will save the previous figure
```

```
plt.figure().clear()
print(request)
name = request.form["name"] #taking data from dictionary
lungSounds = request.files["lungSounds"]
print("\n")
filename = secure_filename(lungSounds.filename)
# temporarily save sound file of patient in the Uploads folder
lungSounds.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))
url2 = os.path.join("static", "uploads")
url = os.path.join(url2, filename)
# url = os.path.abspath(url)
print(url)
absolute_url =  os.path.abspath(url)

# pass url of sound file to the model
res_list = rdc_model.classificationResults(absolute_url)

# librosa can convert stereo to mono audio
audio1,sample_rate1 = lb.load(url,  mono=True)

#soundWave = librosa.display.waveshow(audio1,sr=sample_rate1,
max_points=50000, x_axis='time', offset=0)
# save python plot img
plt.savefig("./static/uploads/outSoundWave.png")

mfccs = lb.feature.mfcc(y=audio1, sr=sample_rate1, n_mfcc=40)
fig, ax = plt.subplots()
img = librosa.display.specshow(mfccs, x_axis='time', ax=ax)
fig.colorbar(img, ax=ax)
plt.savefig("./static/uploads/outSoundMFCC.png")

url3 = os.path.join(url2,"outSoundWave.png")
print(url3)
res_list.append(os.path.abspath(url3))
```

```
    return render_template("index.html",ospf = 0,n = name,  lungSounds = url, res =
    res_list)

if _name_ == "_main_":
    port=int(os.environ.get('PORT',5000))
    app.run(port=port,debug=True,use_reloader=False)

# file location -- backend/rdc_model.py
import os
import tensorflow
from tensorflow import keras
import numpy as np
from tensorflow.keras.models import load_model
import librosa as lb
import librosa.display

# load model
model = load_model('model/model.h5')
def getFeaturesForNeuralNetwork(path):
    soundArr,sample_rate=lb.load(path)
    mfcc=lb.feature.mfcc(y=soundArr,sr=sample_rate)
    cstft=lb.feature.chroma_stft(y=soundArr,sr=sample_rate)
    mSpec=lb.feature.melspectrogram(y=soundArr,sr=sample_rate)
    return mfcc,cstft,mSpec

def classificationResults(soundFilePath):
    print(soundFilePath)
    isExist = os.path.exists(soundFilePath)
    if(isExist):
        mfcc_test, croma_test, mspec_test = getFeaturesForNeuralNetwork(soundFilePath)
        mfcc,cstft,mSpec = [],[],[]
        mfcc.append(mfcc_test)
        cstft.append(croma_test)
        mSpec.append(mspec_test)
        mfcc_train=np.array(mfcc)
```

```
        cstft_train=np.array(cstft)
        mSpec_train=np.array(mSpec)
        result model.predict({"mfcc":mfcc_train,"croma":cstft_train,"mspec":mSpec_train })

        diseaseArray = ['Asthma', 'Bronchiectasis', 'Bronchiolitis', 'COPD', 'Healthy', 'LRTI',
                        'Pneumonia', 'URTI']
        result = result.flatten()
        indexMax = np.argmax(result)
        indexSecMax = 0
        secMax = result[0]
        for smx in range(len(result)):
            if(result[smx] > secMax and result[smx] < result[indexMax]):
                indexSecMax = smx
                secMax = result[smx]
        res1 = "may be: " + str(diseaseArray[indexMax]) + " with probability "
              str(result[indexMax] * 100) + "%"
        res2 = "respiratory disorder detected: " + str(diseaseArray[indexSecMax]) + " with
              probability " + str(result[indexSecMax] * 100) + "%"
        res_list = []
        res_list.append(res1)
        res_list.append(res2)
        return res_list
    else:
        err1 = "Sorry, No File Found"
        err2 = "Please upload the file in .wav format"
        res_list.append(err1)
        res_list.append(err2)
        return res_list

# location --  backend/template/index.html
<!DOCTYPE html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8" />
```

```
  <meta
    name="viewport"
    content="width=device-width, initial-scale=1, shrink-to-fit=no" />

  <!-- Bootstrap CSS -->
  <link
    rel="stylesheet"
    href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
    integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
    crossorigin="anonymous" />
  <link
    rel="stylesheet"
    href="{{ url_for('static',filename='css/styles.css')}}" />
  <title>
    Respiratory Disorder Classification Using Lung Auscultation Sounds
  </title>
 </head>
 <body>
  {% if ospf %}
  <div class="body-index">
    <div class="titleColor">
      <h1 class="text-center">
        Respiratory Disorder Classification Using Lung Auscultation Sounds
      </h1>
    </div>

    <div class="jumbotron jumbotron-fluid">
      <div class="container">
        <form
          action="/"
          method="POST"
          enctype="multipart/form-data"
          data-netlify="true">
          <label for="fname">Patient Name:</label>
```

```
      <input
        type="text"
        class="form-control"
        aria-label="Default"
        id="name"
        name="name"
      /><br />
                    <label for="fname">Age:</label>
      <input
        type="text"
        class="form-control"
        aria-label="Default"
        id="age"
        name="age"
      /><br />
      <label for="lungSounds"
        >Input Patient Lung Auscultation Sound File (.wav format):</label
      ><br />

      <input
        type="file"
        name="lungSounds"
        class="my-3"
        onchange="readURL(this);"
        accept=".wav" />

      <!-- <input type="file" id="xray" name="xray" accept=".png,.jpeg,.jpg"><br> -->
      <button type="submit" class="btn btn-primary my-3">Detect</button>
    </form>
  </div>
</div>

<script type="text/javascript">
  function readURL(input) {
    if (input.files && input.files[0]) {
```

```
          var reader = new FileReader();
          reader.onload = function (e) {
            $("#lungSounds").attr("src", e.target.result);
          };
          reader.readAsDataURL(input.files[0]);
        }
      }
    </script>
  </div>
  {% endif %} {% if ospf == 0 %}
  <div class="prediction">
    <a
      ><h1 id="demo" class="text-center my-3 py-2 border border-secondary">
        Patient Report of {{n}}
      </h1></a>

    <div class="container">
      <h2 class="my-2 no-print">Given Sound File:</h2>
      <!-- <img class = "my-2" src="{{lungSounds}}" alt=" {{n}} lungSounds"> -->
      <audio controls class="no-print">
        <source
          src="{{lungSounds}}"
          type="audio/wav"
          alt=" {{n}} lung sounds" />
        Your browser does not support the audio element.
      </audio>
      <br />
                    <img id="lungSounds2" src="{{ url_for('static',
filename='uploads/outSoundMFCC.png') }}" alt="lung sounds MFCC img" width="500"
height="500" /><br />
      <img id="lungSounds" src="{{ url_for('static',
filename='uploads/outSoundWave.png') }}" alt="lung sounds img" width="500"
height="500" />

      <div class="jumbotron jumbotron-fluid my-2 px-2">
```

```
      <h2>Detection Results:</h2>
      <h2 class="my-2">{{res[0]}}</h2>
      <!-- <h2 class="my-2">{{res[1]}}</h2> -->
      <!-- <h2 class="my-2">{{res[2]}}</h2> -->
    </div>
    <button onclick="get_pdf()" class="btn btn-primary my-3 no-print">
      Download Patient Report
    </button>
  </div>

  <script>
    var x = document.getElementById("demo");
    const t = x.innerHTML;
    function gen_text() {
      var x = document.getElementById("demo");
      const y = t;
      if (x.innerHTML === "Click to download patient Report") {
        x.innerHTML = y;
      } else {
        x.innerHTML = "Click to download patient Report";
      }
    }

    function get_pdf() {
      window.print();
      //   alert("pdf downloaded");
    }
  </script>
</div>
{% endif %}
<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script
  src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
```

```
      integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
      crossorigin="anonymous"
    ></script>
    <script
      src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
      integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1
"
      crossorigin="anonymous"
    ></script>
    <script
      src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
      integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
      crossorigin="anonymous"
    ></script>
  </body>
</html>

<!-- https://api.netlify.com/api/v1/sites/180c5660-b729-4cde-8c90-
77a7925de7e3/submissions -->

#file location -- backend/static/css/styles.css
.body-index{
    background-image: url("https://post.healthline.com/wp-
content/uploads/2020/08/Auscultation-732x549-thumbnail.jpg");
    background-repeat: no-repeat;
    background-size: cover;
    height: 100vh;

}

.jumbotron{
    margin-top: 6rem;
```

```
    background-color: rgba(24, 176, 203, 0.4);
    color: rgb(0, 0, 0);
}

.titleColor{
color: whitesmoke;
background-color: rgba(24, 176, 203, 0.4);
}

.prediction{
    background-image: none;
}

@media print
{    body {
    print-color-adjust: exact;
}
    .no-print
    {
        display: none !important;
    }
}
```