**CSE237A - FINAL PROJECT REPORT**

1. **TITLE**: **AUDIO BASED ACTIVITY RECOGNITION**
2. **MEMBERS**: Sivasankar Palaniappan -spalania@ucsd.edu, A53275703

Chandhini Grandhi – [cgrandhi@ucsd.edu](mailto:cgrandhi@ucsd.edu), A53272378

1. **MOTIVATION:** Smart home is a large and upcoming field, with vast number of applications. They are mainly divided into activity recognition and actuation/automation. Large amount of work has been done in the field of automation. Activity recognition on the other hand, is an upcoming field. Most of the work present today has been performed using smart watch or wearable sensor to detect basic activities. This can yield good results but involves large number of sensors, along with large amount of processing of data originating with different sensors present. Moreover, the cost is high and also power dissipated will be very large. The subject may be uncomfortable wearing too many sensors or wearables. A Fitbit is not affordable by all. Even with large number of sensors, some of the activities cannot be detected. A person may just be sitting but it is not possible to find out if he or she is drinking water or eating. To detect such activities, we propose a model of using microphone which will detect the noise or the sound and find out the activity using an appropriate training model. On the system end, using a raspberry pi makes the entire setup as portable, reducing space complexity and avoiding reducing cost as the system is cheaper than using a standalone CPU or a laptop.
2. **RELATED WORK:** In general, there are large number of academic works related to activity recognition using various sensors such as accelerometer, gyro meter, smart watch, Fitbit. To compare our work, with the previous work, only academic papers pertaining activity recognition using audio was considered.

a. In this field, the amount of work done has been minimal compared to the method mentioned above.

(i)- “Audio-Based Activities of Daily Living (ADL) Recognition with Large-Scale Acoustic Embeddings from Online Videos “performs audio detection but uses audio from online videos, and does not manually obtain data sets.

(ii)-“A Similarity Analysis of Audio Signal to Develop a Human Activity Recognition Using Similarity Networks” uses mobile phones to obtain the audio files and obtains the activity using similarity analysis by calculating Euclidean Distances.

(iii)-"Audio-based human activity recognition using Non-Markovian Ensemble Voting," manages to classify large number of activities by using a sound book which consists of bag of sounds (similar to bag of words) and detects activities using a voting process by comparing with entries from sound book.

b. Our work is different from the current literature in the following aspects

(i) Our project manages to obtain data using an inexpensive setup by just recording it ourselves without having to search for large number of video embeddings for audio of several activities.

(ii) Performing a similarity network would take a large amount of time, since the Euclidean Distances have to be calculated which can be a highly time taking process. Instead we have performed a comparison of various classification-based machine learning algorithms, by considering accuracy and time to execute as parameters, which have not been done by the previous literature.

(iii) Activity recognition based on audio has been done on general purpose systems and has hardly been done on a standalone microcomputer such as a raspberry pi which gives an advantage of being portable and performs all activities which can be done on a general purpose CPU.

1. **HARDWARE COMPONENTS:**

Raspberry Pi and microphone – Less and cheap hardware is the novelty of our project. We have been able to successfully connect the microphone to the raspberry pi and take clean audio samples.

Microphone: We used Plug and Play USB Microphone which can be bought from [here](https://www.amazon.com/gp/product/B077VNGVL2/ref=ppx_yo_dt_b_asin_title_o04_s00?ie=UTF8&psc=1). Raspberry Pi: Raspberry Pi3 Model B Quad-Core 1.2 GHz 1GB RAM.

Power Supply for Raspberry Pi: Official Raspberry Pi Power Supply of 5V, 2.5A.

Below is the picture of the Hardware setup we used:



Hardware Design choices:

* We used this because it was the least expensive and supports a wide range of frequencies (50-16000 Hz) for sound reproduction.
* We chose Raspberry Pi Model 3 as we wanted to create a portable system which can be used anywhere with all features of a normal CPU. Moreover large amount of literature is present for using Rpi for IOT and hence we felt that raspberry pi would be a suitable choice for our project. As data keeps increasing, overhead has become a major issue and hyperdimensional computing (HD) is a growing field to counter this problem. Audio is a large data based parameter and training this on a single system can be a hindrance. So a RPI can be used as a part of HD for future purposes.
* We chose the official power supply to prevent under volting since we are dealing with large amounts of data, to be on the safer side.

1. **SOFTWARE COMPONENTS:**

* Python – Version 3.5
* Raspberry Pi3 Kernel – Raspbian (Debian) version 4.14
* Scikit-Learn – Version 0.21
* Numpy – Version 1.10.1
* TKinter – Version 8.6
* Alsa Audio **–**Version 1.1.7
* Jupyter Notebook-Version 5.7.4
* IDLE- For Python 3.5

Software Design Choices:

* Since we were implementing machine learning algorithms we decided to use a programming language which had libraries for doing the same. Since we were working on audio we needed signal processing libraries as well. MATLAB and Python were two programming languages which implemented both but for a raspberry Pi it was evident that Python was a better choice as the Debian had a pre loaded python software which was free and took lesser space of the memory while MATLAB was paid software which was very large in size ( 2GB of HDD ,needing 4-6GB for installation). Hence we decided to go with Python.
* We used the latest kernel in order to use the latest and most available features and also that we would have better technical support , in case we ran into any issues.
* We used the default python libraries for machine learning (Scikit learn), numerical computations (Numpy) , GUI (tkinter) ,due to their ease of use and simplicity and large number of features and documentation which were available.
* Alsa Audio is Linux’s default audio library which lets to interface devices, record and play sound using command line.
* Jupyter Notebooks were used due to the easy and comfortable UI , which makes subdivide the code into several cells and helps in the debugging process as well.
* IDLE was used for running the completed code as it seemed more professional for running the GUI.

1. **INTEGRATING HW AND SW:**

The novelty of our project was providing a system with minimal hardware and we interfaced a microphone with the raspberry pi. The microphone was connected to the USB port of the Raspberry Pi and the microphone was configured by finding out the device number, name and card number by using the arecord and aplay commands which are part of the ALSA library. Using these the configuration file was written to set the microphone as the default playing and recording equipment. ( Present as .asoundrc file in the code). Despite having a single hardware we faced difficulties as follows:

-We found the recording from the microphone to be very noisy or static after setting it up. We found that the issue was that we were speaking very close to the microphone.

- Using the microphone we were able to record audio with the alsa library which took the default sampling rate of 48KHz and gave mono output. But for python we were not able to use the sound-device library due to the change in the sampling rate. So finding out the appropriate library or command to record or play an audio was a challenge.

- We could not use the pyAudio library as the latency of obtaining the audio was higher (due to continuously streaming using callback mode) than when using the alsa CLI command with subprocess library.

Since we had to collect audio samples from laundry, vacuum cleaner we had to use an ethernet to connect raspberry pi to move to various places and perform various activities. Although the ethernet was not technically a hardware device part of our project, we still faced significant difficulties and large amount of time in connecting the ethernet to the laptop and the raspberry pi and making them work. We solved the issue by reloading the OS (Raspbian). Also we gave a default IPV4 address in the file cmdline.txt in the boot of the OS to communicate with the raspberry pi.

1. **EXPERIMENTAL SECTION:**
2. **CONCLUSION:**

a. We managed to accomplish the following

1. Create an inexpensive activity recognition system based on audio under 100$ by using raspberry pi of 45$ and microphone of 15$ .

2. The system we created is portable and easy to carry from one place to another.

3. Despite using an inexpensive system we managed to create a system with an accuracy of greater than 75%.

4. Provide two different flavors of activity detection i.e by recording a sound and predicting it (non-real time) and also by predicting the activity as and when the sound is played. (real time).

5. Created two separate GUIs(one for real time and one for non-real time) for easy usage ie abstraction in the user end.

6. Compared various classification models of machine learning algorithms for the raspberry pi in terms of execution time and accuracy.

7. Perform the activity recognition using the best machine learning algorithm for raspberry pi based on results obtained in the comparison. Random forest gave maximum accuracy with the least time as seen in the results sections.

Real time implementation required the concept of threading which is an important concept of embedded systems where we had to record audio in one thread and process in the other thread. Also audio requires processing where we had to take the FFT to convert into frequency domain and then obtain the power spectrum and compute the DCT for getting the MFCC values. These are very important components of signal processing which is an important concept in embedded systems as there are large number of digital signal processors (DSP). DSP and threading are concepts which are discussed in the embedded systems class as well. Interfacing the microphone using USB is another important aspect of embedded systems as USB is a very important communication interface.

Overall the project as a whole is closely related to IOT (internet of things) which is nothing but a mixture of embedded systems, networking, data analysis, cloud and distributing computing. Sound is prevalent everywhere and this project uses audio as a commodity to predict the activity performed by the user. This project helps in the future development of embedded systems by contributing to the field of internet of things . Apart from activity recognition aspect of internet of things, this project also contributes to the field of hyper dimensional computing which helps in reducing the overhead of large data computation due to IOT and machine learning by splitting the computations to various dimensions. We provide help for this by computing the efficiency and accuracy of various machine learning algorithms which can help in providing a frame work for hyper dimensional computing.

b. The next steps will be the following :

1. Reduce the size of the dimensions of the feature vector by using principal component analysis to quicken the prediction

2. Try time domain audio features such as zero crossing rate, gradient, mean, centroid,standard deviation, periodicity of the energy spectra which will avoid computation of FFT and help in saving time. Apart from this, different techniques of FFT such as STFT can be tried for solving the above issue.