CS 321 Project

Smart Glove

README

Group 8

Group Members
Abhinav Mishra - 160101005
Nitin Kedia - 160101048
Sparsh Bansal - 160101072
Vivek Raj - 160101076

Overview

1. The project is a smart glove built using raspberry pi and various sensors. The different features of the project include:
    A. Obstacle detection for ease of navigation:
    Ultrasonic sensor is used to detect objects. The blind person lifts his/her arm forward and is alerted by actuating buzzer.The frequency of buzzer varies in accordance with the distance of the forthcoming object.
    B. Gestures to speech output:
    People with speech impairment find it difficult to communicate in a society where most of the people do not understand sign language. Our project focuses on this issue enabling the impaired ones to communicate specific needs with people using gestures. For each gesture there is a specific audio output from the speaker.
    C. Home automation control:
    To make the smart glove useful to a general audience, the same gestures can also be used for home automation providing easy control of home appliances using gestures.
    D. Medicine Scheduler:
    Alarms can be set corresponding to the scheduled time of medicine intake, our glove serves as a reminder triggering the buzzer on the set time. Proper intake is ensured by listening via MQTT to smart locker system opening which symbolises medicine intake.
    E. Fall Detection:
    When a person falls, an abrupt change in acceleration is detected and an alert is sent to emergency contact number through an android app.

2. An app is used to react to the gestures made by the glove.
3. The glove also has two push buttons which are used to switch between different functions of the glove.

Installation

1. SSH into Raspberry Pi present in the glove.
2. Our glove publishes the change in gestures to a server using MQTT. therefore, it needs a broker to publish the data. One of the ways to that is to run a Mosquitto server in a Linux system. To install Mosquitto

    ```
    sudo apt-get install -y mosquitto mosquitto-clients
    ```

    To start listen to a topic, run this command replacing the “test_channel” and “localhost”.

    ```
    mosquitto_sub -h localhost -v -t test_channel
    ```

3. Our program uses Paho python library. Install package using

```
git clone https://github.com/eclipse/paho.mqtt.python.git
cd paho.mqtt.python
python setup.py install
```

4. Finally change the variable SERVER to appropriate server IP of broker in the given glove.py, MPU9150.py and main.py python code. Finally run the code in terminal using `python main.py`. The glove is now broadcasting various messages to the MQTT server.
5. If you want listen to messages of MQTT server, subscribe to /gesture, /fall , /home_automation, /alarm and /locker/0.
6. Install the given app “Smart Glove” and give all permissions in settings.
7. In Pi which is connected to relay switch for home automation, install paho package as given above. Run `python automation.py` in that Pi. Connect various home appliances to the relay switch which can be controlled via gestures.

Uses

1. Gestures to audio
    - Using the app, the user can map the gestures to his/her desired audio output.
    - While in gesture to audio mode, on making specific gesture by bending one or a combination of fingers. The attendant/receiver with app can listen to requested audio mapped to the corresponding gesture.
2. Gestures to home automation
    - Activate this mode using outer push button.
    - Each gesture corresponds to specific home appliance. Using the gesture user can switch the appliance on/off.
3. Fall Detection and alert
    - To set the Emergency contact, press “Emergency contact” button in the app and set appropriate phone number and message.
    - Whenever a fall is detected, the given message is sent to the emergency contact.
4. Medicine Scheduler
    - Set the time in the companion app. The glove buzzer will trigger at designated time continuously till an acknowledgement is received from Smart Locker that it has been opened thereby indicating the dose has been taken.
5. Obstacle Detection and Navigation
    - Activate this mode using the inner push button.
    - Point the wrist towards the desired direction and if an
    - is within 50cm the buzzer will buzzz with frequency inversely proportional to distance..

Hard-coded Values

1. The server IP address is hardcoded.
2. The threshold values which determines when a finger will be registered bent is taken as the average of ADC reading when the flex sensor is straight and bent.
3. Threshold for fall detection is taken to be 2 time acceleration due to gravity. This is done to capture the momentary free fall before a just as large impact in the opposite direction after hitting the ground.