Raspberry Pi-Based Smart Infant Monitoring System

Darshan Doddaghatta[†]
Computer Science
SUNY Binghamton
Binghamton, NY USA
ddoddag1@binghamton.edu

Swaroop Gowdra
Shanthakumar
Computer Science
SUNY Binghamton
Binghamton, NY USA
sgowdra1@binghamton.edu

ABSTRACT

Nowadays, one of the most significant challenges that faces many families is baby care. Parents cannot continuously observe or monitor their babies all the time. Baby monitors helping reassuring millions of parents that their children are safe. Aim of the project is to produce a baby monitor system which will monitor the baby and alert the parents when a cry is detected. The idea is to design a system that will simplify the process of monitoring the baby by using Raspberry Pi device. The proposed system will have many features such as: audio recording, displaying live video streaming and push messages notification to the android app. Finally, the proposed system is tested and proved its effectiveness and functionality.

KEYWORDS

Internet of Things, Raspberry Pi , Sound processing, Firebase, Cry detection.

1 Introduction

The smart infant monitor system allows parents to work without any worries about their children and assist them to keep an eye on them remotely. Therefore, the aim of this project is to build a low-cost system that provides high-quality features.

The smart infant monitor system constantly monitor sound from the baby and checks for a cry. When a cry is detected it pushes a notification to the parent using Firebase system. When parent receives the notification on their android app, they can connect to the live video stream and check the state of the baby remotely. We aim to design a cry detection algorithm which will always run on the raspberry pie and also a push message notifying communication system between client android app and raspberry pie system via Google Cloud Firebase System.

2 System Design

2.1 Hardware.

The major Components are

2.1.1 Raspberry Pi. We have used Raspberry Module as our computation device which hosts our software and collects reading from attached peripherals and performs actions based on set condition. Processor speed ranges from 700 MHz to 1.4 GHz for the Pi 3 Model B+ or 1.5 GHz for the Pi 4; on-board memory ranges from 256 MiB to 1 GiB random-access memory (RAM), with up to 4 GiB available on the Pi 4..

2.1.2 Raspberry Pi Camera. The Raspberry Pi Camera v2 is a high quality 8-megapixel Sony IMX219 image sensor custom designed add-on board for Raspberry Pi, featuring a fixed focus lens. It's capable of 3280 x 2464-pixel static images, and also supports 1080p30, 720p60 and 640x480p60/90 video. It attaches to Pi by way of and uses the dedicated CSi interface, designed especially for interfacing to cameras. We have used this hardware to collect video data and perform actions, stream video.

2.1.3 USB Microphone. They are the easiest way of getting a microphone working Raspberry Pi. Advantages of using a USB microphone is that it is plug and play. The Raspbian operating system will automatically detect the microphone when its plugged in. We have used this hardware to collect voice data and perform actions on the collected sound.

.2.1.4 Android Device. We have used an android device as an client device which receives the information sent by the pi and alert's user who is monitoring the infant.

2.2 **Software**

The major software utilized is listed below:

2.2.1 Raspberry Pi OS. It is Debian-based operating system for Raspberry Pi.

2.2.2 Python Programming environment. We have used python programming language to build the cry detection.

2.2.3 Android Studio. We have used java has a programming language to build an android app using android studio as software framework.

2.2.3 Firebase. We have used Google Cloud Messaging for mobile notification service it enables pie to send notification data or information to the mobile application.

3 Implementation

3.1 Cry Detection

It is important that to the parents who are far from their infants to have a quick notification in case of crying. In our proposed system, we take this feature into consideration. Fig. 2 illustrates the flowchart for the cry detection.

We are using Convolutional Neural Network to classify the sound recorded which is processed every 10 seconds. We are training the model with 3 different types of sounds which are labeled as Noise, Silent and Cry. This dataset is split into train and test sets. The sound is captured every 10 seconds and written into the file in the WAV format. The audio file is read, and the features are extracted by applying Mel Frequency Ceptral Coefficients using the Librosa library. We pass these features for training the model. Our model consists of 14 layers which are convolutional 2d layers, pooling, dropout and dense layers. The model is trained for 100 epochs and the model is saved checkpointing to the trained model. We got an accuracy of 90 percent when evaluation of the model was performed with test set.

For the prediction part, we load the model which is saved and the audio which is recorded to file is read and features are extracted and passed to model for the prediction. The dense layer which is a softmax layer gives the probability of belonging to a class and that is how the sound is classified to which category it belongs.

3.2 Video Streaming

It is crucial to the parents to know why the infant is crying when detected. Once the cry is detected, the parent is alerted with a notification which prompts the user if he wants to watch the video stream of the baby to know what is wrong and act accordingly.

We are using a RTMP protocol to stream video from Raspberry Pi Camera. On the client side, RTMP is implemented in the Mobile Application. A socket port is associated with the user and used to stream the video using the same. This is handled using push notification. The socket connection details are sent to the user when notified and when user chooses to watch the video, it is automatically decoded, and connection is established.

3.2 Push Notification

For notifying the parents when there is detection of cry, we are using the Firebase Cloud Messaging service. Each user is associated with a

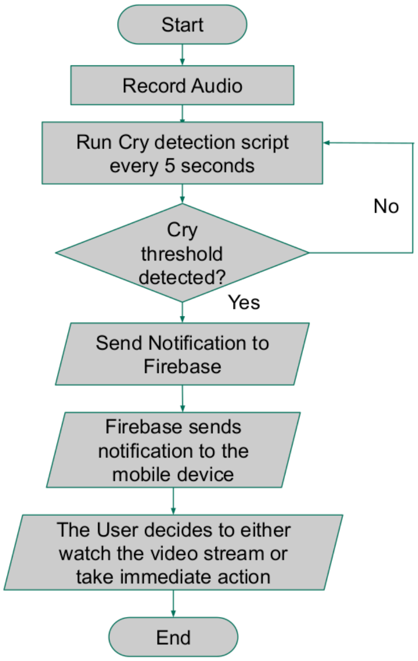
Firebase token. When the user logs in to the application, a firebase token is registered into the console and sent also to our server to link the user with that firebase token. Once the cry is detected, we see the user linked firebase token and trigger a message with the firebase token. Then the firebase triggers a notification to the user. We also send payload information with the notification which is extracted in the client end and used for video streaming.

3.3 Mobile Application

Mobile application plays the vital role in taking actions. Since a parent is notified with the alerts of the baby cry and choosing to do a live video stream or not . We have built an Android application for this purpose.

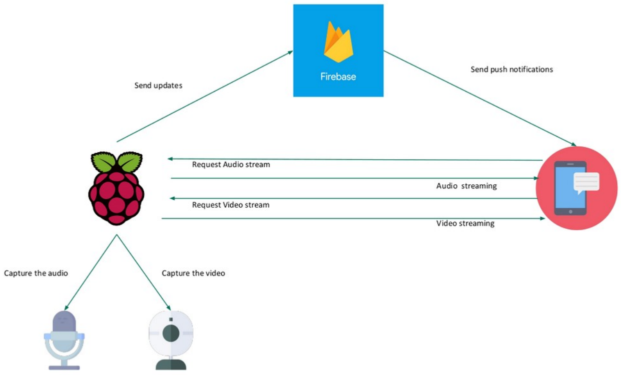
Data Flow

Below is the data communication between our modules.


Start
Record Audio
Run Cry detection script every 5 seconds
Cry threshold detected?
No
Yes
Send Notification to Firebase
Firebase sends notification to the mobile device
The User decides to either watch the video stream or take immediate action
End

Control Flow

Below is the communication between our modules.



CONCLUSION

We built an Android application to Live video stream. We have trained a cnn model for cry detection and have created an communication between sever side app and client mobile app using Firebase as our interface. We have reached our goal for sensing audio and classifying it with help of cnn model. We believe our application will monitor and notify.

ACKNOWLEDGMENTS

The internet has a big part in this project. We learned about programming android application,socket programming,integration with google Firebase and working on machine learning python modules. Thus, the project has made us realize our potential by applying theoretical CS concepts into a real-world working prototype. We would like to thank our professor Mo Sha in Binghamton university, and our teaching assistant Junyang Shi for their helps and for leading and encouraging us to create an IOT project as a solution to a real life problem.

REFERENCES

[1] https://www.raspberrypi.org/documentation/

[2] https://developer.android.com/docs

[3] http://firebase.google.com/docs/

[4] https://docs.python.org/3/