

Network Application Design Project Proposal: Emote Music Player

Team 7

Names: Shashank Shinde, Sanaya Sanghvi, Sasha Morgan

Date: 03/25/2018

For: Dr. William Plymale

Table of Contents

Concept of Operation	3
System Overview	4
System Diagram.....	4
Testable Requirements.....	5
Hardware list.....	6
Project Schedule.....	6
Development Tasks.....	6
Milestones	6

Concept of Operation

Purpose: To create a music player that recognizes the user's facial expression and plays music depending on their current emotion.

Goal: Our system analyses a person's emotions using image recognition and plays music accordingly. For example, if the person shows signs of anger or stress through contempt or other negative emotions, the system will play calming music, however if the person is in a positive mood, it will continue to play uplifting music. Thus, our primary goal is to elevate a person's mood using appropriate music.

Importance: Our project is important because it uses music to improve a person's mental wellbeing. Emotional regulation is an essential component to mental health. As an example, some studies have shown classical music to be a form of antidepressant. Using our emotive music player, we aim to alleviate the user's mental stress and anxiety, and maintain a sense of calm.

User Interaction: The user puts their music preference in a USB stick which is then plugged into a raspberry pi. After booting the system, the user faces the camera so it can take a screenshot to measure their current emotional state. The system then plays the appropriate song, and once the song is over the camera takes another screenshot to detect the next emotion.

Outcome: If the user is anxious, after playing calming music, their mood should improve until the emotion of "joy" or other positive emotions exceeds that of "anger", "sad" or "contempt", which signifies that our system has accomplished its goal of improving their mood, or calming them down.

System Overview

The following section provides a logical breakdown of the concept of operation.

System Diagram

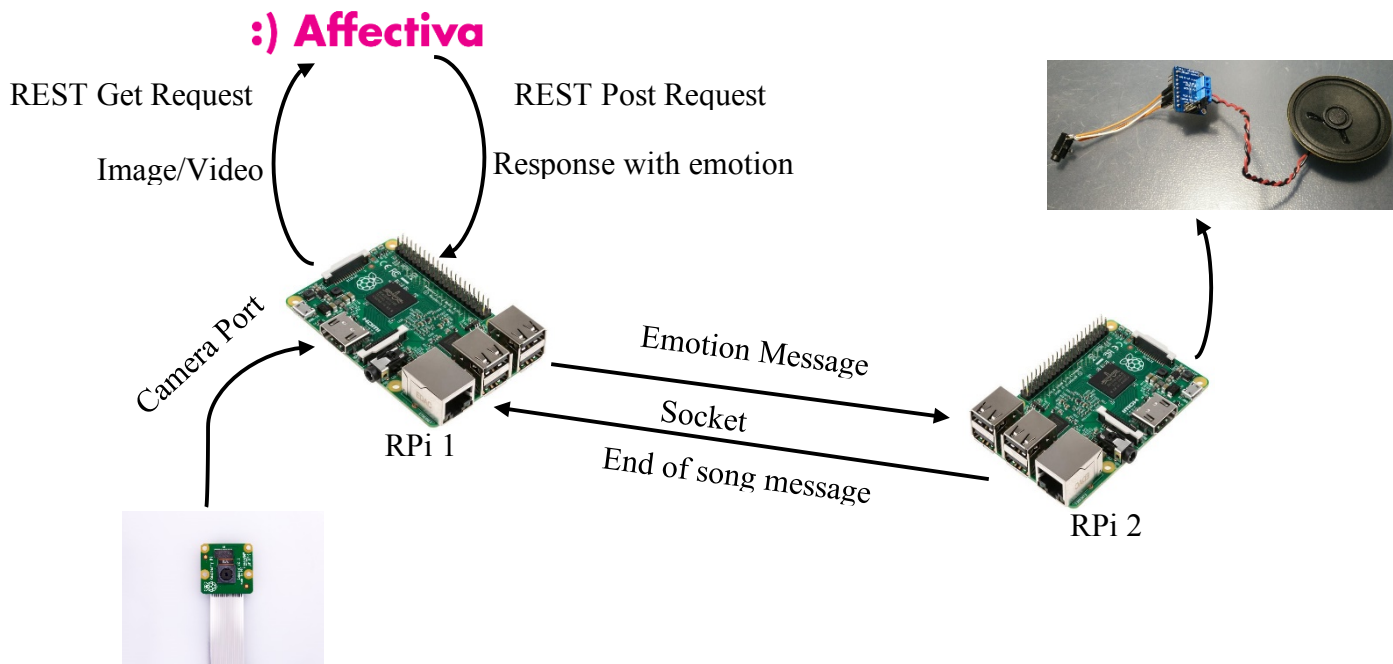


Figure 1: *System Diagram of project*

In figure 1, the RP1 represents raspberry pi 1. It is used to interface with the camera using the camera module. The camera takes a still image of the person and sends it back to RP1. The RP1 sends this image to the Affectiva API through an HTTP POST request to their Media Processing API. The Affectiva API uses its deep learning algorithms to find the emotions represented in the image and a GET request returns a JSON response.

The response is parsed in RP1 to find the emotions represented in the image, which include “Anger”, “Contempt”, “Disgust”, “Fear”, ‘Joy”, “Sadness” and “Surprise”. The predominant emotion is stored by the RP1.

The RP1 communicated with RP2 (raspberry pi 2) via sockets and sends a message containing the predominant emotion. The RP2 finds the correct song for that emotion and plays it through the speaker. Once the song has finished playing, it sends a “Song finished” message back to RP1 which uses the camera to take another screenshot and repeat the process.

Testable Requirements

1. The raspberry pi 1 must interface with the camera using the camera port labeled CAMERA on the board.

The raspberry pi should be able to interface with the camera to take a still image, display the image to the user, and send it to the affective api

2. The raspberry pi 1 must connect to the affectiva API and send the image using a POST request, which requires sending a “job” to the API, which includes creating a job URL.

The response from the POST request should include the job url in the “Location” header needed to perform the get request.

3. The raspberry pi 1 must send a GET request to the affectiva API’s index service using the job url.

The response from this request is parsed to find the emotions represented in the image and the predominant emotion is found by comparing the percentages of all the emotions returned.

4. The raspberry pi 1 must send this data using a connection socket to the raspberry pi 2, which finds the right song based on the predominant emotion.

The right type of song is heard through the speaker, and once the song finishes, it sends a message containing “Song has finished” back to raspberry 1.

5. The raspberry pi 1 must receive the “Song has finished” message and interface with the camera to take another screenshot.

Hardware list

- 1) Raspberry Pi Kit (x2) (*Already have it*)
- 2) Raspberry Pi Camera Board v2 (x1)
 - a. Price - \$25.00
 - b. Link - <https://chicagodist.com/products/raspberry-pi-noir-camera-v2-infrared-sensitive-camera?src=raspberrypi>
- 3) Speaker for the Raspberry Pi
 - a. Price - \$7.99
 - b. Link - <https://www.dexterindustries.com/shop/speaker-for-the-raspberry-pi/>

Project Schedule

This section is about the development tasks and milestones.

Development Tasks

Task	Person in charge	Date
Interfacing with camera and capturing still image	Sasha Morgan	04/01/2018
Connect to Affectiva API and send image to API	Sanaya Sanghvi	04/07/2018
Get the response from API	Sanaya Sanghvi	04/07/2018
Send response (emotion) from API to RPi 2	Shashank Shinde	04/12/2018
RPi 2 sends message (song finished) to RPi 1	Shashank Shinde, Sasha Morgan	04/12/2018

Table 1: Development Tasks

Milestones

- 1) Beta Demonstration – April 26 to May 01
- 2) Final Demonstration – May 7