

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**  
**FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN**



**Diseño de Sistemas Controlados por Computador**

**Proyecto final**

**Camera Voice Tracker**

**Estudiante**

**Christian Vergara Marcillo**

[\(chroverg@espol.edu.ec\)](mailto:chroverg@espol.edu.ec)

**Esteban Muñoz Guevara**

[\(edmunoz@espol.edu.ec\)](mailto:edmunoz@espol.edu.ec)

**Profesor**

**Ph.D Federico Domínguez**

[\(federico.dominguez@cti.espol.edu.ec\)](mailto:federico.dominguez@cti.espol.edu.ec)

**II Término, Año Lectivo 2015-2016**

# Proyecto Final

## Cámara sigue voz utilizando Raspberry Pi

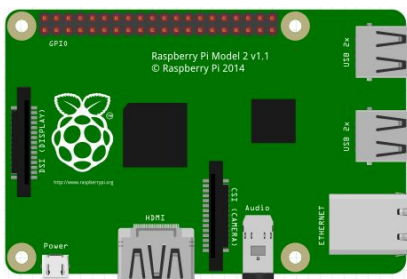
### Descripción

El proyecto consiste en diseñar y programar una cámara que siga la voz registrada con un arreglo de micrófonos a través de una Raspberry Pi, para esto se procesará y filtrarán las señales que envía el arreglo de micrófonos.

### Módulos utilizado

Para el proyecto utilizaremos:

- Raspberry Pi 2
- Servo motor
- Cámara (webcam)
- Modulo wifi
- Protoboard
- Matriz de micrófonos ()



**Figura#1**  
Raspberry Pi 2



**Figura#2**  
Voice tracker II



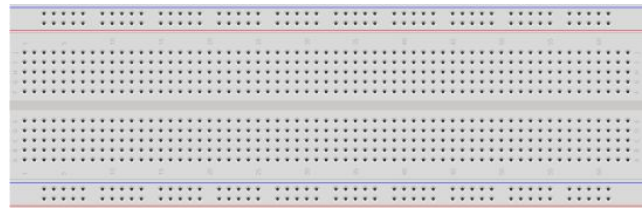
**Figura#3**  
Servomotor



**Fegura#4**  
Camera



**Figura#5**  
Módulo Wifi



**Figura#6**  
Protoboard

En el Raspberry Pi se conectará el servomotor que a su vez estará conectado (con un soporte) a la cámara web, la salida obtenida por el arreglo de micrófonos también está conectada al Raspberry Pi para detectar el origen del sonido, la cámara web deberá estar conectada también al Raspberry Pi para poder visualizar las imágenes obtenidas, el video obtenido por la webcam deberá enviarse mediante internet.

### Funcionalidades

- **Detección del origen del sonido mediante el arreglo de micrófonos**

Para la detección del audio o el sonido que hará que el motor se mueva en esa dirección, utilizaremos un arreglo de micrófonos existente en el Laboratorio, el cual con procesamiento de audio genera en su salida una señal que nos proporcionará hacia dónde debe moverse el servo.

- **Movimiento del servo en la dirección del sonido**

EL movimiento del servo se realizará una vez obtenido el origen de la señal, el servo tendrá adherida la webcam, mediante algún soporte, y está registrará el streaming de donde se produce el origen del audio.

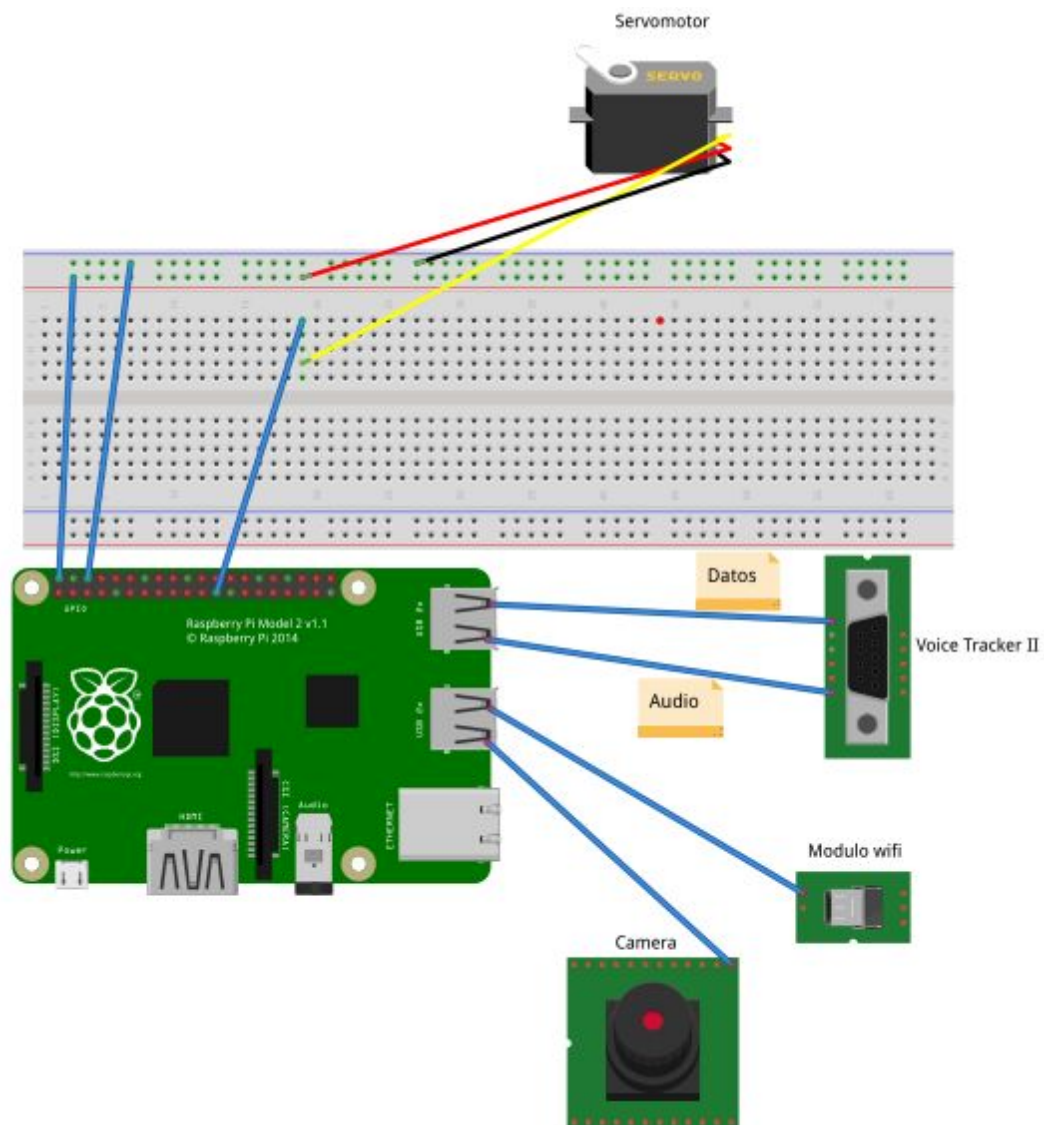
- **Streaming de la webcam**

El streaming de video deberá ser enviado mediante internet desde la raspberry pi, para esto utilizaremos el módulo wifi, y la visualización será mediante internet.

### Retos:

- Filtrar la señal de audio obtenida por el arreglo de micrófonos para su posterior procesamiento con la raspberry pi.
- Conocer cómo implementar el funcionamiento de un servo en una raspberry pi.
- Visualizar el streaming de video en internet.

## Diagrama



**Figura#7**  
Diagrama

La figura#7 muestra la manera en que se encuentra el diagrama del proyecto, el voice Tracker II(Figura#2) tiene 1 entrada y 1 salida, una la cual envía el audio hacia el raspberry y el otro envía datos en forma binaria, el cual indica el origen del sonido que puede llegar dentro de un ángulo de 180° con respecto a su posición.

Este dispositivo va conectado al puerto usb del raspberry pi (Figura#1), de igual manera el módulo wifi (Figura#5) y la cámara(Figura#4).

## Configuración en Raspberry Pi

Para realizar una conexión entre una laptop y la raspberry pi, debe conectar la Raspberry pi a la laptop, mediante un cable de red, esto creará una red entre ambos dispositivos, se debe configurar una IP estática tanto en la laptop y en la Raspberry, la máscara de subred puede quedar como se muestra:

En la laptop:

192.168.1.5

255.255.255.0

En la Raspberry Pi:

192.168.1.89

255.255.255.0

Para configurar la IP en la Raspberry Pi se debe modificar el documento que contiene las configuraciones de las interfaces de red:

```
sudo nano /etc/network/interfaces
```

En la interfaz de ethernet (eth0) se ingresa la IP y la Máscara de subred , y se cambia de “manual” a “static”

Es necesario reiniciar la raspberry con el comando:

```
sudo reboot
```

Para activar la configuración, se realiza con el comando:

```
sudo ifup eth0
```

Para verificar que las configuración esté correcta:

```
sudo ifconfig
```

Esto nos permitirá establecer una conexión con la raspberry y para conectarnos a la misma podemos usar cualquier programa que permita conectarse como cliente a un servidor vía ssh, (El programa utilizado en el proyecto fue Putty)

Si deseamos ver el escritorio de la Raspberry podemos usar programas como TightVNC

Para realizar el streaming web, se utilizó la herramienta MOTION para raspberry pi

Antes de proceder con el streaming web se debe verificar que la cámara sea detectada por la Raspberry, para esto se ejecutan los siguientes comandos:

Para listar los dispositivos conectados a la

```
lsusb
```

Para probar la camara podemos utilizar luvview

```
sudo apt-get install luvview
```

Una vez instalado, lo iniciamos con luvcview y si la cámara está correctamente detectada por la Raspberry , debería mostrarse en una pantalla la imagen capturada por la cámara, cabe indicar que eso no será posible si se está utilizando la RPi mediante ssh, debemos ejecutarlo desde el escritorio de la misma.

### **Motion:**

- Actualizar la Raspberry Pi  
sudo apt-get update  
sudo apt-get upgrade
- Instalar Motion  
sudo apt-get install motion

Es necesario configurar Motion para poder utilizarlo adecuadamente.

- Para esto modificamos el archivo de configuración de Motion:

```
sudo nano /etc/motion/motion.conf
```

- Buscamos las siguientes líneas de código y las seteamos de la siguiente manera:

```
daemon on
webcam_localhost off
Optional (Don't include the text in brackets)
webcam_maxrate 100 (This will allow for real-time streaming but requires more bandwidth)
framerate 24
width 640 (This changes the width of the image displayed)
height 480 (This changes the height of the image displayed)
```

- Es necesario cambiar el parámetro del documento motion con:  
sudo nano /etc/default/motion

```
start_motion_daemon=yes
```

- Para iniciar el servicio de motion:  
sudo service motion start      o      sudo motion start
- Para detenerlo:  
sudo service motion stop      o      sudo motion stop

Para visualizar el video es suficiente ingresando la ip de la Raspberry, seguida del puerto en el que transmite Motion (por default 8081), así: 192.168.1.89:8081

### **Detección del origen del sonido mediante el arreglo de micrófonos y rotación de servo motor**

Los arreglos de micrófonos envían, o se comunican mediante un puerto serial la información del origen del sonido:

La señal se transmite mediante con la siguiente configuración:

Binarios de 8, No Paridad, 1 Bit de Stop; Comúnmente referida como: 8,N,1. a una velocidad de 2400 baud.

Estos números binarios al traducirlos a Decimales representan valores entre 0 y 255, en donde 0 indica completamente a la Derecha y 255-255 completamente a la izquierda

**Problema: Recopilacion de informacion y traducción de binarios a decimales.**

**Solución:**

Para obtener esta información puede usarse cualquier programa que lea información recibida a través de un puerto serial, o puede ser recopilada y traducida con un lenguaje de programación, en nuestro caso utilizamos python 2.7.



**255**

**125**

**0**

```
import serial
while True:
    dev = serial.Serial('/dev/ttyUSB0',baudrate=2400,timeout=5.0)
    a = bin(ord(dev.read()))
    dec = int(a,2)
    print dec
```

La sección de código anterior permite obtener el origen del sonido que transmite el arreglo de micrófonos en el puerto serial, detectado en la interfaz usb.

**Problema: Filtrado de ruido para el movimiento del servomotor.**

**Solución:**

La información recopilada es filtrada con un filtro de PROMEDIO MÓVIL con 30 elementos, además se hizo una decimación de 30 elementos.

Finalmente se leen cada 30 elementos y de estos se establecen sectores (6 sectores) dependiendo de la rotación del servomotor, esto, porque el arreglo de micrófonos proporciona valores similares para un mismo sector y genera ruido en el servomotor, el servomotor solo rota en 180 grados, entonces eso representa 6 secciones de 30 grados, para esto se traducen los números (0-255) a porciones de 30 grados, la función que realiza esta traducción se encuentra en el script de python que se encuentra en este repositorio.

## Enlace repositorio



<https://github.com/edmunoz/Proyecto-final-dscc>

## Referencias

<http://razzpisampler.oreilly.com/ch05.html>

<http://pimylifeup.com/raspberry-pi-webcam-server/>

<https://dsputpl.files.wordpress.com/2010/03/moving-average.ppt>

[http://elinux.org/RPi\\_Setting\\_up\\_a\\_static\\_IP\\_in\\_Debian](http://elinux.org/RPi_Setting_up_a_static_IP_in_Debian)

<http://razzpisampler.oreilly.com/ch05.html>

<https://www.youtube.com/watch?v=ddlDgUymbxc>