
Detector de rostros

Gerson Oswaldo Ruiz Ramirez - 202011405
Paulo Fernando Merida Salazar - 202002042
Byron Estuardo Solís González- 201906588
Victor Abdiel Lux Juracán - 201403946
Juan Carlos David Gomez Chanax - 201612106
Anthony Alexander Aquino Santiago - 202001923

I. INTRODUCCIÓN

En ocasiones, al analizar cámaras de vigilancia o videos específicos, es importante extraer de las grabaciones, datos relevantes que permitan generar información. La IoT es aplicable para realizar análisis en diversas situaciones, una de las cuales es la detección de rostros en grabaciones. En este caso se aplicó la IoT en una app de escritorio, la cual analiza la imagen que proporciona la cámara web local e informa al usuario si se encontraron rostros en la cámara. Así mismo, envía una alerta a un dispositivo electrónico, encendiendo un led y sonando una alarma. indicando si se detectó o no un rostro. La app de escritorio informa también al usuario la cantidad de rostros que se registran a lo largo del tiempo, incluso si la app se cierra. Éste tipo de app, puede ser útil para implementarla en cámaras de seguridad en hogares, para, por ejemplo, encender o apagar luces dependiendo si se detectan caras o no, o para sonar alarmas fuertes en las entradas de las casas si se detectan intrusos.

II. Objetivos

A. General

- Desarrollo de un sistema capaz de monitorear la presencia de rostros al analizar la imagen de una cámara.

B. Específicos

- Alertar al usuario de la app, la presencia de rostros, utilizando un dispositivo electrónico.
- Visualizar el registro histórico de la detección de rostros desde la primera ejecución de la app.

III. Desarrollo de la práctica STACK DESIGN FRAMEWORK

A. Infraestructura del Producto

a. Materiales físicos:

- Arduino Uno con ATmega328
- Protoboard
- Leds
- Bocina
- Resistencias

b. Materiales digitales

- Base de datos en MySQL
- App de escritorio utilizando Python con la librería OpenCV para la detección de rostros.
- App en Processing para observar el registro de rostros detectados a lo largo del tiempo.

B. Sensores

- No se utilizaron sensores en éste proyecto.

C. Conectividad

El dispositivo está compuesto por un Arduino, el cual por medio de un puerto serial se comunica con la laptop que ejecuta la app de escritorio. A través de ese puerto, la app de escritorio envía alertas si detecta o no rostros, utilizando la detección de rostros de la librería OpenCV. Si detecta un rostro, se envía un valor y el arduino enciende un led y emite un sonido de aprobación, si no, envía otro número para encender otro led y emitir una alerta de no detección. Así mismo, si se detecta un rostro, se incrementa un valor de contador guardado en una base de datos MySQL. La base de datos es consultada por medio de la app en Processing para desplegar el dato de rostros detectados en tiempo real. Todo se almacena localmente.

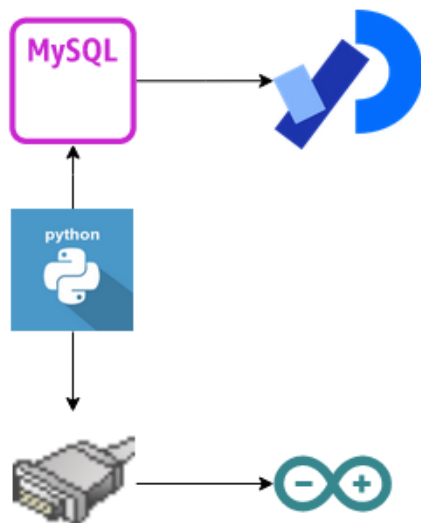


Figura 1: Conectividad del dispositivo

D. Analítica

Para la parte de analítica del dispositivo, se utilizan los datos obtenidos mediante la cámara en Python, para hacer sonar una alarma y encender un LED según el dato obtenido.

La forma en que se obtienen las medidas a través de la cámara es:

- Bandera si hay rostro o no: Se envía al arduino un caracter para que lo lea y determine si debe sonar la alarma de confirmación de rostro o no detección de rostros.

caracter: 'g' si hay o 'r' si no detecta rostros.

E. SmartApp

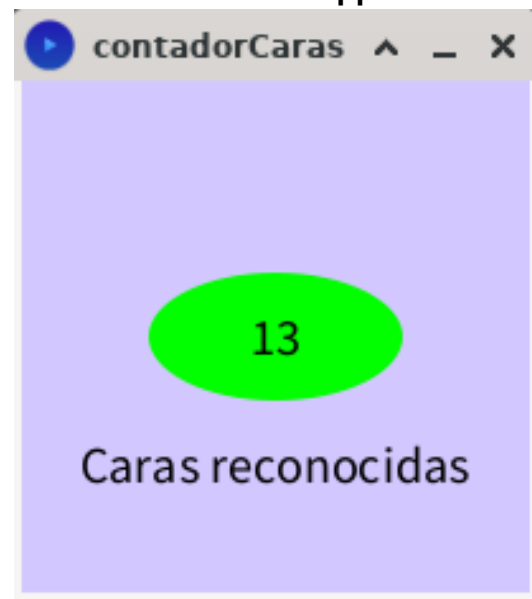


Figura 2: Registro de las rostros reconocidos

F. Boceto del prototipo

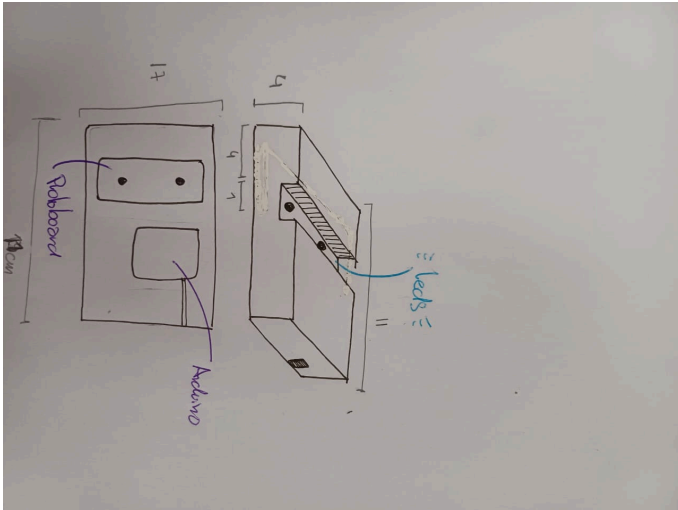


Figura 3: Boceto del prototipo

G. Construcción del prototipo

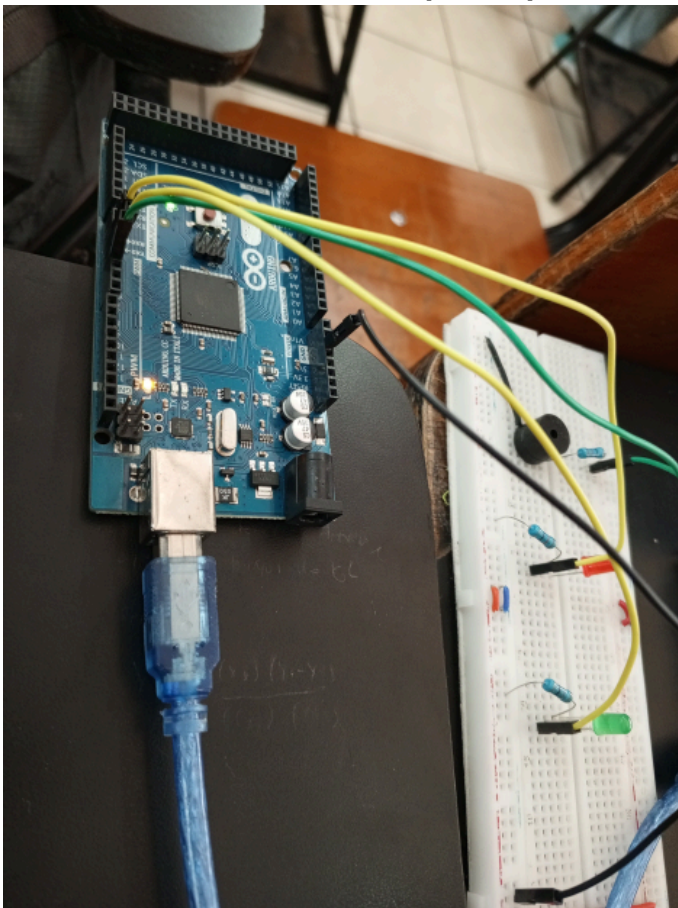


Figura 4: Construcción del prototipo

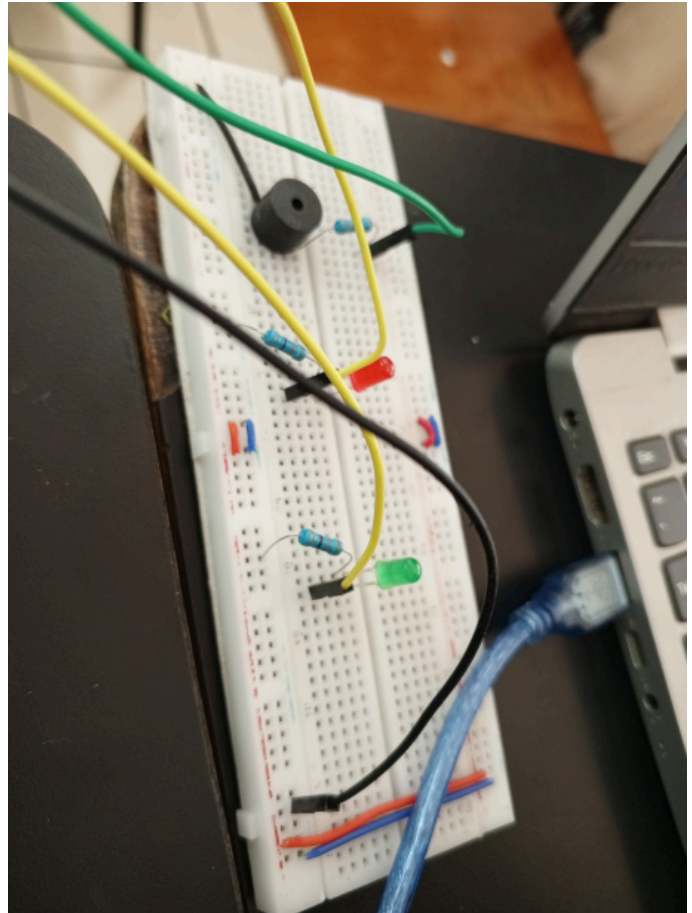


Figura 5: Construcción del prototipo

H. Aplicación

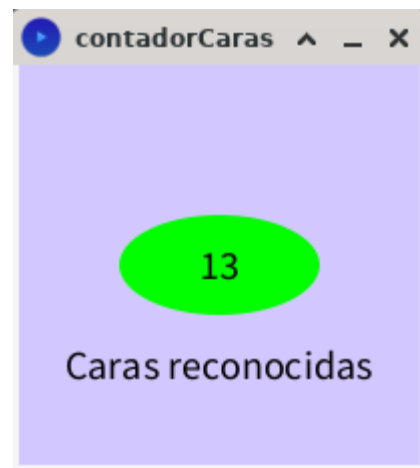


Figura 6: Registro de los rostros reconocidos

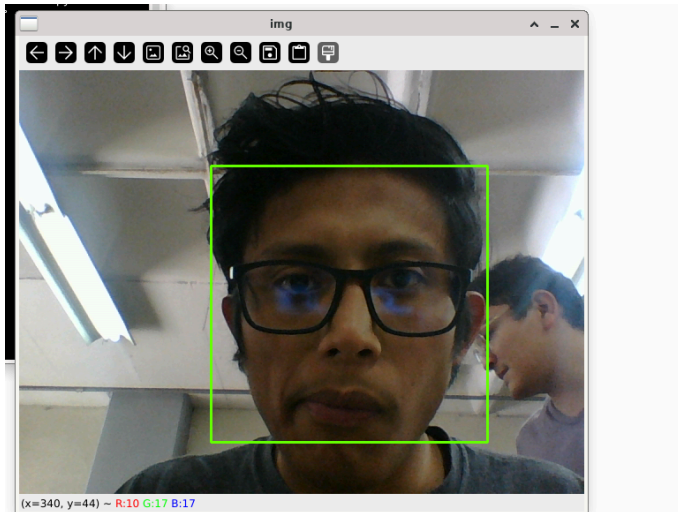


Figura 7: App de escritorio usando la cámara web

Link del repositorio:

https://github.com/pauloMerida/ACE2_1S24_G5.git