

# **ESCUELA POLITÉCNICA NACIONAL**

## **ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

### **IMPLEMENTACIÓN DE UN SISTEMA DE RASTREO Y ENCENDIDO AUTOMÁTICO VEHICULAR, MEDIANTE RECONOCIMIENTO FACIAL**

**TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN  
ELECTRÓNICA Y TELECOMUNICACIONES**

**LUIS MIGUEL CARMILEMA SIMBAÑA**

luiss\_car@hotmail.com

**JHONY DANIEL PÉREZ MALTE**

jhony.perez@epn.edu.ec

**DIRECTOR: Ing. FANNY PAULINA FLORES ESTÉVEZ MSc.**

fanny.flores@epn.edu.ec

**CODIRECTOR: Ing. MÓNICA DE LOURDES VINUEZA RHOR MSc.**

monica.vinueza@epn.edu.ec

**Quito, junio de 2019**

## **DECLARACIÓN**

Nosotros, Luis Miguel Carmilema Simbaña y Jhony Daniel Pérez Malte, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado en ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

**LUIS MIGUEL CARMILEMA  
SIMBAÑA**

---

**JHONY DANIEL PÉREZ MALTE**

## **CERTIFICACIÓN**

Certificamos que el presente trabajo fue desarrollado por Luis Miguel Carmilema Simbaña y Jhony Daniel Pérez Malte, bajo nuestra supervisión.

---

Ing. Fanny Flores MSc.  
**DIRECTOR DE PROYECTO**

---

Ing. Mónica Vinuesa Rhor MSc.  
**CODIRECTOR DEL PROYECTO**

## DEDICATORIA

*¿Cuál es el mejor regalo?, desafíos contra ti mismo, metas para subir al siguiente nivel y ponerte tú mismo a prueba. Aprendí que el progreso no es un camino recto, sino que tiene una serie de subidas y bajadas, caminos equivocados. Entonces cuál es mi punto, la vida no es justa, la vida no es fácil, la vida es incierta. Simplemente trato que mis acciones hablen más que mis palabras.*

*Nos enseñan siempre que las cosas buenas vienen para aquellos que esperan, sin embargo, la verdad es que las cosas buenas vienen para aquellos que trabajan duro todos los días y sin descanso, se dirigen hacia sus metas con un propósito y deseo que nadie puede impedir, ese ejemplo me lo llevo siempre de mis padres, ustedes Amparito Simbaña y Patricio Carmilema.*

*Trabajar duro para lograr cada objetivo y la culminación de una de las etapas en mi carrera profesional es el inicio de un gran sueño, porque sé que con entrega y responsabilidad todo realmente todo es posible, ustedes son la prueba viva, porque literalmente empezaron sin nada.*

*Yo no habría llegado a donde estoy sin esta ética de trabajo inculcada día a día en mi hogar y este logro va dedicado a ustedes papi y mami.*

*Finalmente, dedico este trabajo a mis hermanos, Pato por mostrarme que se debe tener fortaleza para afrontar los retos que nos pone la vida. Gaby y Samy, por ser mis cómplices en momentos felices y tristes de mi vida y ayudarme a superarlos con amor.*

*Luis Miguel*

*Este proyecto va dedicado a mis padres, quienes han sido mi guía durante toda mi carrera estudiantil, gracias a sus valores inculcados, respeto, humildad, lealtad y perseverancia, he podido escalar uno de los múltiples peldaños que aún faltan por recorrer hasta alcanzar el éxito.*

*Jhony Daniel*

## **AGRADECIMIENTO**

*A la ciencia que trata la realidad actual y a Dios que se ocupa de la moralidad humana solo así no hay contradicción, gracias por crear este universo tan maravilloso en donde vivimos.*

*Agradecemos a la Escuela Politécnica Nacional, a esta gran institución por brindarnos la oportunidad de obtener una profesión y ser personas útiles a la sociedad.*

*A nuestra codirectora de tesis, Ing. Mónica Vinueza quien con su conocimiento y experiencia supo guiarnos a lo largo de nuestra carrera profesional.*

*Finalmente, un sincero agradecimiento a nuestra directora de tesis, Ing. Fanny Flores, por brindarnos la confianza y sobretodo creer en nosotros.*

*Gracias totales.*

*Luis Miguel, Jhony Daniel*

# ÍNDICE DE CONTENIDO

DECLARACIÓN .....	I
CERTIFICACIÓN .....	II
ÍNDICE DE CONTENIDO .....	V
ÍNDICE DE FIGURAS .....	VIII
ÍNDICE DE TABLAS .....	X
RESUMEN.....	XI
<i>ABSTRACT</i> .....	XII
<b>1. INTRODUCCIÓN</b> .....	1
1.1 MARCO TEÓRICO .....	3
<i>HARDWARE</i> .....	3
<i>RASPBERRY PI 3 B+</i> .....	3
Introducción .....	3
Características.....	3
Procesador .....	4
Puertos GPIO .....	4
Puertos USB.....	5
Puerto <i>Ethernet</i> .....	5
Audio/MIC .....	5
Puerto <i>CSI</i> .....	5
Puerto <i>HDMI</i> .....	5
Puerto <i>PWR</i> .....	5
Puerto <i>DSI</i> .....	6
Cámara <i>Raspberry PiNoir V2</i> .....	6
<i>LCD TOUCH SCREEN DISPLAY 3.5"</i> .....	7
<i>GPS</i> .....	7
Funcionamiento del GPS.....	8
Receptor <i>GPS UBLOX NEO-6M</i> .....	8
Diodo <i>LED</i> .....	9
Resistor .....	9
Interruptor eléctrico o <i>Switch</i> .....	9
Relé .....	9
Transistor 2N3904.....	10
Tarjeta Micro-SD .....	10
<i>SOFTWARE</i> .....	11

<i>Raspbian</i> .....	11
<i>OpenCV</i> .....	11
<i>Python 3</i> .....	11
<i>VNC</i> .....	12
<i>Centos 7</i> .....	12
Servidor <i>LAMP</i> .....	12
<i>phpMyAdmin</i> .....	13
<i>HTML</i> .....	13
<i>Java script</i> .....	13
IP Pública.....	14
<i>DNS</i> .....	14
SISTEMAS DE RECONOCIMIENTO FACIAL EN AUTOMÓVILES .....	14
Algoritmo <i>LBP</i> .....	15
<b>2. METODOLOGÍA</b> .....	16
2.1 MÉTODO COMPARATIVO .....	16
2.2 MÉTODO ANALÍTICO .....	17
2.3 MÉTODO EXPERIMENTAL .....	17
<b>3. RESULTADOS Y DISCUSIÓN</b> .....	19
3.1 DISEÑO DEL SISTEMA .....	19
<i>HARDWARE</i> .....	19
<i>RASPBERRY PI 3 B+</i> .....	20
Conexión de la cámara <i>PiNoir V2</i> a la <i>Raspberry Pi</i> .....	20
Conexión del Receptor <i>GPS UBLOX NEO-6M</i> en la <i>Raspberry Pi</i> .....	21
Conexión de la <i>LCD touch screen display 3.5"</i> , a la <i>Raspberry Pi</i> .....	22
Conexión de un <i>Switch</i> y <i>LED</i> indicador a la <i>Raspberry Pi</i> .....	23
Conexión de la <i>Raspberry Pi</i> al motor de arranque del vehículo .....	24
Diseño del circuito electrónico de control para el encendido del motor de arranque del vehículo.....	25
Diagrama final del sistema para el encendido y rastreo automático vehicular.....	27
<i>SOFTWARE</i> .....	28
Captura y detección de rostro frontal con <i>OpenCV</i> .....	28
Reconocimiento facial.....	31
Librería <i>face_recognition</i> .....	34
Ubicación <i>GPS</i> .....	35
Notificaciones de alertas .....	35
<i>Switch</i> de inicio y <i>LED</i> indicador.....	37

SERVIDOR CENTRALIZADO.....	37
Página <i>web</i> .....	38
Conexión entre el módulo <i>Raspberry</i> y el servidor <i>Web</i> .....	39
Base de Datos MariaDB.....	40
API de <i>Google Maps</i> .....	43
3.2 CONSTRUCCIÓN DEL SISTEMA.....	43
Ruteo del Circuito.....	43
Medidas de la placa.....	44
Proceso de construcción de la placa.....	44
3.3 CONFIGURACIÓN PREVIA.....	47
Configuración previa en la <i>Raspberry</i> .....	47
Cargar el sistema operativo <i>Raspbian Stretch</i> en la SD.....	47
3.4 PRUEBAS DE FUNCIONAMIENTO.....	50
Prueba de Cámaras.....	50
Reconocimiento de Usuario.....	51
Notificaciones al correo electrónico.....	54
3.5 COSTOS DE IMPLEMENTACIÓN.....	56
4. <b>CONCLUSIONES Y RECOMENDACIONES</b> .....	57
4.1 CONCLUSIONES.....	57
4.2 RECOMENDACIONES.....	59
5. <b>REFERENCIAS BIBLIOGRÁFICAS</b> .....	61
Bibliografía.....	61
ANEXOS.....	66
ANEXO A: TEORÍA COMPLEMENTARIA.....	67
ANEXO B: MANUAL DE USUARIO.....	67
ANEXO C: MANUAL DE MANTENIMIENTO.....	67



## ÍNDICE DE FIGURAS

Figura 1.1	Estructura del módulo <i>Raspberry</i> .....	3
Figura 1.2	Pines GPIO de la tarjeta <i>Raspberry Pi 3 B+</i> .....	4
Figura 1.3	Cámara <i>Raspberry PiNoir V2</i> .....	6
Figura 1.4	<i>LCD touch screen display 3.5"</i> .....	7
Figura 1.5	Sistema de Posicionamiento Global.....	7
Figura 1.6	Módulo <i>GPS UBLOX NEO-6M</i> .....	8
Figura 1.7	Tarjeta SD de 32 GB clase 10.....	10
Figura 1.8	Logo del sistema operativo <i>Raspbian</i> .....	11
Figura 1.9	Reconocimiento facial en automóviles.....	14
Figura 1.10	Técnica de <i>LBP</i> .....	15
Figura 1.11	Histograma normalizado del <i>LBP</i> .....	16
Figura 3.1	Diagrama general del Sistema.....	19
Figura 3.2	Localización de los elementos del sistema en el habitáculo del vehículo.....	20
Figura 3.3	Conexión de la cámara <i>PiNoir V2</i> a la <i>Raspberry Pi</i> .....	21
Figura 3.4	Conexión del módulo <i>GPS</i> a la <i>Raspberry Pi</i> .....	22
Figura 3.5	Conexión de la <i>LCD touch screen display 3.5"</i> a la <i>Raspberry Pi</i> .....	22
Figura 3.6	Conexión de <i>Switch</i> y <i>led</i> indicador a la <i>Raspberry Pi</i> .....	23
Figura 3.7	Circuito electrónico de control para el encendido del motor.....	24
Figura 3.8	Diseño de la placa.....	25
Figura 3.9	Ruteo de la placa.....	26
Figura 3.10	Visualización de la placa en 3D.....	26
Figura 3.11	Diagrama completo de conexiones.....	27
Figura 3.12	Fotografía frontal del rostro.....	28
Figura 3.13	Fotografía en escala de grises y corte.....	29
Figura 3.14	Detección del rostro.....	30
Figura 3.15	Diagrama de flujo de la captura y detección frontal del rostro.....	31
Figura 3.16	Diagrama de flujo de reconocimiento facial.....	33
Figura 3.17	Escritorio de <i>Centos 7</i> .....	38
Figura 3.18	Dirección IP Estática.....	39
Figura 3.19	Conexión de la <i>Raspberry</i> con el Servidor <i>Web</i> .....	40
Figura 3.20	Identificación para el Ingreso a la Base de Datos.....	41
Figura 3.21	Base de Datos.....	41
Figura 3.22	Última ubicación registrada en la DB.....	42
Figura 3.23	<i>Key</i> de la <i>API</i> de <i>Google Maps</i> .....	43
Figura 3.24	Diseño y ruteo de la placa de control.....	44
Figura 3.25	Proceso de construcción de la placa.....	45
Figura 3.26	Cable conector entre switch de encendido y motor de arranque.....	46
Figura 3.27	Acoplamiento con el sistema de encendido.....	46
Figura 3.28	<i>Software SD Card Formatter</i> .....	47
Figura 3.29	<i>Software SD Card Formatter</i> finalizado.....	48
Figura 3.30	<i>Software Win32 Disk Imager</i> .....	48
Figura 3.31	Conexión remota mediante <i>PuTTY</i> .....	49
Figura 3.32	Interfaz del <i>SO Raspbian Stretch</i> .....	49
Figura 3.33	Cámara <i>PiNoir</i> (a) iluminación alta, (b) iluminación mínima.....	50
Figura 3.34	Escala de Grises.....	51
Figura 3.35	Bucle de reconocimiento facial.....	52
Figura 3.36	Usuarios (a), (b), (c) son desconocidos para el sistema.....	53
Figura 3.37	Reconocimiento exitoso (a) mínima iluminación, (b) alta iluminación.....	53
Figura 3.38	Recepción en los correos destinatarios.....	54

Figura 3.39 Ingreso a la aplicación. ....	55
Figura 3.40 Ubicación (a) normal, (b) con zoom. ....	55

## ÍNDICE DE TABLAS

Tabla 3.1 Clasificadores en cascada de <i>OpenCV</i> .....	34
Tabla 3.2 Costos del Proyecto. ....	56

## RESUMEN

El considerable incremento del robo de vehículos registrado en los últimos años en el país, mediante métodos de clonación de llaves y controles universales, pone a prueba la verdadera efectividad de las seguridades automotrices al momento de presentarse este tipo de delito. En consecuencia, el propietario piensa dos veces antes de dejar parqueado su vehículo en la calle.

Para afrontar esta problemática, se implementó un sistema de encendido automático vehicular mediante reconocimiento facial, basado principalmente en un microordenador de nueva generación, que se encarga principalmente de detectar y reconocer al verdadero propietario del vehículo, haciendo uso de la visión artificial a través de una cámara.

En primera instancia, se analizaron todos los componentes que conformarán el prototipo, empezando por el módulo central, una tarjeta *Raspberry pi 3 B+*. Además, este microordenador viene integrado un módulo *wifi*, *ethernet* y *bluetooth*. También cuenta con una entrada *HDMI*, *Pi camera V2*. Estas interfaces permiten la visualización y la obtención de imágenes; con la ayuda de la librería *OpenCV* y *Python 3*, se procede a realizar el reconocimiento facial usando el algoritmo *LBP (Local Binary Pattern)*.

Adicionalmente, un módulo GPS proporcionará la ubicación en tiempo real, con la ayuda de un aplicativo web ubicado en un laboratorio de la EPN. Complementariamente se enviará un correo electrónico de alerta, indicando la ubicación en la cual fue encendido el vehículo.

Finalmente, se han realizado todas las pruebas de funcionamiento, tanto en software como en hardware, demostrando el correcto funcionamiento de todo el sistema, adicionalmente se añade un manual de usuario y de mantenimiento.

## **ABSTRACT**

*The considerable increase in the theft of vehicles registered in recent years in the country, through cloning methods of keys and universal controls, test the true effectiveness of automotive securities at the time of this type of crime. Consequently, the owner thinks twice before leaving his vehicle parked on the street.*

*To address this problem, an automatic vehicular ignition system was implemented through facial recognition, based mainly on a new generation microcomputer, which is mainly responsible for detecting and recognizing the true owner of the vehicle, making use of artificial vision through a camera.*

*In first instance, all the components that make up the prototype were analyzed, starting with the central module, a Raspberry pi 3 B + card. In addition, this microcomputer comes integrated with a wifi, ethernet and bluetooth module. It also has an HDMI input, Pi camera V2. These interfaces allow the visualization and the obtaining of images; With the help of the OpenCV library and Python 3, facial recognition is carried out using the LBP (Local Binary Pattern) algorithm.*

*Additionally, a GPS module will provide the location in real time, with the help of a web application located in a laboratory of the EPN. In addition, an alert email will be sent, indicating the location where the vehicle was turned on.*

*Finally, all the functional tests have been carried out, both in software and in hardware, demonstrating the correct functioning of the entire system.*

# 1. INTRODUCCIÓN

Actualmente, la mayoría de sistemas de encendido vehicular presentes en el mercado automotriz, presentan una gran brecha de inseguridad; por tanto, muchas empresas automotrices, hoy en día suman esfuerzos para conseguir sistemas más robustos ante esta problemática. [1] [2]

Puesto que, el poseer un vehículo de última generación no garantiza que sus seguridades impidan el robo parcial o total del mismo, se ha evidenciado el desarrollo de nuevas tecnologías que impidan o dificulten acciones delictivas. La evolución de los sistemas de encendido para autos, ha pasado del simple hecho de tirar un hilo con una polea para encender motores, hasta el mismo hecho de encender un motor desde la comodidad de un escritorio a través de internet.

Todos los métodos de encendido existentes, actualmente son vulnerables. Cabe también resaltar que el último informe anual del Ministerio del Interior, presenta cifras alarmantes sobre el robo a vehículos particulares en todo el país. Se establece que en el año 2017 se presentaron 3715 robos de autos en el Ecuador. [3] Donde se pone en evidencia que ya no solo se trata de un problema social aislado; sino más bien, se trata de un problema social generalizado; dado que, en la actualidad, poseer un vehículo pasa de ser un lujo a conformar una herramienta esencial de trabajo.

Ante esta problemática, se propone implementar un sistema de encendido vehicular, usando las características faciales del propietario del automóvil. De este modo el vehículo se encenderá, únicamente reconociendo el rostro del propietario. Es decir, ya no se usará el encendido tradicional a través de la llave, sino que con una cámara digital se receptorán los rasgos faciales del propietario del vehículo, y una vez reconocido el rostro del mismo, el auto arrancará automáticamente sin necesidad de tener presionado ningún botón, ni ningún dispositivo electrónico adicional.

Adicionalmente, se propone la implementación de un sistema automatizado de rastreo satelital, que brindará no solo mayor seguridad, sino también la capacidad de ubicar el vehículo en cualquier lugar y a cualquier hora. El encendido automático vehicular usando reconocimiento facial, es innovador,

tiene un costo al alcance, y está dirigido a personas que tienen una conciencia sobre la seguridad de sus bienes. El uso de la llave para encender el vehículo, será cuestión del pasado.

Un grupo de investigadores publicó el trabajo titulado “Ciérrelo y piérdalo”, sobre la inseguridad de los sistemas de apertura y desbloqueo, mediante llaves electrónicas, con las que grandes marcas equipan sus vehículos; y los ha puesto en entredicho. En dicha publicación se analiza sobre la inseguridad de los sistemas de apertura en automóviles, mediante el uso de controles y llaves universales, permitiendo al delincuente llevarse el vehículo con mayor facilidad. [2] [4]

El reconocimiento facial, aparte de ser un sistema innovador, es incluso superior en seguridad a un detector de huellas. Hoy en día, es habitual el uso de este tipo de tecnologías en compañías que manejan bienes e información de un costo muy elevado. Con este sistema, no se corre el riesgo de olvidarse una clave o ingresarla incorrectamente, dado que, las técnicas de la biometría aprovechan el hecho de que las características del cuerpo humano son únicas y fijas. [4]

Para la ejecución de este sistema se procede a determinar todos los requerimientos necesarios para el encendido automático; seguido de esto, se diseñará e implementará un circuito electrónico de control, así como también se procederá a la configuración de la librería de visión artificial en *OpenCV*. A continuación, se procederá con el acoplamiento de todos los elementos que conformarán el sistema de reconocimiento facial para el encendido automático vehicular y rastreo satelital.

De este modo, se aprovechará todas las ventajas que brinda la fundación *Raspberry Pi*, al ser un sistema embebido que utiliza un lenguaje de alto nivel, y la gran versatilidad de acoplamiento en cualquier sistema operativo; al constituirse en un sistema de código abierto y con licencia GPL (*General Public License*). [5]

Finalmente, para el fácil manejo del sistema embebido, aplicado al encendido automático vehicular, se pone a disposición un manual de usuario y un manual de mantenimiento.

## 1.1 MARCO TEÓRICO

- **HARDWARE**

### **RASPBERRY PI 3 B+**

#### **Introducción**

La fundación Raspberry ha desarrollado un micro-ordenador, al alcance de todas las personas, revolucionando el sector de micro-ordenadores y servidores personales. De manera que; por un módico precio, se puede tener un dispositivo capaz de adaptarse a casi cualquiera de las necesidades del usuario. [6]

La construcción de esta tarjeta data desde el 2006, en la universidad de Cambridge, cuyos primeros modelos no cubrían todas las expectativas de los usuarios, tomando en consideración, que todos los diseños fueron estrictamente experimentales, pero por otro lado esto ayudó a completar y dar resultado a una placa verdaderamente eficiente y de un tamaño muy reducido. [7]

#### **Características**

La última versión de Raspberry Pi 3 B+, como se puede apreciar en la Figura 1.1, está soportada sobre un procesador de 1.4 GHz, con una arquitectura de 64 bits, y una memoria RAM de 1 GB. Cuenta también con un puerto CSI (*Camera Serial Interface*) que permite conectar una cámara nativa PiNoir. Además, del puerto DSI (*Display Serial Interface*), en el que se puede conectar una pantalla externa.



Figura 1.1 Estructura del módulo Raspberry. [8]



Adicionalmente, cuenta con 4 puertos USB 2.0, mismos que permiten la conexión de periféricos tales como: teclado, mouse, USB, *Web Cam*. Finalmente, el puerto HDMI (*High-Definition Multimedia Interface*), permite conectar una pantalla táctil (*touch screen display 3.5"*). [7] [8]

## Procesador

La placa *Raspberry Pi 3 B+*, viene integrada con un procesador BCM2837 Cortex-A53, el mismo que ofrece un alto nivel de eficiencia energética; lo que permite que el rendimiento del núcleo sea superior al del procesador Cortex-A7. Por otra parte, la arquitectura ARMv8, incluye una serie de nuevas características. Esto incluye procesamiento de datos de 64 bits, como también soluciona el direccionamiento virtual extendido, y los registros de propósito general de 64 bits. [9]

## Puertos GPIO

El propósito de los pines GPIO (*General Purpose Input/Output*), es enlazar el software con el mundo exterior. Todo depende del aplicativo específico que el usuario desee realizar. Estos pines permiten desarrollar tareas, como el simple hecho de rotar un motor, hasta establecer una comunicación directa usando I2C, a través de un bus, hacia una computadora. Estos pines GPIO se pueden apreciar en la Figura 1.2. [10]

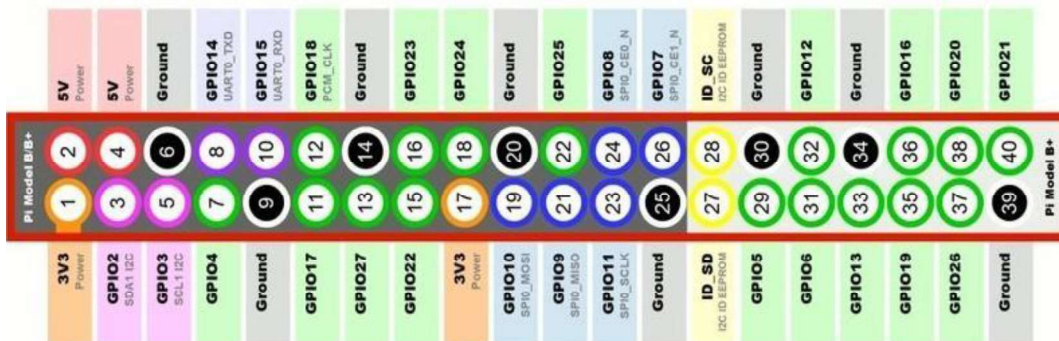


Figura 1.2 Pines GPIO de la tarjeta Raspberry Pi 3 B+. [11]

### **Puertos USB**

El modelo *Raspberry Pi 3 B+*, contiene 4 puertos USB 2.0, al igual que su antecesor *Raspberry Pi 3*. En este aspecto, no ha mejorado en su capacidad ni en su velocidad, por lo que se mantiene igual. [7] [8]

### **Puerto Ethernet**

Es un gran avance que se ha conseguido con la tarjeta *Raspberry Pi 3 B+*, ya que ha aumentado la velocidad de 100 Mbps a 300 Mbps. Tomado en consideración que se trata de 1 Gb *Ethernet*, pero por el cuello de botella que se forma, no permite rendir al 100%. [7] [8]

### **Audio/MIC**

La tarjeta *Raspberry Pi 3 B+*, al igual que algunas de sus versiones anteriores, posee una salida analógica de tipo mini Jack de 3.5 mm. [8]

### **Puerto CSI**

Permite la conexión de una cámara externa, nativa de la familia *Raspberry*, *RaspiCam* para el día y *PiNoir* para la noche. [11]

### **Puerto HDMI**

Este puerto permite la conexión de una pantalla externa con alta definición, o cualquier monitor externo que sea compatible con la *Raspberry Pi 3 B+*. [8] [11]

### **Puerto PWR**

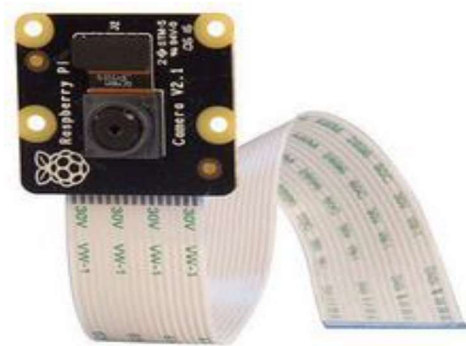
La energización que posee la tarjeta *Raspberry Pi 3 B+*, se puede administrar mediante un cable USB, o a su vez mediante PoE (*Power over Ethernet*). Siempre que los valores de voltaje oscilen entre 3.3 VDC y como mínimo 2.5 Amperios. [7]

## **Puerto DSI**

Este puerto es otra de las ventajas que nos brinda la familia *Raspberry*, ya que, adicionalmente del puerto HDMI y de la conexión remota, también se puede conectar una pantalla, a través de un bus de datos directamente a la tarjeta. [11]

## **Cámara *Raspberry PiNoir V2***

Esta cámara *Raspberry PiNoir V2* (Figura 1.3) corresponde a la versión de visión nocturna. Contiene un sensor de imagen de 8 megapíxeles con lente de enfoque fijo, y se conecta a la *Raspberry Pi* a través de un cable plano al conector CSI (*Camera Serial Interface*). Gracias a esta interfaz, se garantiza la transferencia rápida y eficaz de imágenes, en comparación con una webcam convencional. [13]



*Figura 1.3 Cámara Raspberry PiNoir V2. [13]*

Características principales de la Cámara Raspberry Pi V2:

- Resolución de 8 megapíxeles.
- Sin filtro infrarrojo excelente para condiciones de poca luz.
- Incluye filtro azul.
- Modos de video 1080p60, 720p60, y 640 x 480p60/90.
- Sensor Sony IMX219.
- Resolución del sensor 3280 x 2464 píxeles.

### **LCD TOUCH SCREEN DISPLAY 3.5"**

La *LCD touch screen display 3.5"*, (Figura 1.4), proporciona una manera fácil de visualizar la información que viene de la *Raspberry Pi*. Se conecta a través del puerto GPIO (*General Purpose Input/Output*), y HDMI (*High Definition Multimedia Interface*). [13]



Figura 1.4 *LCD touch screen display 3.5"*. [13]

Principales características de la *LCD touch screen display 3.5"*:

- LCD retro iluminada.
- Resolución de 480x320 pixeles.
- Soporta cualquier versión de *Raspberry Pi*.

### **GPS**

GPS (*Global Positioning System*) está formado por una red de 24 satélites que fueron colocados en una órbita por el Departamento de Defensa de Estados Unidos. Como se puede observar en la Figura 1.5. Aunque en sus inicios fue usado para aplicaciones militares, el gobierno hizo que el sistema esté disponible para uso civil en la década de 1980. [14]

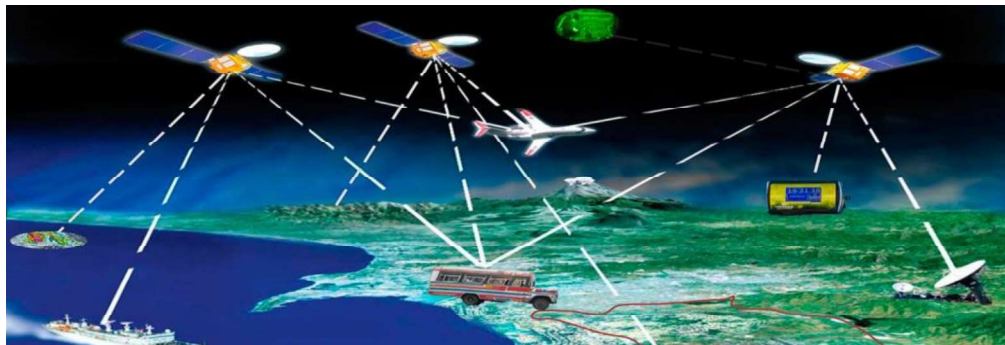


Figura 1.5 *Sistema de Posicionamiento Global*. [14]

## Funcionamiento del GPS

Los receptores GPS toman la información enviada por los satélites GPS, y utilizan la trilateración para calcular la ubicación exacta del usuario.

Los 5 pasos principales para el funcionamiento del sistema GPS:

1. Triangulación de los satélites.
2. El receptor GPS mide distancias, utilizando el tiempo de viaje de las señales de radio. Se utiliza la velocidad de la luz (300.000.000 m/s), la cual define la velocidad con la que viajan las ondas de radio a través del espacio y así se puede conocer dichas distancias.
3. Para medir el tiempo de viaje de estas señales, el GPS necesita un control muy estricto de tiempo.
4. El GPS necesita conocer exactamente en dónde se encuentran los satélites en el espacio.
5. Finalmente, el GPS corrige cualquier demora en el tiempo de viaje de la señal que esta puede sufrir mientras atraviesa la atmósfera.

## Receptor GPS UBLOX NEO-6M

El módulo GPS UBLOX NEO-6M (Figura 1.6) es un receptor GPS de buen rendimiento con una antena cerámica. Se comunica a través de puerto serial UART(*Universal Asynchronouns Receiver Transmitter*), mide parámetros de latitud, longitud, altura y velocidad desde cualquier punto de la Tierra donde se encuentre. [15]

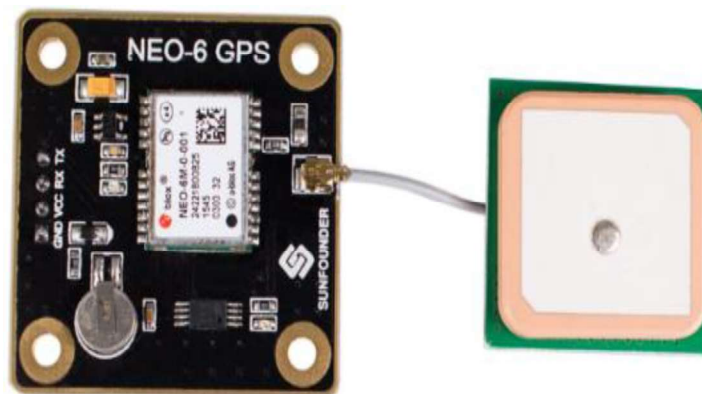


Figura 1.6 Módulo GPS UBLOX NEO-6M. [15]

## Principales características del módulo GPS UBLOX NEO-6M:

- Voltaje de alimentación (3.3 - 5) VDC.
- Led indicador de señal.
- Sistema de coordenadas: WGS-84.
- Sensibilidad de captura -148 dBm.
- Sensibilidad de rastreo: -161 dBm.
- Máxima altura medible: 18000 m.
- Máxima velocidad 515 m/s.
- Exactitud: 1micro segundo.
- Frecuencia receptora: L1 (1575.42 Mhz).
- Tiempo de inicio primera vez: 38s en promedio.
- Tiempo de inicio: 35s en promedio.

### **Diodo LED**

El diodo LED (*Light Emitting Diode*) es un dispositivo electrónico semiconductor que emite luz de espectro reducido, cuando es polarizada directamente la unión PN (positivo, negativo). [16]

### **Resistor**

El resistor o resistencia es un elemento pasivo que produce la oposición que ofrece un conductor al paso de la corriente eléctrica, causando en sus terminales una diferencia de voltaje. [17]

### **Interruptor eléctrico o Switch**

El interruptor eléctrico o Switch es un dispositivo que permite desviar o interrumpir el curso de una corriente eléctrica, tiene dos posiciones: encendido y apagado. [18]

### **Relé**

El relé es un dispositivo electromagnético, el cual funciona como un interruptor, que es controlado por un circuito eléctrico. Está compuesto por una bobina y un

electroimán. Se acciona uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes. Voltaje de trabajo 12 V y corriente de relé 25 mA. [19]

#### **Transistor 2N3904**

El 2N3904 es un transistor de conmutación rápida, es un dispositivo electrónico semiconductor utilizado para entregar una señal de salida en una respuesta a una señal de entrada.

Este transistor es de tipo NPN,  $V_{max}= 40V$ ,  $I_{max}= 200mA$  y HFE (*Forward Current Gain*)  $min= 50$ . [20]

#### **Diodo 1N4007**

El diodo rectificador, se usa principalmente para convertir la corriente alterna en directa.

#### **Tarjeta Micro-SD**

Es el dispositivo en el cual, se albergan todos los programas, incluido el sistema operativo, que permiten el correcto funcionamiento de la tarjeta *Raspberry Pi 3 B+*. Para esta versión se recomienda como mínimo, una capacidad de 32 GB (Figura 1.7) y clase 10. [21]

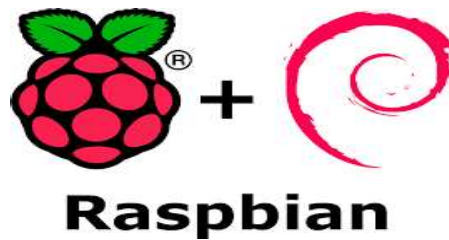


Figura 1.7 Tarjeta SD de 32 GB clase 10. [24]

- **SOFTWARE**

### ***Raspbian***

Raspbian es una versión *Debian Linux* adaptada, Figura 1.8; esta herramienta que brinda la familia *Raspberry*, abre un mundo de posibilidades, si al momento de programar se trata. Este sistema operativo es totalmente libre, ya que está basado en una arquitectura Debian. Además, *Raspbian* es compatible con todas las versiones anteriores lanzadas al mercado. [22]



*Figura 1.8 Logo del sistema operativo Raspbian. [22]*

### ***OpenCV***

Esta librería de visión por computadora y de código abierto, OpenCV (*Open Source Computer Vision Library*), permite el aprendizaje automático de máquinas. La librería *OpenCV* cuenta con más de 2500 algoritmos totalmente optimizados. Esta tecnología de última generación, puede detectar y reconocer rostros, identificación de objetos, clasificar acciones humanas en video, rastrear objetos en movimiento, extraer modelos 3D de objetos.

Adicionalmente, encuentra imágenes similares de una base de datos, previo entrenamiento y almacenamiento en dicha base de datos. [23] [24]

### ***Python 3***

*Python* es un lenguaje de propósito general e interpretado, lo que se entiende, que no requiere compilación previa del código para su ejecución. Adicionalmente, es un lenguaje multiplataforma ya que permite la ejecución del código en distintas plataformas y sistemas operativos, sin la necesidad de cambiar el código fuente. Finalmente, por ser uno de los lenguajes de fácil



aprendizaje e interpretación, es usado en empresas de gran prestigio como es el caso de la NASA. [25]

## **VNC**

VNC (*Virtual Network Computing*), es un sistema gráfico multiplataforma de uso compartido, lo que permite controlar de una forma remota la interfaz del escritorio de la *Raspberry*. Esto quiere decir, poder observar el escritorio de la tarjeta *Raspberry*, en la ventana de algún dispositivo móvil, o a su vez, visualizar en la pantalla de una computadora. Esta comunicación remota se ejecuta a través del protocolo de comunicación SSH (*Secure Shell*), mediante la arquitectura cliente-servidor. [26] [27]

## **Centos 7**

Al ejecutar *Linux*, queda demostrado que *Centos 7*, es el sistema operativo más popular, superior, eficiente, ligero y fiable, si al momento de manejar y controlar un servidor *Web* se trata. Adicionalmente, *Centos 7* fue diseñado como un proyecto de código abierto gratuito de nivel empresarial, con las mismas capacidades de rendimiento, estabilidad y fiabilidad que el sistema operativo de pago *Redhat Enterprise Linux (RHEL)*. [28] [29]

## **Servidor LAMP**

LAMP (*Linux, Apache, MySQL y PHP*), la unión de estas cuatro herramientas esenciales y totalmente independientes, permite el uso y existencia de aplicaciones web, ya sea de tipo estáticas o dinámicas. Estas páginas web, van desde una página básica con características limitadas, hasta sistemas integrados de video en *streaming*. El apogeo de esta infraestructura tecnológica, radica en el hecho de que, el software utilizado es de uso libre y gratuito.

Sin tomar en cuenta todas las ventajas de operatividad, soporte, fiabilidad, adaptabilidad y múltiples características, que hacen posible que este sistema sea utilizado en un alto porcentaje a nivel empresarial. [30] [31]

Por otra parte, el resultado de poseer un hardware adecuado, con prestaciones tecnológicas de punta, y herramientas informáticas de uso libre, como es el caso del servidor LAMP, hace posible que la administración de la App esté controlada totalmente por su administrador. No es necesario contratar un Hosting privado, para que los usuarios puedan acceder al aplicativo; lo puede hacer de forma simple, con tan solo un DNS (*Domain Name System*), previamente configurado. [32]

### ***phpMyAdmin***

Esta herramienta de *software* libre escrita en PHP, permite la administración de base de datos, tablas, columnas, relaciones, índices, usuarios, permisos, etc. Este instrumento tecnológico, fue diseñado con el propósito de gestionar MySQL y MariaDB a través de la web. De modo que, el usuario haciendo uso de una interfaz, tenga la capacidad de ejecutar cualquier instrucción de SQL (*Structured Query Language*). [33]

### ***HTML***

El lenguaje de programación HTML (*HyperText Markup Language*), dedicado exclusivamente en el diseño de páginas web. Este lenguaje se complementa con las hojas de estilo en cascada CSS (*cascading style sheets*). La evolución de este tipo de marcado y presentación de páginas web, claramente facilitan el trabajo, en cuanto a la apariencia de las páginas web se refiere. Dando como resultado, páginas interactivas, dinámicas y sobretodo amigables con los usuarios que visualizarán su contenido. Y lo más sobresaliente es que todos estos estilos son totalmente gratuitos. [34]

### ***Java script***

Es un lenguaje de programación encaminado principalmente al diseño de sitios web. Lo más sobresaliente de este lenguaje, es que no necesita de compilación ya que se trata de un lenguaje interpretado. Esto es realizado por los navegadores web, y funciona del lado del cliente, más no del lado del servidor;

permitiendo de esta manera manejar grandes números de usuarios en línea, eliminando la posibilidad de colapsar la página web. [35]

### **IP Pública**

Una dirección IPv4 pública está conformada por 4 octetos, separados por puntos y representada en números decimales. Permite que un ordenador con una IP pública, sea visible (accesible) desde cualquier otro ordenador conectado a internet. [36]

### **DNS**

El servidor DNS (*Domain Name System*), resuelve la petición que hace un usuario a un nombre de dominio, para transformarla en una dirección IP, correspondiente a un servidor web. [37]

- **SISTEMAS DE RECONOCIMIENTO FACIAL EN AUTOMÓVILES**

En la actualidad, no existen muchos vehículos con esta tecnología; sin embargo, grandes corporaciones de marcas de renombre, como es el caso de Ford, Toyota, Citroën, Volvo etc., han empezado el desafío de implantar sistemas totalmente digitales y de control. Dichos sistemas se encargan del monitoreo y respuesta hacia posibles indicios de riesgo de inseguridad, al momento de conducir un automóvil. Como se puede observar en la Figura 1.9.



*Figura 1.9 Reconocimiento facial en automóviles. [44]*

Como es el caso de Ford, con el proyecto Mobbi (*Mobile Interior Imaging*), este sistema trata de integrar microprocesadores, que controlen múltiples funciones del conductor. A través de movimientos de los ojos, subir o bajar el volumen del estéreo, abrir o cerrar ventanas con tan solo un movimiento de la cabeza. Así como también detectar fatiga, por medio de las expresiones faciales que presente el conductor, y enviar una advertencia de que se está quedando dormido, e incluso que el vehículo se detenga por completo si detecta indicios que el conductor está a punto de quedarse dormido. [39] [38]

### Algoritmo LBP

Esta técnica de operador de textura altamente efectiva, proporciona una clasificación de objetos dentro de la visión por computador. Su propósito es analizar los píxeles de una imagen; mediante la comparación de estos píxeles, se logra obtener una serie de números binarios que se transformarán en números decimales, cuyo resultado será un vector o histograma único para cada rostro. [40]

Para el análisis de las texturas, mediante la división de pequeños cuadrados en la imagen, se procesan solo los rasgos más representativos del rostro que se muestra en la Figura 1.10, seguido de un resultado de números binarios, los cuales terminan en un número ponderado, representado en un número decimal que va de 0 a 255.

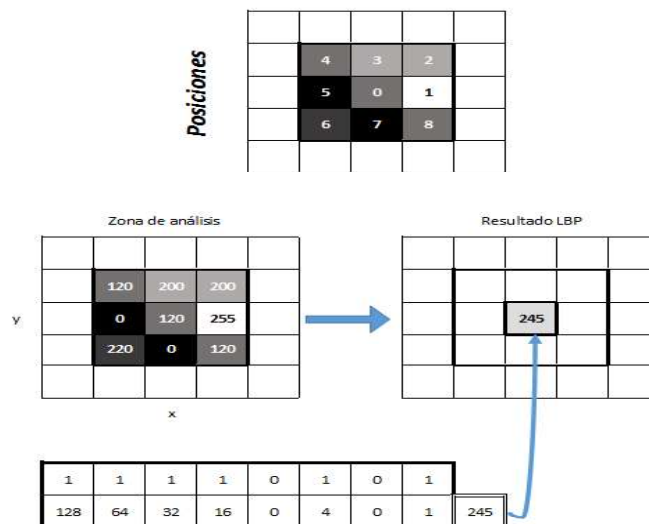


Figura 1.10 Técnica de LBP. [41]

Esta técnica de reconocimiento de objetos y de rostros ha sido una de las más efectivas, gracias a la colaboración de múltiples investigadores, donde concluyen, que el algoritmo con mejores prestaciones es el LBP (*Local Binary Patterns*). Opera en números enteros; por lo tanto, la carga del procesamiento computacional es inferior a otros algoritmos, pues el tiempo empleado para realizar dicho análisis, es un par de veces menor que otros métodos, los cuales también se encuentran en la librería de *OpenCV*. [41]

Finalmente, se puede observar el resultado de la codificación de la relación del pixel central con los pixeles aledaños vecinos, representado en un histograma, como se puede apreciar en la Figura 1.11.

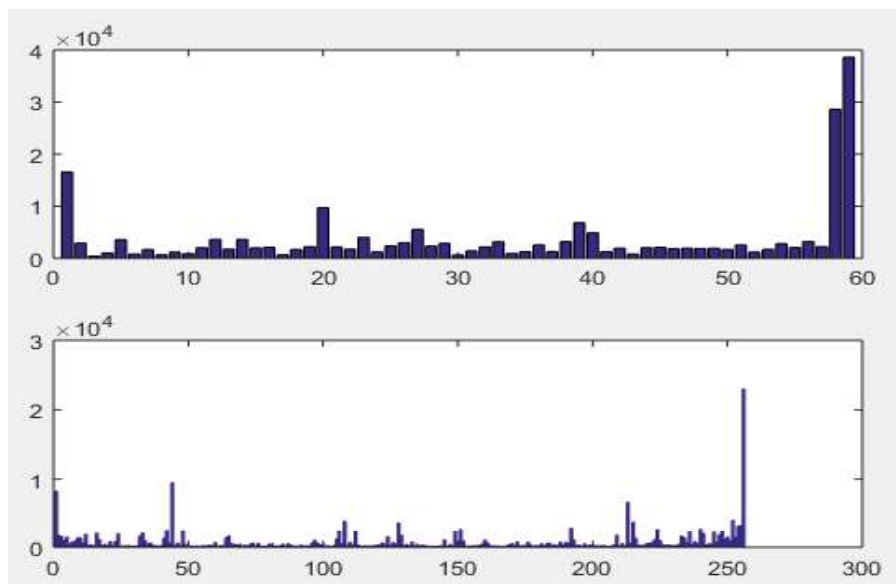


Figura 1.11 Histograma normalizado del LBP. [41]

## 2. METODOLOGÍA

### 2.1 MÉTODO COMPARATIVO

Dado que, el sistema de reconocimiento facial y rastreo satelital, requiere de un alto procesamiento de datos, como también altas velocidades de conexión a internet para manejar aplicaciones a tiempo real, se requiere evaluar diferentes sistemas embebidos, que permitan cumplir con el objetivo planteado

inicialmente. Por consiguiente, la tarjeta *Raspberry Pi 3 B+*, ofrece todas las características necesarias, en comparación con otras tarjetas de desarrollo que presenta el mercado actual.

## **2.2 MÉTODO ANALÍTICO**

Mediante este método se pudo examinar, conocer, aprender e interpretar todos los componentes elementales del *hardware*, que componen la tarjeta *Raspberry Pi 3 B+*. Se analizó su arquitectura, funcionalidad y adaptabilidad con diferentes módulos externos que conforman todo el sistema, brindando un efectivo funcionamiento.

Por otro lado, dicho método permitirá identificar el *software* adecuado, como es el caso del *Python 3*, el mismo que faculta la integración de las diferentes librerías, que se emplearán para realizar el procesamiento de reconocimiento facial y rastreo satelital.

## **2.3 MÉTODO EXPERIMENTAL**

El método experimental se utilizó para elaborar el control de encendido del vehículo, por medio de técnicas de identificación biométricas, las cuales garantizan la no vulnerabilidad de la seguridad al momento de encender el vehículo.

Este método permitió el empleo de repositorios en línea, que contienen proyectos de código abierto, en los cuales, se permite usar el código fuente. Gracias a la colaboración de comunidades de software libre, se facilita la reutilización de dichos proyectos, para dar paso al desarrollo de nuevas iniciativas.

Para la implementación de este sistema de encendido vehicular mediante el reconocimiento facial, se hizo uso de un módulo *Raspberry Pi 3 B+*; el mismo que posee interfaces que permitirán conectarse a una cámara *RaspiCam*. Esta cámara cuenta con las características necesarias para realizar dicho reconocimiento facial. Adicionalmente, para brindar las garantías de seguridad al vehículo, se colocó un módulo GPS, que determina la ubicación del automóvil. Finalmente, se envía la ubicación del automóvil al correo electrónico del usuario.

Además, se diseñó un circuito electrónico general en *Proteus Design Suite* basado en relés fundamentalmente. En dicho circuito se adecuará elementos de protección, contra sobrecargas eléctricas, generadas por el motor de arranque del vehículo.

Se configuró la tarjeta *Raspberry Pi 3 B+*, usando las diferentes librerías disponibles en el sistema operativo Linux, al tratarse de un sistema operativo de código abierto permite la modificación en el código fuente. Dicho código permitió el acondicionamiento del sistema en base a los requerimientos necesarios, para controlar la cámara *RaspiCam* en pequeños segmentos de tiempo, de esta manera se tiene una mejor resolución, y por tanto mayor efectividad al momento de realizar el escaneo del rostro del propietario del vehículo.

Se adaptó el sistema de encendido automático a un vehículo. Para ello se determinó, una ubicación estratégica en el habitáculo del automóvil para los módulos *Raspberry Pi3* y *GPS*, los mismos que cuentan con una fuente de alimentación regulada, para prevenir sobrecargas eléctricas. Así como también se procedió al acoplamiento de dichos módulos con el sistema electrónico de arranque del vehículo.

Posteriormente, se realizaron las pruebas del funcionamiento del sistema implementado en el vehículo. Cabe recalcar que las pruebas de funcionamiento se realizaron tomando en cuenta los siguientes aspectos: prueba de cables y conectores, prueba de conexiones, prueba de fijación de todo el módulo. Además, pruebas de funcionamiento del módulo GPS.

También se comprobó la configuración y funcionamiento del sistema operativo *Raspbian*; posteriormente, se procedió a la verificación del funcionamiento del reconocimiento facial por medio de la cámara *RaspiCam* y la tarjeta *Raspberry Pi 3 B+*. En general, se realizó una prueba de operatividad de la placa base y la alimentación. Además, se incluye un manual de usuario para el correcto funcionamiento y mantenimiento del sistema de reconocimiento facial. Finalmente, se presentó un presupuesto referencial del proyecto.

### 3. RESULTADOS Y DISCUSIÓN

#### 3.1 DISEÑO DEL SISTEMA

Para el desarrollo del presente sistema, se detallan dos partes, la primera corresponde al Hardware utilizado en el proyecto, la segunda parte se detalla el software que fue empleado.

Adicionalmente se detalla las conexiones, modificaciones realizadas en el software, como también el uso de cada librería y su funcionamiento. En la Figura 3.1, se presenta un diagrama general de todo el sistema.

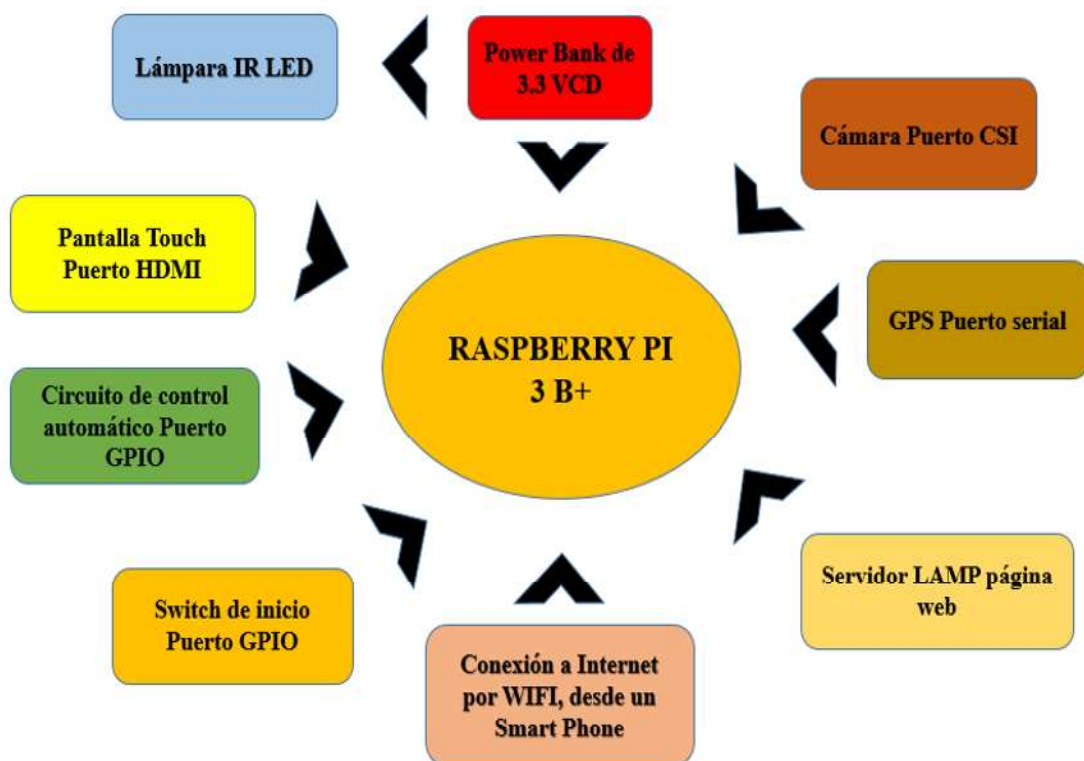


Figura 3.1 Diagrama general del Sistema.

#### HARDWARE

Para el diseño del prototipo de encendido automático vehicular, se han tomado en cuenta varios aspectos importantes a considerar en el habitáculo del vehículo, mismos que se detallarán en el transcurso del desarrollo del sistema. A continuación, se presenta la localización general de cada elemento que



conforma el sistema final ya acoplado en el automóvil, como se puede observar en la Figura 3.2.

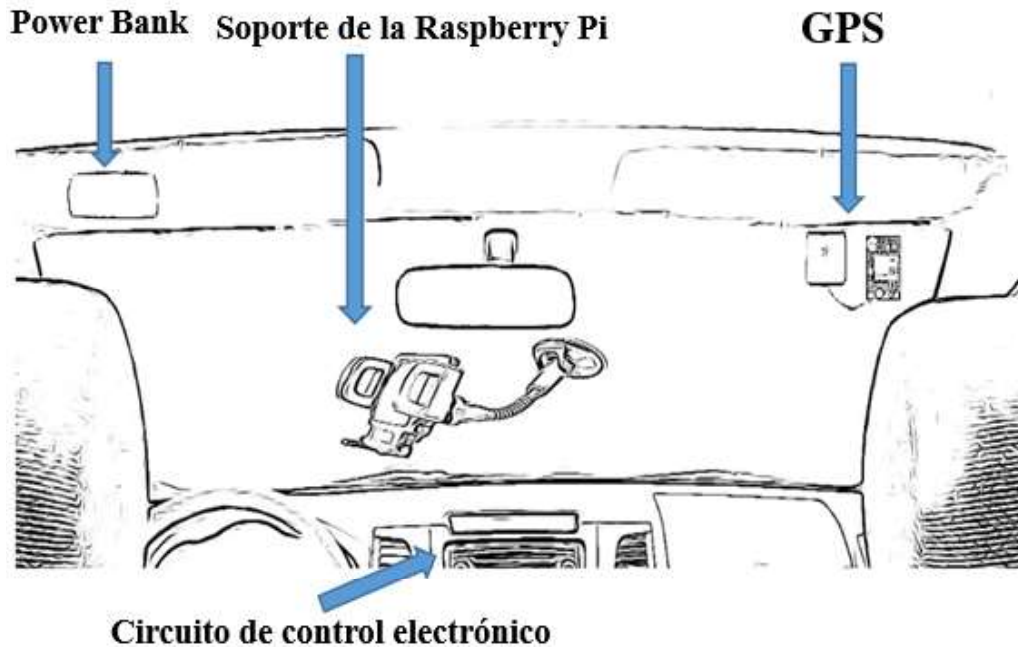


Figura 3.2 Localización de los elementos del sistema en el habitáculo del vehículo.

- **RASPBERRY PI 3 B+**

Este módulo, es el elemento principal encargado de todos los procesos computacionales; además, este dispositivo es de donde convergen absolutamente todos los periféricos empleados en el presente sistema. Las especificaciones técnicas del módulo Raspberry Pi se detallan en el Anexo A.

- **Conexión de la cámara PiNoir V2 a la Raspberry Pi**

En esta etapa es importante recalcar, el uso de tres tipos de cámaras: Webcam estándar, RaspiCam y PiNoir, las dos últimas nativas de Raspberry. La Primera cámara no garantiza ningún grado de seguridad, por la baja resolución al momento de reconocer las características faciales, el nivel de error es muy elevado.

Por otra parte, la cámara *RaspiCam* puede usarse en el día, pero en la noche o en la oscuridad absoluta esta cámara es totalmente ineficaz.

La cámara *PiNoir* garantiza una excelente recepción de captura de imagen y video en el día y en la noche, gracias a la rapidez y compatibilidad con el módulo Raspberry Pi 3 B+, permite que sus 8 mega pixeles proporcione un video de alta calidad.

La comunicación entre el procesador y la cámara se realiza mediante el protocolo de comunicación I2C (*Inter-Integrated Circuit*). Para conectar la cámara a la *Raspberry Pi* se debe encontrar el conector CSI (*Camera Serial Interface*) en la PCB, que a través de un cable plano (Bus de datos) se conecta a la cámara, como se ve en la Figura 3.3.

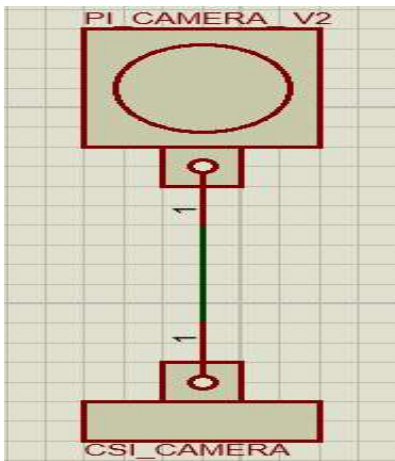


Figura 3.3 Conexión de la cámara *PiNoir V2* a la *Raspberry Pi*.

- **Conexión del Receptor *GPS UBLOX NEO-6M* en la *Raspberry Pi***

Para la conexión del *GPS* con la *Raspberry* se utilizó un cable par trenzado sin blindaje categoría 5e, que soporta una frecuencia de hasta 100 Mhz. El tiempo de inicio de Receptor *GPS* es de 38 segundos, una vez energizado la *Raspberry*. El módulo *GPS* recibe y envía datos por medio de la comunicación *UART* (*Universal Asynchronouns Receiver Transmitter*), este módulo utiliza los pines de transmisor y receptor de la *Raspberry Pi*.

Para la energización de módulo es importante tener una conexión a tierra común centralizada; es decir, que esté conectada directamente al Chasis del vehículo

para evitar errores de las señales recibidas por el GPS. El diagrama de conexión se puede apreciar en la Figura 3.4.

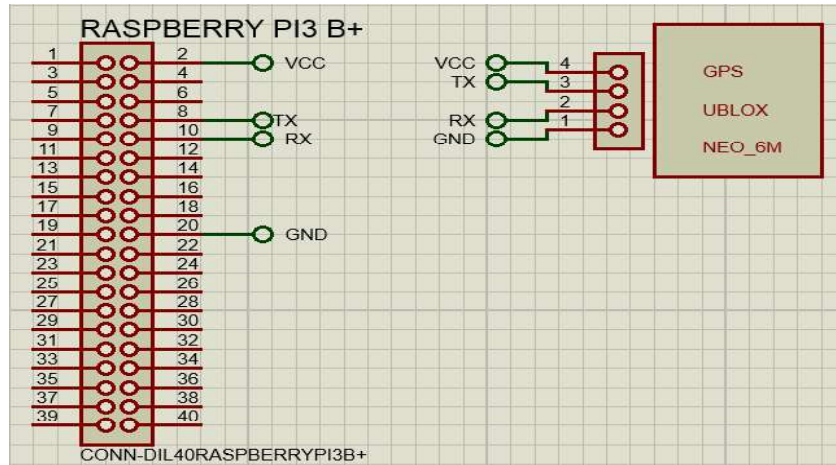


Figura 3.4 Conexión del módulo GPS a la Raspberry Pi.

- **Conexión de la LCD touch screen display 3.5" a la Raspberry Pi**

Este dispositivo, permite la interacción directa entre el conductor y el sistema computacional de encendido automático. Por otra parte, la cámara PiNoir tiene el filtro infrarrojo eliminado, esto hace que, se tenga una mala interpretación de color en el día; por tanto, no se puedan visualizar las diferentes tonalidades de color en el día. Pero esto es irrelevante al momento del procesamiento de la imagen, ya que todo el programa trabaja en escala de grises. Figura 3.5.

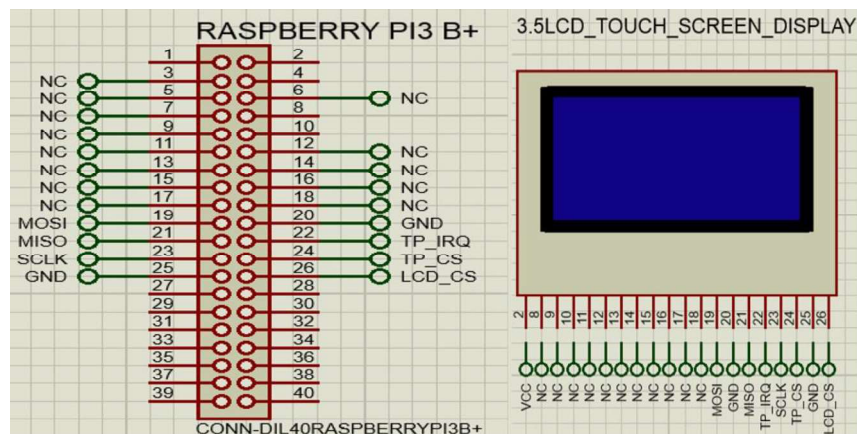


Figura 3.5 Conexión de la LCD touch screen display 3.5" a la Raspberry Pi.

Es importante alinear y balancear la cámara *PiNoir* con respecto a la pantalla *Touch*, para tener un mejor enfoque del conductor; por otra parte, esto brindará confort al usuario al momento de la identificación. Además, es primordial cuidar la integridad física de la pantalla *touch*, con la ayuda de disipadores de calor. La *LCD*, se conecta a la *Raspberry Pi* a través del puerto *GPIO* y *HDMI*.

- **Conexión de un *Switch* y *LED* indicador a la *Raspberry Pi***

El *LED* indicador conectado al puerto *GPIO* 12 (pin 32) de la *Raspberry*, se encenderá cuando el usuario haya sido reconocido exitosamente. Por otra parte, el *Switch* conectado al puerto *GPIO* 16 (pin 36), es el único elemento que permite la habilitación total del sistema, de esta manera se enviará un estado de “1” lógico, previo al reconocimiento satisfactorio del conductor. Posteriormente, se activará el arranque del vehículo, seguido de esto, se encenderá el motor del vehículo. Esta conexión se observa en la Figura 3.6.

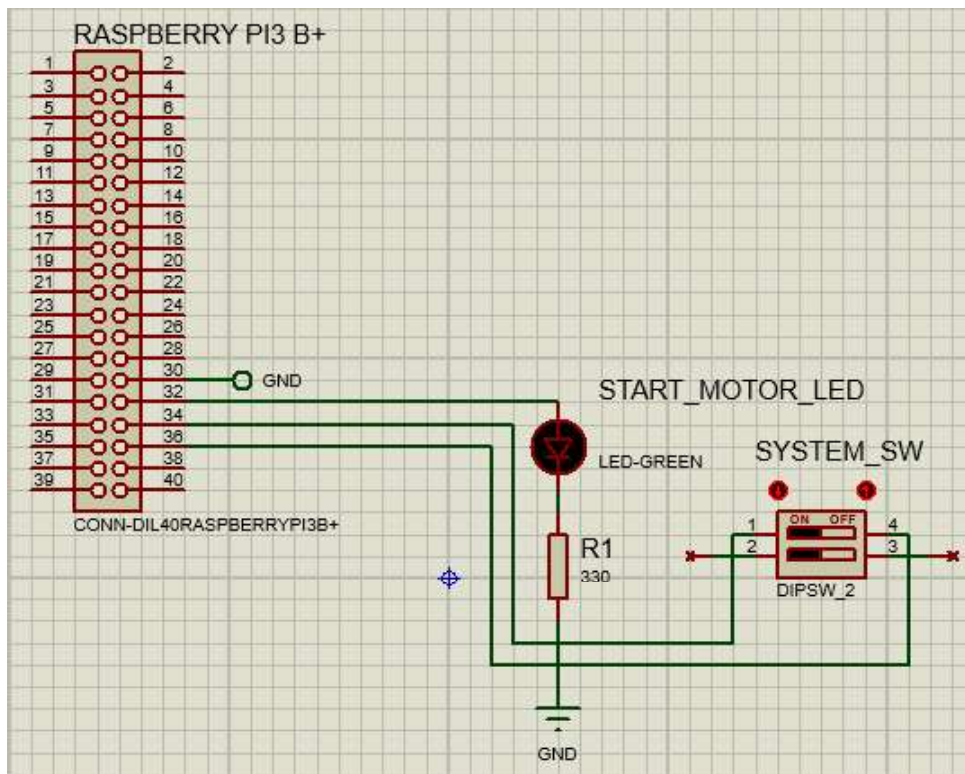


Figura 3.6 Conexión de Switch y led indicador a la Raspberry Pi.

Para calcular el valor del resistor utilizado, se ha considerado los siguientes criterios: en las salidas de los puertos GPIO existe 5 VDC, la corriente consumida por los leds es de 20 mA, entonces:  $5V/20mA=250 \Omega$ . Un valor de resistor físico real, cercano al calculado ( $250 \Omega$ ) es de  $330 \Omega$ .

- **Conexión de la Raspberry Pi al motor de arranque del vehículo**

Se consideró el diseño de un circuito electrónico de control, cuyo elemento principal es un transistor 2N3904, que está conectado directamente a dos relés en serie, como se muestra en la Figura 3.7. Este módulo de doble relé, a su vez se encargará de encender el motor de arranque del vehículo.

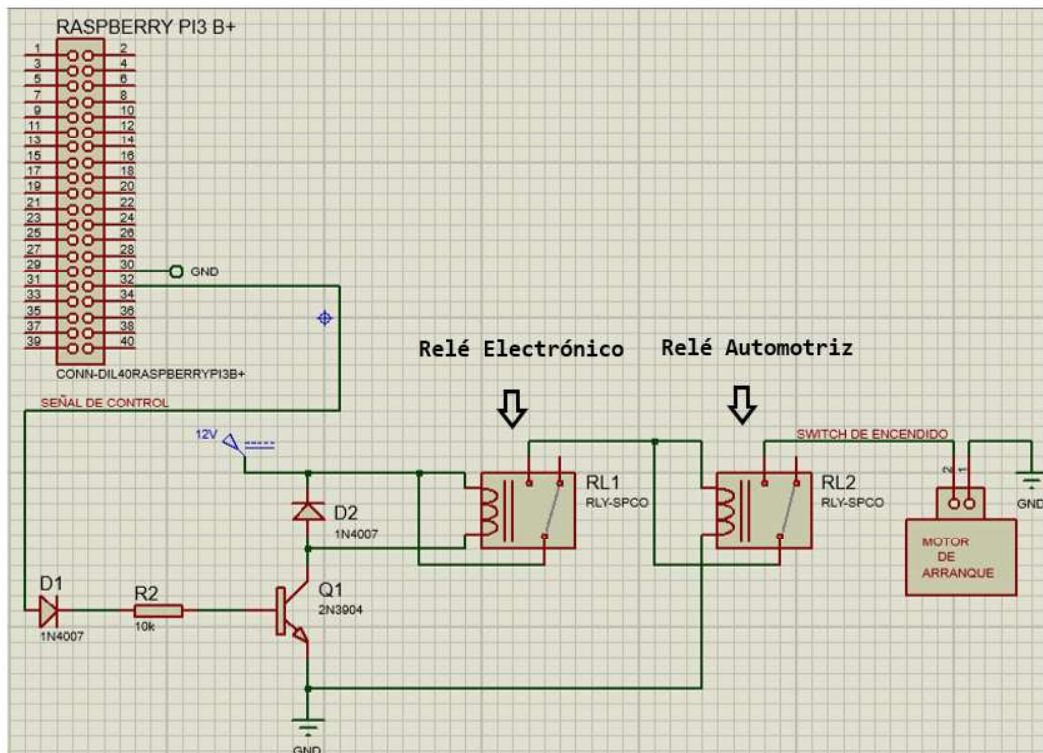


Figura 3.7 Circuito electrónico de control para el encendido del motor.

El puerto GPIO 12 (pin 32) envía 5 VDC a su salida, siempre y cuando el reconocimiento facial del usuario haya sido exitoso. Cuando la salida del puerto GPIO 12 se encuentre a 5 VDC, provocará que se supere la tensión de umbral de la base del transistor 2N 3904 (0.6V) y, por lo tanto, circulará una corriente entre base y masa, dicha corriente hará que el transistor se encuentre en estado

de conducción (entre colector y emisor), cerrando el circuito de la bobina del relé; finalmente, activará de forma automática a dicho relé.

En primera instancia se probó solo con el Relé Automotriz, lo que ocasionó que el transistor se dañara en reiteradas ocasiones, la solución para este inconveniente fue usar un Relé electrónico de 12 VDC intermedio que active al Relé Automotriz de 40 A.

Al desactivar el relé, se interrumpe el paso de corriente a través de la bobina. El campo magnético presente, se induce por un periodo muy corto en dicha bobina, lo que desencadena una tensión muy elevada de polaridad opuesta en sus terminales. Este pico de tensión puede dañar el transistor. La solución más simple es conectar en paralelo un diodo rectificador 1N4007, inversamente polarizado, de tal modo que absorba estos picos de tensión de polaridad opuesta.

Cálculo de la resistencia de base: 
$$R = \frac{(V_{in} - 0.6) HFE}{I_{relé}} = \frac{(5V - 0.6) 50}{0.025A} = 8800 \Omega$$

Un valor de resistor físico real, cercano al calculado (8800  $\Omega$ ) es de 10k $\Omega$ .

- **Diseño del circuito electrónico de control para el encendido del motor de arranque del vehículo.**

Para el acoplamiento entre la Raspberry Pi 3 B+, y el motor del arranque del vehículo, se ha diseñado una tarjeta PCB, usando el software Proteus 8 Profesional (ISIS), como se presenta en la Figura 3.8.

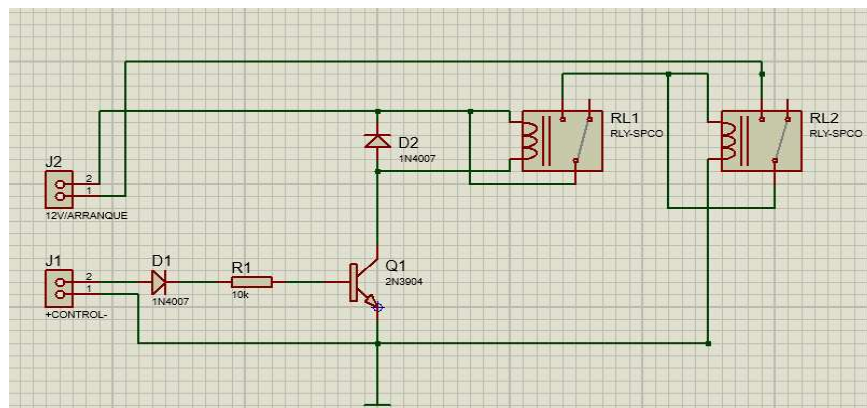
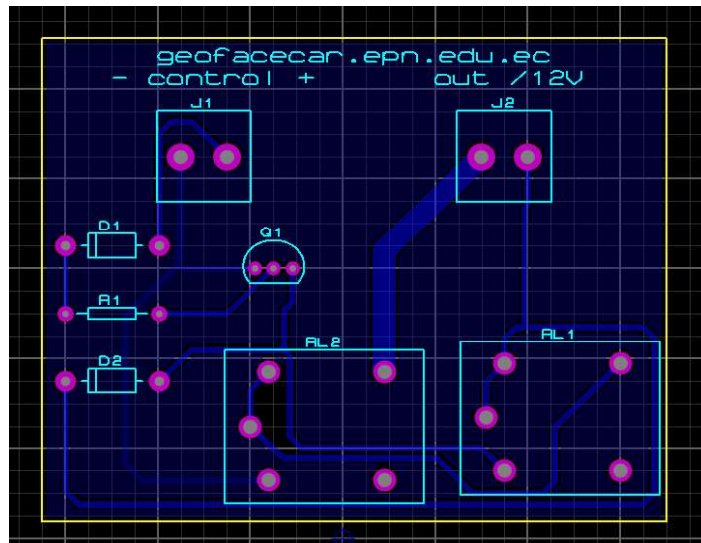


Figura 3.8 Diseño de la placa.

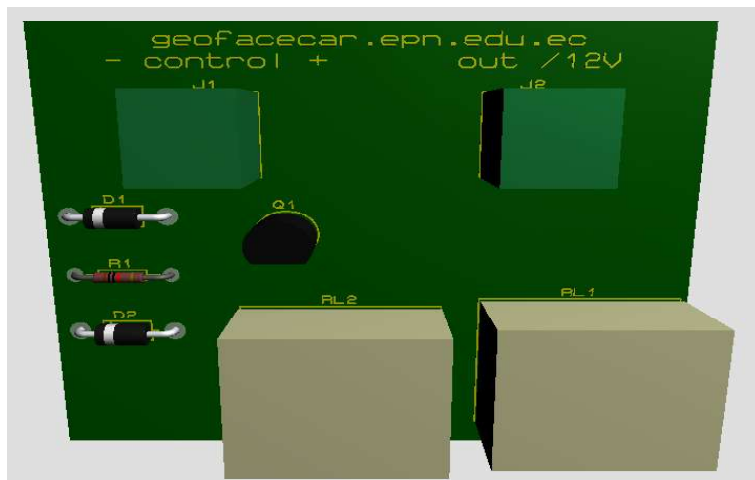
- **Ruteo del Circuito**

Para el ruteo del circuito electrónico completo, se utilizó el programa de diseño ARES, el mismo que permite ubicar los elementos, distancias entre ellos y las dimensiones de la placa. Este circuito impreso se lo puede observar en la Figura 3.9.



*Figura 3.9 Ruteo de la placa.*

Para tener un mejor enfoque del circuito electrónico, y una perspectiva global de la tarjeta, el software Proteus 8 permite visualizar en 3 dimensiones. De esta manera, se puede tener una visión realista del acabado final de la placa y de todos los elementos que conformarán la misma, como se muestra en la Figura 3.10.



*Figura 3.10 Visualización de la placa en 3D.*

- **Diagrama final del sistema para el encendido y rastreo automático vehicular**

En la Figura 3.11 se muestra el diagrama electrónico del sistema de encendido y rastreo automático vehicular, con un diseño cómodo y flexible para el usuario.

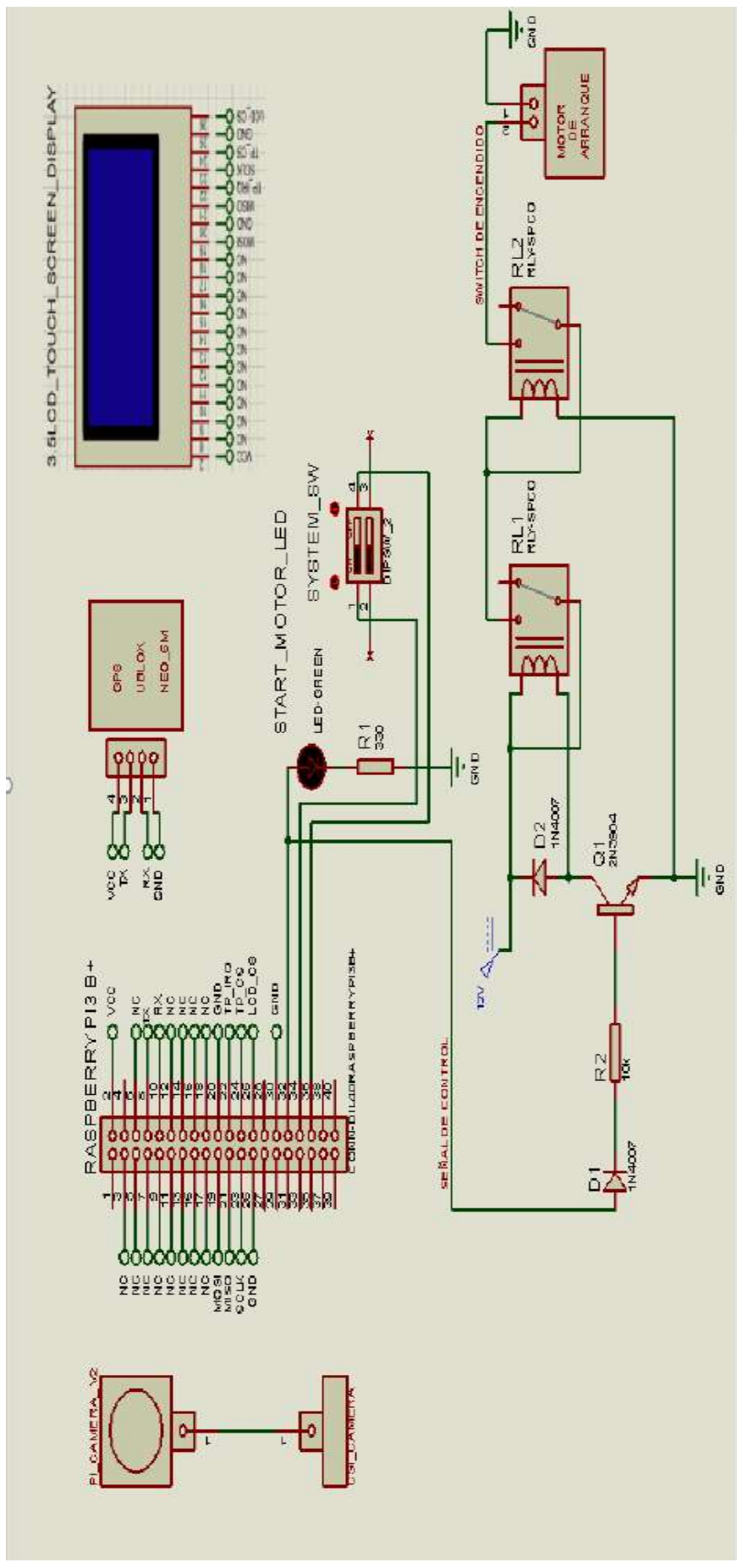


Figura 3.11 Diagrama completo de conexiones.



## **SOFTWARE**

- **Captura y detección de rostro frontal con *OpenCV***

En el proceso de detección y captura frontal del rostro, inicialmente se obtienen fotografías del usuario a ser detectado. Durante este periodo de almacenamiento del rostro en el sistema, se debe mantener una expresión facial normal. La distancia recomendada entre la persona y la cámara del módulo es de 0.3 m a 0.6 m, procurando un correcto enfoque con la cámara PiNoir.

Una vez realizadas las capturas del rostro (Figura 3.12), y posteriormente inicializado el clasificador, se pueden detectar los rasgos del rostro; para ello, se empleó la función `detectMultiScale` de OpenCV. Se ha usado la siguiente sintaxis:

```
faces = haar_cascade.detectMultiScale(mini)
```



*Figura 3.12 Fotografía frontal del rostro.*

Para realizar la detección de rostro frontal en un marco, se ha utilizado el clasificador Haar en cascada de OpenCV `haarcascade_frontalface_alt.xml`, el cual es un archivo XML, que permite detectar caras frontales completas; este archivo contiene datos de OpenCV. Se ha usado la siguiente sintaxis:

```
fn_haar = 'haarcascade_frontalface_alt.xml'
```

Posteriormente, se realiza la edición a escala de grises y corte a 112 x 92 píxeles de las fotografías. Estas imágenes tendrán la extensión .PNG, esto permite trabajar con menos información y detectar más detalles.

Como se puede observar en la Figura 3.13, la fotografía del rostro frontal se encuentra aplicada el filtro de escala de grises y corte. Se ha usado la siguiente sintaxis:

```
Gray = cv2.cvtColor (im, cv2.COLOR_BGR2GRAY)  
  
(im_width, im_height) = (112, 92)
```



*Figura 3.13 Fotografía en escala de grises y corte.*

Posteriormente, se crea una lista de imágenes y nombres correspondientes, las mismas, que se guardan en las carpetas att\_faces, orl\_faces y Daniel. Estas carpetas servirán posteriormente, para la verificación de identidad del usuario en la base de datos, para ello se hace uso de la librería face\_recognition, presente en OpenCV. Para esto se ha usado la siguiente sintaxis:

```
fn_dir = 'att_faces/orl_faces'  
  
fn_name = "Daniel"
```

A continuación, se devuelve una lista de rectángulos, en los que detecta un posible rostro, en una imagen capturada mediante video; seguido de esto, se crea un vector, en el que se determinan los parámetros necesarios para el almacenamiento en la correspondiente carpeta del rostro detectado.

Se devuelve 4 valores: la ubicación X, Y del rectángulo, y la anchura y altura del rectángulo W, H. Se ha usado la siguiente sintaxis:

```
(x, y, w, h) = [v * size for v in face_i]
```

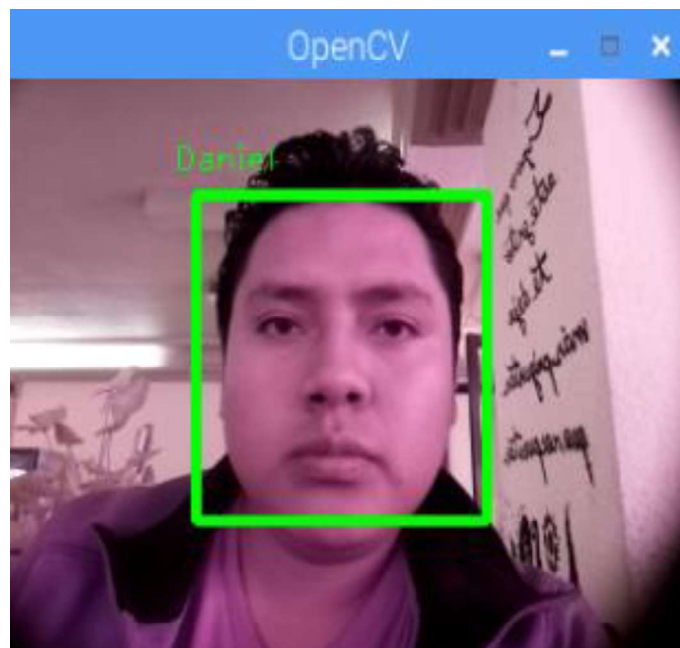
```
face = gray[y:y + h, x:x + w]
```

```
face_resize = cv2.resize(face, (im_width, im_height))
```

Finalmente, se toma estos valores para dibujar un rectángulo, usando la función `rectangle`; para lo cual, se ha usado la siguiente sintaxis:

```
cv2.rectangle(im, (x, y), (x + w, y + h), (0, 255, 0), 3)
```

El resultado del entrenamiento, determina un rectángulo que enmarca el rostro encontrado y detectado como Daniel, esto se puede observar en la Figura 3.14.



*Figura 3.14 Detección del rostro.*

La detección de rostros tiene como objetivo, encontrar los rostros en una imagen y extraerlos para usarlos con el algoritmo de reconocimiento facial, este proceso se puede observar en la Figura 3.15.

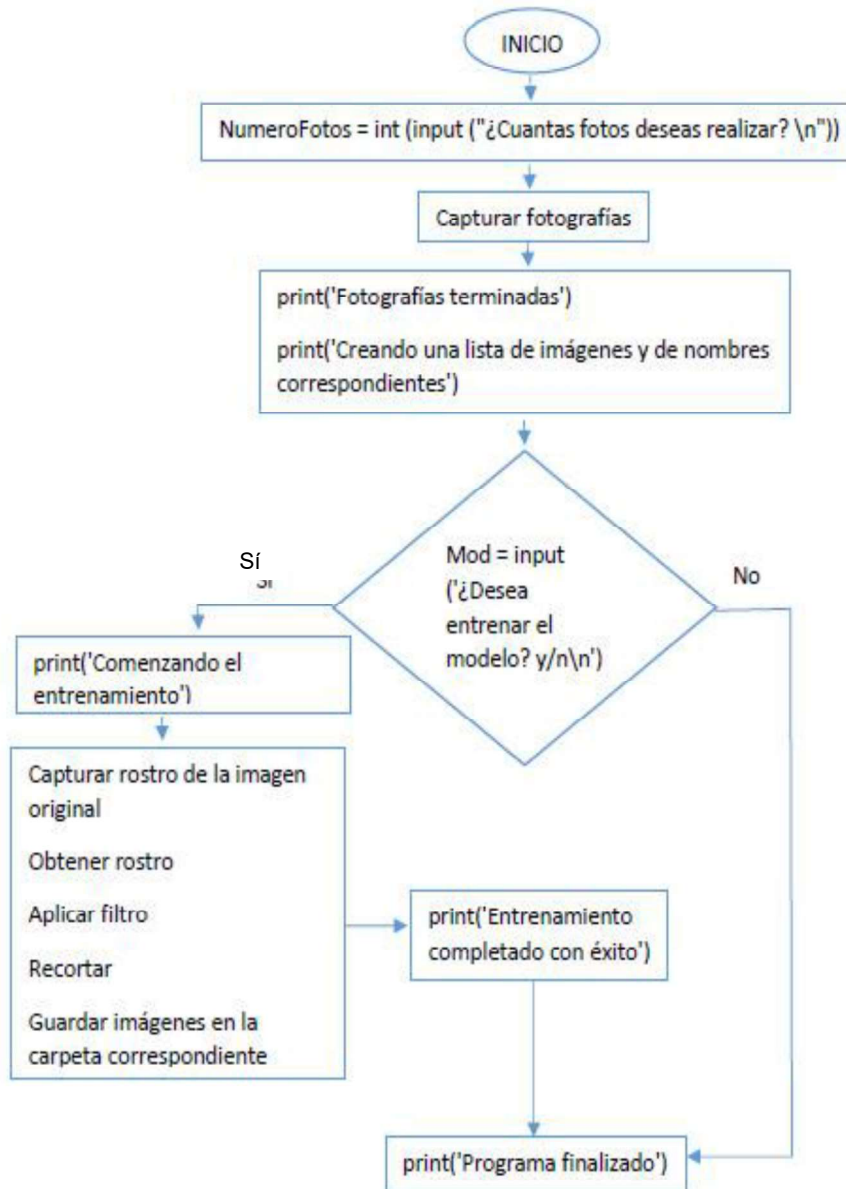


Figura 3.15 Diagrama de flujo de la captura y detección frontal del rostro.

- **Reconocimiento facial**

El Switch conectado al puerto GPIO 16 (pin 36) de la Raspberry Pi, da inicio al reconocimiento facial. La cámara se encenderá con una resolución de 76800 pixeles, y la pantalla conectada a la Raspberry mostrará el inicio del bucle de 0

a 15; este contador se volverá a reiniciar hasta que se haya reconocido al usuario. Desde la salida HDMI se podrá visualizar al usuario en tiempo real. El video en que se procede a capturar las imágenes, se verificará con la base de datos previamente almacenada.

Para esto se realizó un script, el cual permite cargar la imagen a la base datos, que resulta de la etapa de captura y detección. Para proceder a compararla con la persona que esté en ese momento frente a la cámara encendida, se ha usado la siguiente sintaxis:

```
daniel_image = face_recognition.  
load_image_file("/home/pi/Desktop/ej3/rostros/Daniel/25.png")  
daniel_face_encoding =  
face_recognition.face_encodings(daniel_image)[0]
```

En esta etapa, también se usó un temporizador, el reconocimiento se realizará cada segundo; esto quiere decir que, cada vez que transcurra 1 segundo, y haya un rostro válido, este rostro será procesado y comparado con la base de datos. En el caso que se detecte un rostro por 15 veces, de las cuales, si el reconocimiento del conductor ha sido correcto por 10 veces consecutivas, el LED se encenderá por 1 segundo, indicando, que el reconocimiento facial ha sido exitoso. En la pantalla se visualizará el mensaje "Bienvenido Daniel" y el motor del vehículo se encenderá. Para esto se ha usado la siguiente sintaxis:

```
if name=="Daniel":  
    face_rec_counter+=1  
    print ("face_rec_counter: "+str(face_rec_counter))  
    if face_rec_counter>=security:  
        print("SE DETECTO A Daniel ENCENDER AUTO")  
        face_rec_counter=0  
        camera.annotate_text = "Bienvenido Daniel"  
        GPIO.output(32, True) ## el led se enciende  
        time.sleep(2)  
        GPIO.output(32, False) ## el led se apaga  
        enviar_correos()  
  
camera.stop_preview()
```

De lo contrario, si el rostro no es identificado mínimo 10 veces consecutivas, dicho contador se reiniciará automáticamente, en un ciclo cerrado; hasta que,

haciendo uso del *Switch* se detenga el sistema de reconocimiento facial. Se ha usado la siguiente sintaxis:

```
else:  
face_rec_counter=0  
if tryes_counter> security+5:  
tryes_counter=0  
face_rec_counter=0
```

Esta técnica de verificación o autenticación de una imagen facial, básicamente compara la imagen facial de entrada con la imagen facial grabada en la base de datos. Es básicamente una comparación de 1x1. En la figura 3.16, se describe su funcionamiento.

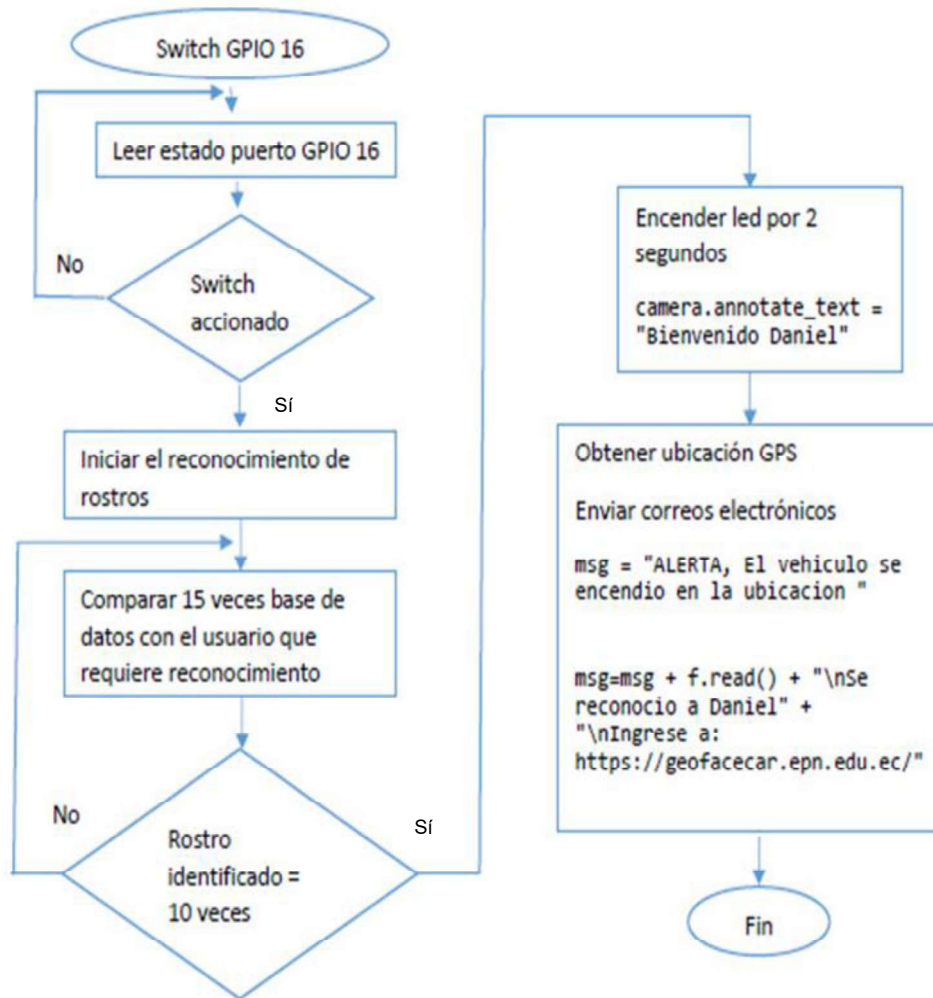


Figura 3.16 Diagrama de flujo de reconocimiento facial.

- **Librería face\_recognition**

Se utilizó la librería face\_recognition, para el análisis de las características faciales del usuario extraídas de la captura de una imagen, para compararlas con la base de datos. Con esta herramienta es posible realizar la identificación facial en tiempo real.

La librería face\_recognition escrita en Python, donde se usa framework dlib, se define como software multiplataforma de propósito general para el aprendizaje automático de máquinas. Este diseño está fuertemente influenciado por componente de software independiente. Es un código abierto lanzado bajo una licencia de software Boost.

Por ser un software multiplataforma, lo puede ejecutar desde cualquier sistema operativo. La librería face\_recognition ofrece una fiabilidad del 99,38%, según los test de LFW (*Labeled Faces in the Wild*). [40]

Esta librería puede realizar las siguientes acciones:

- Encontrar caras en una imagen dada.
- Encontrar y manipular rasgos faciales en una imagen.
- Identificar caras en imágenes.
- Reconocimiento fácil en tiempo real.

Se puede encontrar el código fuente completo de la librería face\_recognition alojado en los repositorios de GitHub.

Para realizar el reconocimiento del rostro, esta librería posee varios tipos de clasificadores Haar en cascada de OpenCV, específicamente para cada rasgo del rostro, como se observa en la tabla 1.

Tabla 3.1 Clasificadores en cascada de OpenCV. [41]

<b>TIPO DE CLASIFICADOR EN CASCADA</b>	<b>NOMBRE ARCHIVO XML</b>
Detector de ojos (derecho izquierdo)	haarcascade_lefteye_2splits.xml
Detector de boca	haarcascade_mcs_mouth.xml
Detector de nariz	haarcascade_mcs_nose.xml

Para la instalación de la biblioteca `face_recognition` se usó la siguiente configuración en la terminal:

```
sudo pip3 install face_recognition
```

Una vez instalada esta útil librería de Python, se realiza el script que permite el reconocimiento facial del usuario.

- **Ubicación GPS**

La ubicación GPS se dividió en dos etapas. La primera consiste en enviar una notificación de alerta por correo electrónico al usuario, indicando que el auto ha sido encendido; además, se adjunta las coordenadas actuales del auto. La segunda etapa es la del rastreo continuo del auto por medio de la página web, mediante el URL <https://geofacecar.epn.edu.ec/>. En la página web se visualizará la ubicación del auto a través de la API Google Maps.

- **Notificaciones de alertas**

Las alertas notificarán la ubicación actual del auto, señalando que el mismo ha sido encendido. Estas alertas se remitirán mediante Gmail al correo electrónico del usuario.

La librería `smtplib` viene integrada en Python, esta permite que las alertas se envíen por correo electrónico. Se creó el script que realiza las funciones de envío de texto desde la Raspberry Pi a cuentas de correo electrónico; para esto, se debe crear una cuenta de correo en Gmail y después realizar un login al correo creado. La sintaxis para estas funciones son las siguientes:

- *Correo electrónico remitente*

```
# Datos
username = 'reconocimiento.facial.tesis@gmail.com'
password = 'esfot2018'
```

- *Correos electrónicos destinatarios*



```
Toaddrs = ['danielperez@live.com', 'luiss_car@hotmail.com']  
msg=""
```

Para enviar la ubicación actual del auto en el mensaje de correo electrónico, se necesita añadir el script `actualización_gps_v1.py`, este se actualizará constantemente cada 6 segundos, con los datos de longitud y latitud. Estos datos se guardan en un archivo de texto, nombrado `posición.txt`, para posteriormente ser enviados mediante las alertas. La sintaxis para esta función es la siguiente:

```
def actualizacion_gps():  
    while True:  
        time.sleep(10)  
        os.system("sudo python  
/home/pi/Desktop/ej3/actualizacion_gps_v1.py")  
  
hilo_gps=Thread(target = actualizacion_gps, args = ())  
hilo_gps.start()
```

Por último, se necesita la conexión al servidor SMTP (*Simple Mail Transfer Protocol*), por el puerto 587, y se envía el correo con los datos actuales de longitud y latitud del auto. Adicionalmente, se adjunta la dirección de la página web <https://geofacecar.epn.edu.ec/>; ingresando a ella, se puede visualizar la ubicación actual del auto en la API de Google Maps. La sintaxis para esta función es la siguiente:

```
# Enviando el correo  
msg = "ALERTA, El vehiculo se encendio en la ubicacion "  
f=open("/home/pi/Desktop/ej2/posicion.txt")  
msg=msg + f.read() + "\nSe reconocio a Daniel" + "\nIngrese a:  
https://geofacecar.epn.edu.ec/"  
#print msg  
server = smtplib.SMTP('smtp.gmail.com:587')  
server.starttls()  
server.login(username,password)  
server.sendmail(fromaddr, correo,msg)  
server.quit()  
f.close()
```

- **Switch de inicio y LED indicador**

Se cuenta con un Switch conectado a la entrada del puerto GPIO 16 (pin 36) de la Raspberry Pi, este inicialmente está configurado en pull down; es decir, establece un estado de 0 lógico en reposo y cuando es activado un estado 1 lógico, al puerto GPIO 16. Esto evita los falsos positivos que se producen por el ruido generado por los circuitos electrónicos. Adicional a esto, se estableció el puerto GPIO 12 (pin 32) como salida para el LED indicador. La sintaxis para esta función es la siguiente:

```
GPIO.setmode(GPIO.BOARD)
GPIO.setup(32, GPIO.OUT) ## GPIO 32 como salida para el led
GPIO.setup(36, GPIO.IN, pull_up_down=GPIO.PUD_UP) ## GPIO 36
como entrada para el pulsador
```

Para dar inicio al sistema de reconocimiento facial, es necesario activar el Switch de inicio; si este no está activado, la cámara y pantalla se mantendrán apagadas. La sintaxis para esta función es la siguiente:

```
while GPIO.input(36):
time.sleep(0.5)
camera.stop_preview()
face_rec_counter=0
tryes_counter=0
bd_face_rec_in=True
```

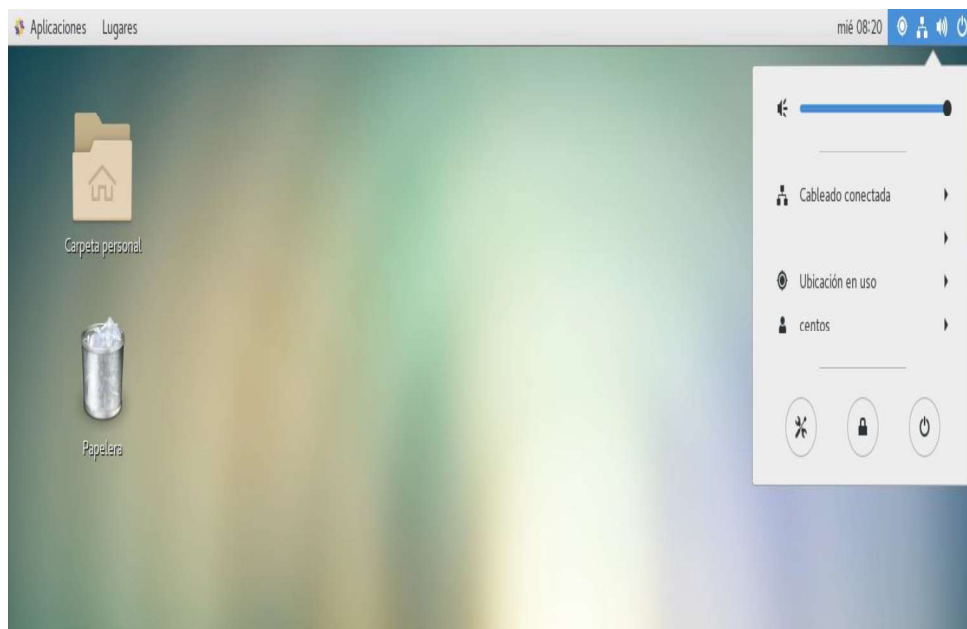
Si el Switch es activado, se encenderá la cámara y pantalla, dando inicio al sistema de reconocimiento facial. La sintaxis para esta función es la siguiente:

```
if bd_face_rec_in:
camera.start_preview()
bd_face_rec_in=False
```

- **SERVIDOR CENTRALIZADO**

Es importante recalcar que gracias a que el sistema operativo Linux es de libre acceso, no es necesario pagar por una licencia. Esta propiedad que ofrece el sistema operativo Linux, permite tener un control absoluto, de la administración y operatividad del aplicativo web.

Se han considerado dos parámetros importantes como son la latitud y la longitud; estos datos de la ubicación del vehículo, se almacenan en sus campos respectivos de la tabla en MariaDB, proporcionados por el receptor GPS. Para lo cual, se ha procedido a instalar un servidor web ubicado en el laboratorio de la EPN; este servidor es el encargado de recibir, almacenar, actualizar y mostrar en tiempo real la ubicación del automóvil. Para ello, se ha instalado la imagen de Centos 7 en un disco de 1 TB, como se puede apreciar en la Figura 3.17.



*Figura 3.17 Escritorio de Centos 7.*

- **Página web**

Para la implementación de la página web, se procedió a instalar un servidor web LAMP. Esta infraestructura de internet, hace posible que cualquier persona, que previamente obtenga las credenciales del login de la página web, pueda acceder a la página web de rastreo satelital, previa instalación de cada uno de los elementos, de los que se conforma el servidor LAMP.

Es indispensable obtener una dirección IP estática, esta dirección fue proporcionada por el departamento de tecnología de la DGIP (Dirección de Gestión de la Información y Procesos) de la EPN.

La configuración automática de las máquinas del laboratorio se encuentra en modo DHCP (*Dynamic Host Configuration Protocol*); por tanto, es indispensable utilizar una dirección IP estática, debido a eso se usó la dirección 172.31.XXX.XXX. proporcionada por la EPN.

Seguido de esto, se procede a instalar el servidor WEB. Figura 3.18.

**Dirección IP Estática**

```
centos@geofacecar:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[centos@geofacecar ~]$ ifconfig  
eno1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 172.31.XXX.XXX netmask 255.255.255.192 broadcast 172.31.116.127  
    inet6 2801:0:270:e6:8312:5b01:ca33:d9f6 prefixlen 64 scopeid 0x0<global>  
l>  
    inet6 fe80::ee0d:27b8:ce40:273e prefixlen 64 scopeid 0x20<link>  
    ether 2c:44:fd:1d:89:d0 txqueuelen 1000 (Ethernet)  
    RX packets 14393 bytes 20018209 (19.0 MiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 8297 bytes 956085 (933.6 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
    device interrupt 20 memory 0xf7c00000-f7c20000  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 64 bytes 5536 (5.4 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 64 bytes 5536 (5.4 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figura 3.18 Dirección IP Estática.

- **Conexión entre el módulo *Raspberry* y el servidor *Web***

Para que exista comunicación entre la Raspberry y la computadora del laboratorio, que en este caso, es la central de almacenamiento de información donde se encuentra albergada la aplicación web de rastreo satelital, se ha realizado una conexión inalámbrica, mediante internet entre el servidor y la Raspberry.

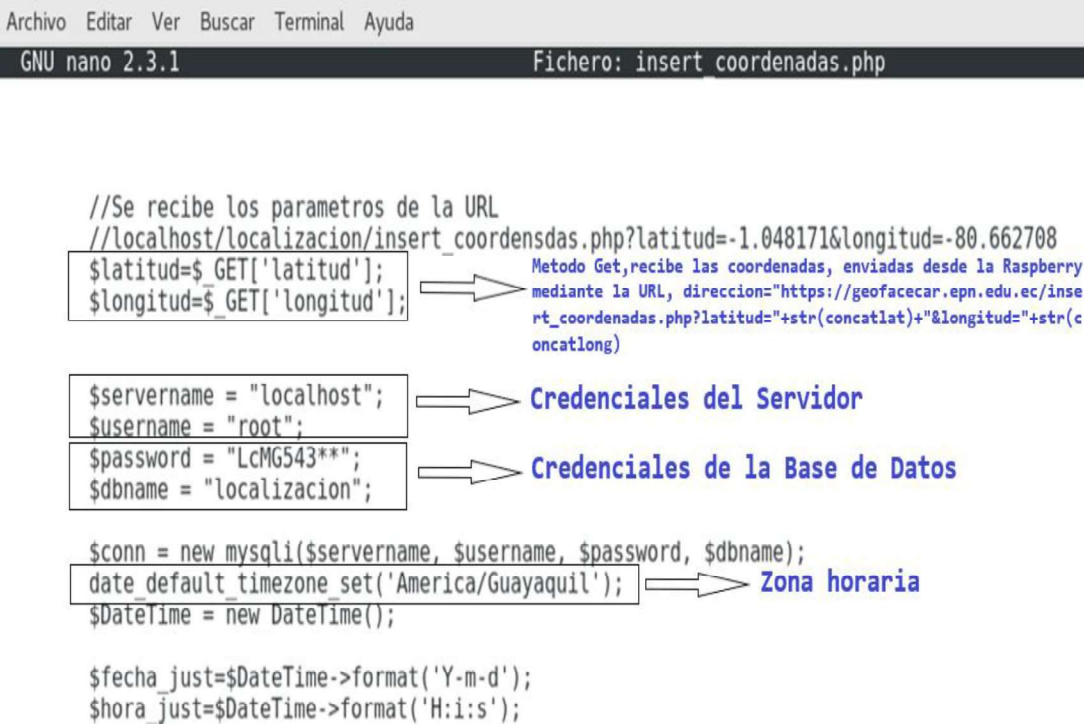
La conexión a internet de la Raspberry se realizó usando un Smart Phone, ya que inicialmente se consideró un Modem USB 3G, este dispositivo perdía

conexión cuando se alejaba de la radio base más cercana, dejando sin conexión a internet a la Raspberry.

La transferencia de datos entre la Raspberry y el servidor se realiza mediante el puerto 3306, este puerto está destinado para el manejo de base de datos en *MYSQL*.

Por medio de este puerto, se inyectan los datos al servidor, en tiempo real, los datos de latitud y longitud.

Se establece un canal de comunicación entre la Raspberry y el servidor Web, como se observa en la Figura 3.19.



```
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.3.1 Fichero: insert coordenadas.php

//Se recibe los parametros de la URL
//localhost/localizacion/insert_coordensdas.php?latitud=-1.048171&longitud=-80.662708
$latitud=$_GET['latitud'];
$longitud=$_GET['longitud'];
Metodo Get, recibe las coordenadas, enviadas desde la Raspberry
mediante la URL, direccion="https://geofacecar.epn.edu.ec/inse
rt_coordenadas.php?latitud="+str(concatlat)+"&longitud="+str(c
oncatlong)

$servername = "localhost";
$username = "root";
Credenciales del Servidor

$password = "LcMG543**";
$dbname = "localizacion";
Credenciales de la Base de Datos

$conn = new mysqli($servername, $username, $password, $dbname);
date_default_timezone_set('America/Guayaquil');
DateTime = new DateTime();
Zona horaria

$fecha_just=$DateTime->format('Y-m-d');
$hora_just=$DateTime->format('H:i:s');
```

Figura 3.19 Conexión de la Raspberry con el Servidor Web.

- **Base de Datos MariaDB**

Por protocolos de seguridad de red, dispuestos por la CSIRT (Centro de respuestas a Incidentes de Seguridad Informática) de la EPN, todo el sistema se maneja estrictamente por consola, quedando deshabilitado por completo el modo gráfico. Como se puede apreciar el ingreso a la base de datos mediante

consola, en la Figura 3.20, previo a la identificación del administrador de superusuario root, seguido de la identificación para el ingreso a la base de datos MariaDB.

```
centos@geofacecar:/home/centos
Archivo Editar Ver Buscar Terminal Ayuda
[centos@geofacecar ~]$ su
Contraseña:
[root@geofacecar centos]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 9
Server version: 5.5.60-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Figura 3.20 Identificación para el Ingreso a la Base de Datos.

A continuación, se ejecuta el comando show databases, donde se muestra el contenido general de la base de datos, como se puede observar en la Figura 3.21.

```
centos@geofacecar:/home/centos
Archivo Editar Ver Buscar Terminal Ayuda
Your MariaDB connection id is 9
Server version: 5.5.60-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| localizacion |
| mysql |
| performance_schema |
+-----+
4 rows in set (0.00 sec)

MariaDB [(none)]> use localizacion;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [localizacion]> show tables;
```

Figura 3.21 Base de Datos.

Una vez ingresado al archivo localización, se debe digitar el comando (show tables); este comando permitirá el acceso al contenido de la tabla localización.

Para tener acceso a la información de la tabla posicionamiento, es necesario ingresar el siguiente comando (DESCRIBE posicionamiento), Figura 3.22. De inmediato se desplegará una tabla con la latitud, longitud, fecha y hora; dichos campos actualizarán la ubicación del vehículo cada 6 segundos, con una precisión de 2.5 metros a la redonda.

```
MariaDB [(none)]> use localizacion;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [localizacion]> show tables;
+-----+
| Tables_in_localizacion |
+-----+
| posicionamiento        |
| usuarios               |
+-----+
2 rows in set (0.00 sec)

MariaDB [localizacion]> DESCRIBE posicionamiento;
+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+
| id         | int(11)   | NO   |     | NULL    |      |
| latitud    | char(70)  | NO   |     | NULL    |      |
| longitud   | char(70)  | NO   |     | NULL    |      |
| fechahora  | datetime | NO   |     | NULL    |      |
+-----+
4 rows in set (0.02 sec)

MariaDB [localizacion]> SELECT * FROM posicionamiento;
+-----+
| id | latitud | longitud | fechahora |
+-----+
| 1  | -0.10361 | -78.4901283333 | 2018-12-17 14:27:09 |
+-----+
1 row in set (0.00 sec)

MariaDB [localizacion]> 
```

Describe los datos contenidos en la tabla

Muestra la última ubicación recibida del GPS

Figura 3.22 Última ubicación registrada en la DB.

Cabe aclarar que, la precisión de la ubicación varía, dependiendo de la línea de vista que se tenga en ese momento con los satélites. Por tal razón, el GPS fue ubicado en la parte superior del parabrisas del vehículo.

- **API de Google Maps**

Para visualizar la ubicación del vehículo en la *API de Google Maps*, se procede a crear una cuenta de *Gmail*, misma que permite obtener una clave. Esta clave se genera gratuitamente, siempre y cuando, se haya especificado el propósito y alcance del aplicativo. En este caso en particular, se determinó el uso exclusivo en modo de desarrollo; con ello, se limita el número de usuarios que pueden observar el aplicativo a 32 usuarios. Para que el mapa se pueda observar en la web, la clave generada debe insertarse en el script de mapa ubicado en el archivo localización. Para ello, se procede como se presenta en la Figura 3.23, utilizando el comando (sudo nano mapa.html).

```
centos@geofacecar:/var/www/html/localizacion/pages
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.3.1 Fichero: mapa.html

</div>

<!-- #wrapper -->

<!-- jQuery -->
<script src="../../vendor/jquery/jquery.min.js"></script>


<!-- Bootstrap Core JavaScript -->
<script src="../../vendor/bootstrap/js/bootstrap.min.js"></script>

<!-- Metis Menu Plugin JavaScript -->
<script src="../../vendor/metisMenu/metisMenu.min.js"></script>

<!-- Custom Theme JavaScript -->
<script src="../../dist/js/sb-admin-2.js"></script>

<script src="https://maps.googleapis.com/maps/api/js?key=AIZaSyDKlvQxEou_zZT4APH1QsfmLoNQ3tQa8BA&callback=initMap"
async defer"></script>

</body>
</html>
```



KEY de la API de Google Maps, permite la visualización del mapa en la pagina Web

Figura 3.23 Key de la API de Google Maps.

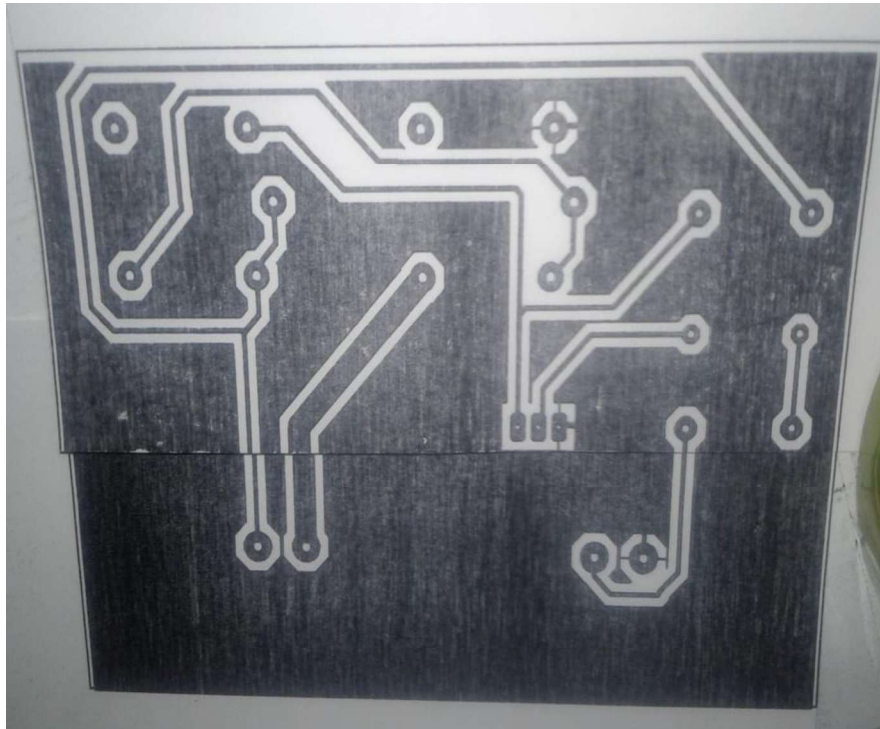
## 3.2 CONSTRUCCIÓN DEL SISTEMA

- **Ruteo del Circuito**

Para la realización de esta PCB, se usó el programa de diseño electrónico Proteus versión 8. Se tomaron en cuenta algunos aspectos importantes como:



tamaño de la placa, ubicación de los elementos y la distancia entre las fuentes de alimentación. En la Figura 3.24, se puede observar el ruteo de la placa de control.



*Figura 3.24 Diseño y ruteo de la placa de control.*

- **Medidas de la placa**

La placa mide 70 mm x 80 mm, tamaño elegido en función de las dimensiones de los relés empleados para este propósito. El material de la base de la placa está constituido por fibra de vidrio; por tanto, aporta consistencia y seguridad a la placa al momento de insertar los diferentes elementos en las pistas.

- **Proceso de construcción de la placa**

Para la transferencia y grabado de las pistas en la fibra de vidrio, primero se procede a imprimir a láser en alta calidad, usando papel fotográfico formato A4 de 0.5 mm de espesor. Seguidamente, se debe planchar el papel fotográfico en la fibra de vidrio, teniendo en cuenta una temperatura adecuada, para que la transferencia de las pistas sea uniforme. A continuación, se utiliza una mezcla de

agua caliente y ácido férrico en polvo. El tiempo de espera para despejar las pistas del lienzo de cobre, depende de la cantidad de ácido férrico que se emplee.

Además, se debe, tener precaución al momento de manipular este ácido, ya que puede ser perjudicial para la salud; por tanto, es indispensable usar guantes de látex. Para la remoción del ácido férrico de las pistas, se usa agua tibia. Finalmente, se perfora con mucho cuidado los orificios donde se ensamblarán los elementos de la placa de control. Como se puede apreciar en la Figura 3.25.



Figura 3.25 Proceso de construcción de la placa.

- **Acople del sistema de reconocimiento facial con el sistema de encendido del auto**

Para realizar la conexión y desconexión del arranque, se procede a desmontar la tapicería que cubre el *Switch* de encendido y se interrumpirá el cable que

conecta a dicho *Switch* con el motor de arranque. Esto se observa en la Figura 3.26.

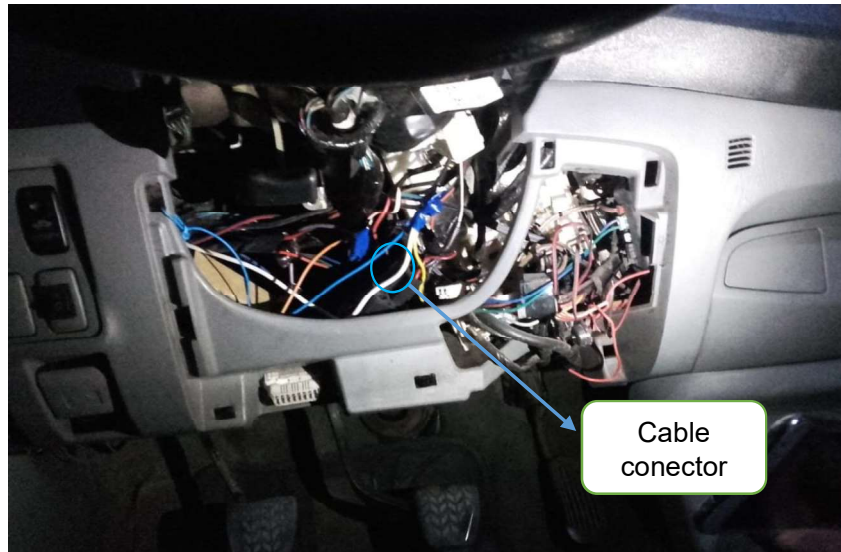


Figura 3.26 Cable conector entre *Switch* de encendido y motor de arranque.

Finalmente, el acople del sistema de reconocimiento facial con el sistema de encendido del auto se lleva a cabo mediante una conexión del puerto GPIO 12 (pin 32) de la *Raspberry* a la placa de control, de igual manera el cable conector del motor de arranque a dicha placa. Esto se puede observar en la Figura 3.27.

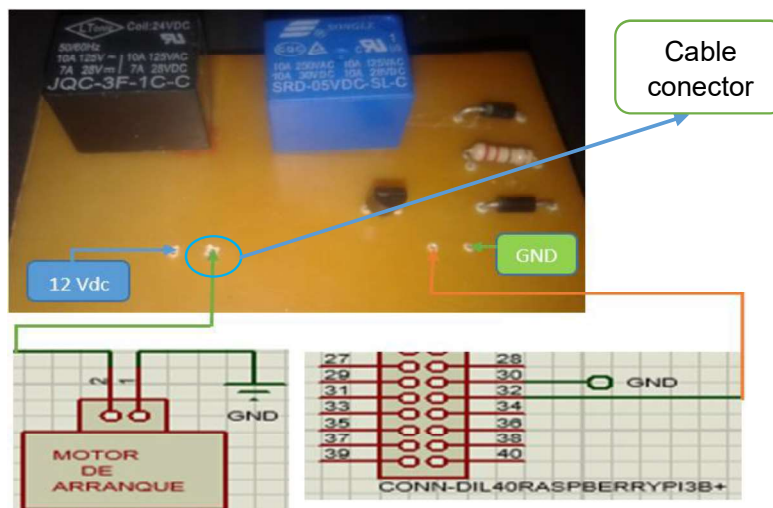


Figura 3.27 Acoplamiento con el sistema de encendido.

### 3.3 CONFIGURACIÓN PREVIA

- **Configuración previa en la *Raspberry***

La configuración previa se ejecuta una sola vez; en caso de que el sistema requiera de nuevas actualizaciones, tan solo se deben ejecutar simples comandos de actualización en el terminal de la Raspberry. Esto permite que no se configure el sistema cada vez que inicia; sino más bien, es recomendable hacer actualizaciones trimestralmente.

- **Cargar el sistema operativo *Raspbian Stretch* en la SD**

Para iniciar la instalación del sistema operativo, primero se debe formatear la SD de 32 GB, utilizando el software *SD Formatter*, como de observa en la Figura 3.28.

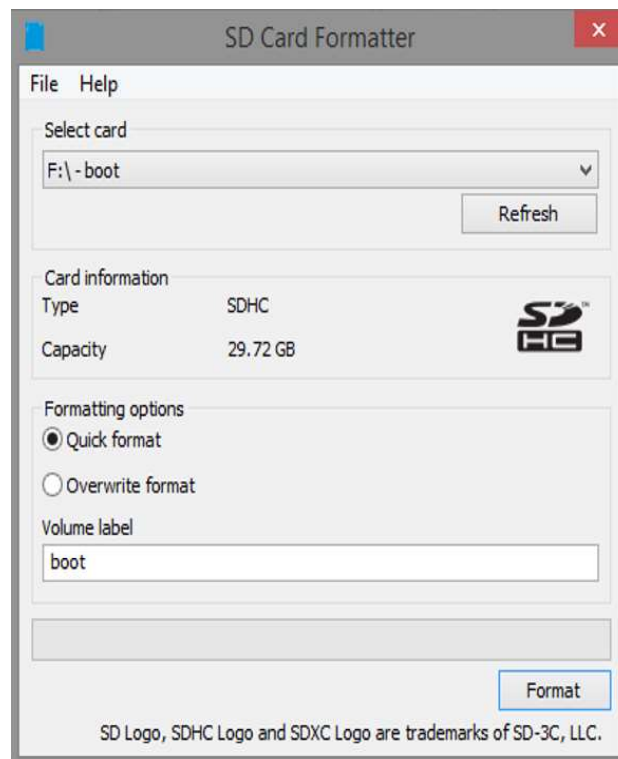


Figura 3.28 Software SD Card Formatter.

Una vez finalizado el proceso de formatear la SD, aparecerá una ventana emergente de confirmación, que la SD ha sido formateada con éxito, como se puede apreciar en la Figura 3.29.

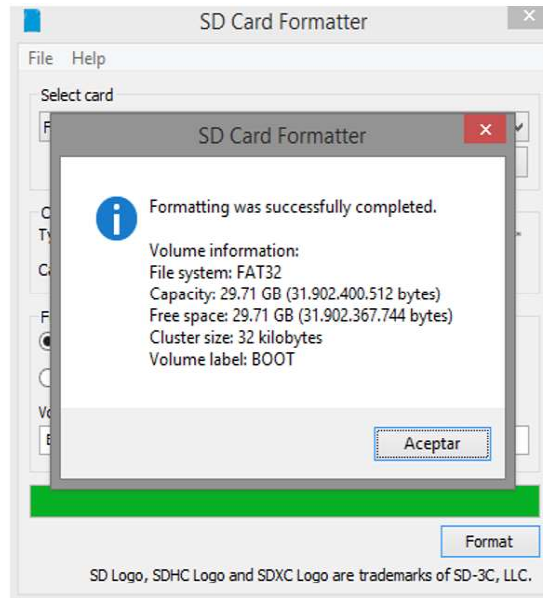


Figura 3.29 Software SD Card Formatter finalizado.

A continuación, se graba el sistema operativo *Raspbian Stretch*, en la SD Card, utilizando el aplicativo *Win32 Disk Imager*. Figura 3.30. Para ello, se usó una SD de 32 GB *Sandisk* clase 10.

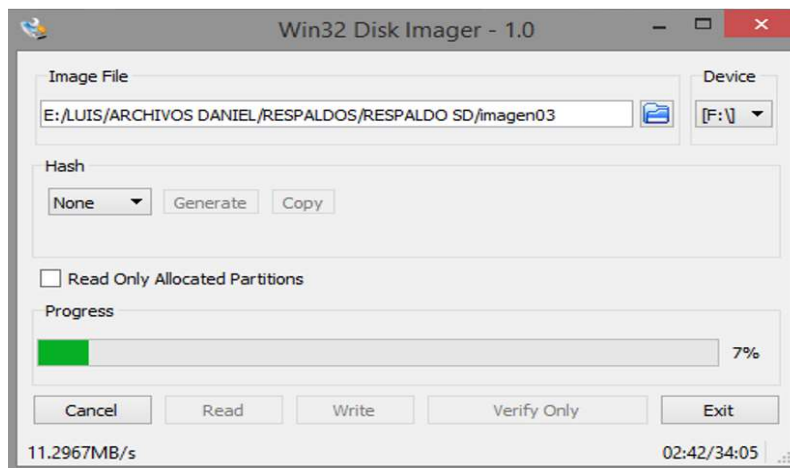


Figura 3.30 Software Win32 Disk Imager.

Una vez terminada la escritura de la imagen, se procede a colocar la micro en el puerto SD de la *Raspberry*.

Para la visualización del escritorio de sistema operativo *Raspbian Stretch*, se usa SSH, mediante el puerto 22, para hacer una conexión remota a una computadora, como se observa en la Figura 3.31.

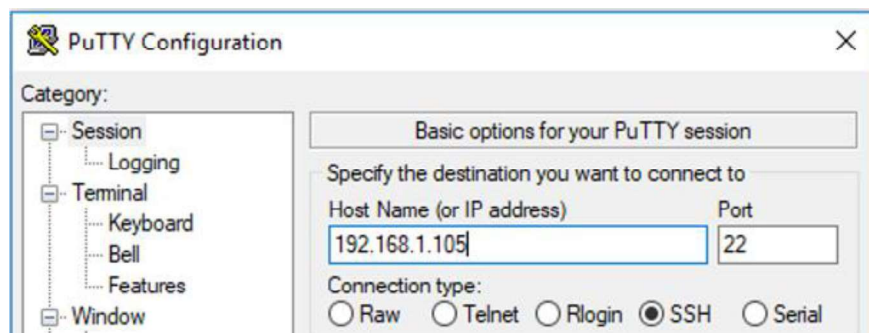


Figura 3.31 Conexión remota mediante PuTTY.

Una vez establecida la conexión remota, después de esperar algunos segundos, se podrá visualizar el escritorio del sistema operativo Raspbian Stretch, como se puede observar en la Figura 3.32.

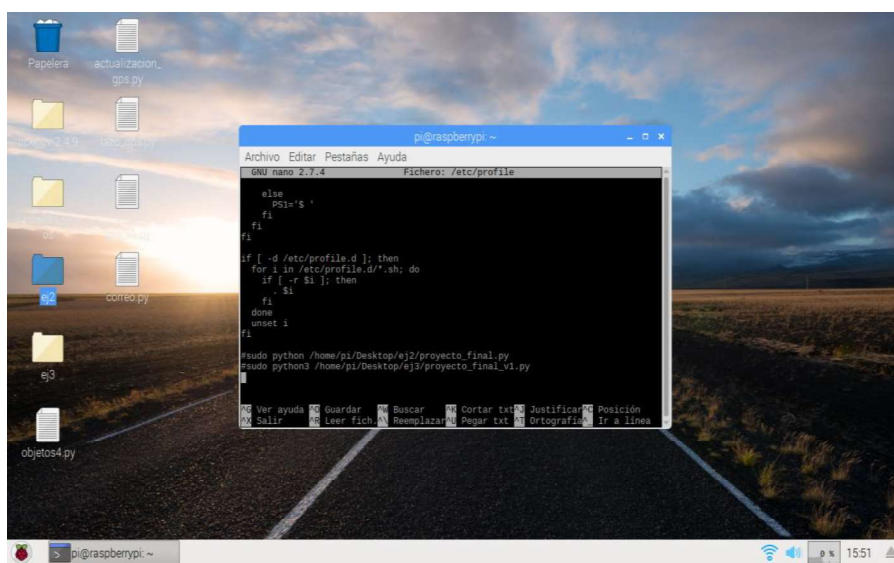


Figura 3.32 Interfaz del SO Raspbian Stretch.

### 3.4 PRUEBAS DE FUNCIONAMIENTO

A continuación, se detalla las diferentes pruebas efectuadas en el sistema para comprobar su correcto funcionamiento.

- **Prueba de Cámaras**

Se utilizaron tres tipos de cámaras. Para definir la más adecuada, en primera instancia, se usó una webcam de 3.85 mm, cuya resolución es de mala calidad. Adicionalmente, al conectarse por medio del puerto USB, genera un retardo de aproximadamente 1.5 segundos al grabar en video; lo que genera que cualquier persona con características similares al individuo registrado en la base de datos, sea reconocido como propietario del auto; sin ni siquiera existir en dicha base. Por tanto, esta opción de cámara fue definitivamente descartada del proyecto.

En segunda instancia, se probó una cámara nativa para *Raspberry RaspPiCam* de 8 mega pixeles, cuyos resultados de calidad de imagen son excelentes, con la cual se puede tener una buena recepción de colores en el día. Pero, el inconveniente con este tipo de cámara, es que no sirve para la noche o en ambientes con poca luz.

Finalmente, se probó con una cámara nativa de Raspberry PiNoir, como se observa en la Figura 3.33, con un resultado de imagen y resolución nítida. Para eliminar el inconveniente de la noche, se implementó una lámpara de LEDs infrarrojos. El retraso del video es imperceptible al ojo humano.



Figura 3.33 Cámara PiNoir (a) iluminación alta, (b) iluminación mínima.

Se puede notar que la cámara PiNoir en el día no tiene una buena percepción de colores, dado que su estructura interna tiene eliminada la parte infrarroja. Sin embargo, al momento del procesamiento esto no repercute ni afecta en lo absoluto, dado que el algoritmo LBP trabaja en formato de grises, permitiendo conseguir menos carga computacional.

Como se puede apreciar en la Figura 3.34, la misma fotografía con o sin iluminación se transforma en un formato a escala de grises, lo que permite trabajar con números enteros, ya que la fotografía es dividida en tonos blancos y negros.



*Figura 3.34 Escala de Grises.*

- **Reconocimiento de Usuario**

La librería `face_recognition` es una colaboración de múltiples investigadores, la misma que es de libre acceso al 100% del código fuente, en la que según miles de pruebas realizadas por sus creadores, presenta una fiabilidad de 99.38 % [40]. El resultado de confiabilidad utilizado por el algoritmo LPB, se enfoca en captar ciertos detalles específicos del rostro, como son el de la boca, ojos, nariz, barbilla etc.

Esta discriminación de zonas específicas del rostro, determinan que el proceso de identificación de una persona u objeto sea muy rápido, comparado con otros métodos y algoritmos de reconocimiento facial.

Para garantizar la fiabilidad del reconocimiento, en la librería `face_recognition` se creó un bucle, en el que se realiza un barrido de imagen a través de video que



va de 0 a 15. Adicionalmente, se delimitó un rango de 0 a 10 dentro de dicho bucle. De esta manera, si el usuario es reconocido durante 10 veces consecutivas, el vehículo se encenderá; caso contrario el bucle se reiniciará indefinidamente, hasta que sea desactivado mediante el Switch de control.

En la Figura 3.35, se puede apreciar el bucle de conteo ascendente. En la parte izquierda se enumeran las veces reconocidas acertadas del usuario, y en la parte derecha se mostrará el número de capturas de imágenes, independientemente corresponden o no al usuario previamente grabado en la base de datos.



*Figura 3.35 Bucle de reconocimiento facial.*

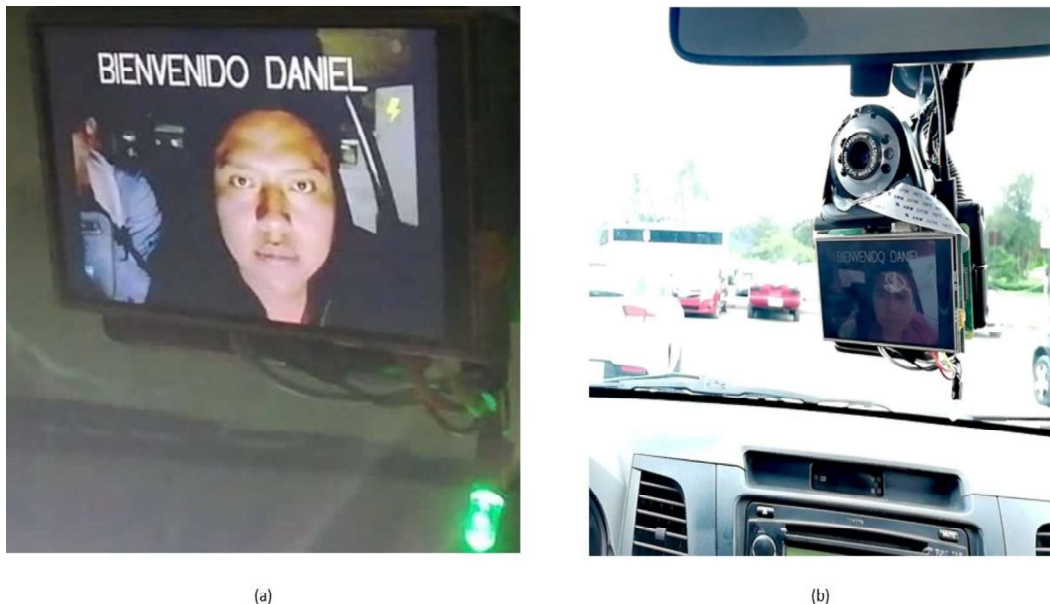
Al someter el sistema de reconocimiento facial a pruebas de identificación con otras personas ajenas a la base de datos, se obtuvieron resultados totalmente satisfactorios. Los mismos se pueden apreciar en la Figura 3.36. Nótese que el sistema no reconoce en todos los casos. El sistema fue probado con personas

que presentan rasgos similares al del usuario, quien es el único almacenado en la base de datos.



*Figura 3.36 Usuarios (a), (b), (c) son desconocidos para el sistema.*

El usuario registrado en la base de datos, es el único detectado y reconocido por el sistema; en la Figura 3.37 se observa que únicamente con el reconocimiento facial del usuario registrado, se permite el encendido del vehículo, una vez culminado el reconocimiento de 10 veces consecutivas. De esta manera, se da paso a un mensaje de bienvenida; simultáneamente, se enciende un diodo LED de color verde.



*Figura 3.37 Reconocimiento exitoso (a) mínima iluminación, (b) alta iluminación.*

- **Notificaciones al correo electrónico**

Una vez reconocido el usuario, se procede a comprobar el correcto funcionamiento del envío de la latitud y longitud, con una tolerancia aproximadamente de 2.5 metros a la redonda. Este error puede crecer, dependiendo de la línea de vista que en ese momento se tenga entre el módulo GPS y los satélites. El envío del correo electrónico se realiza, teniendo en cuenta que exista conexión a internet tanto en la Raspberry como en el servidor, mediante el puerto 587.

Para la comprobación de la notificación al correo electrónico de la ubicación e identificación del usuario, se han tomado dos partes; la primera es el correo remitente 'reconocimiento.facial.tesis@gmail.com', enviado desde la Raspberry. La segunda parte son los correos destinatarios, que para este caso son, 'danielperez@live.com', 'luiss\_car@hotmail.com', en donde se comprueba la correcta recepción de la notificación, como se puede observar en la Figura 3.38.

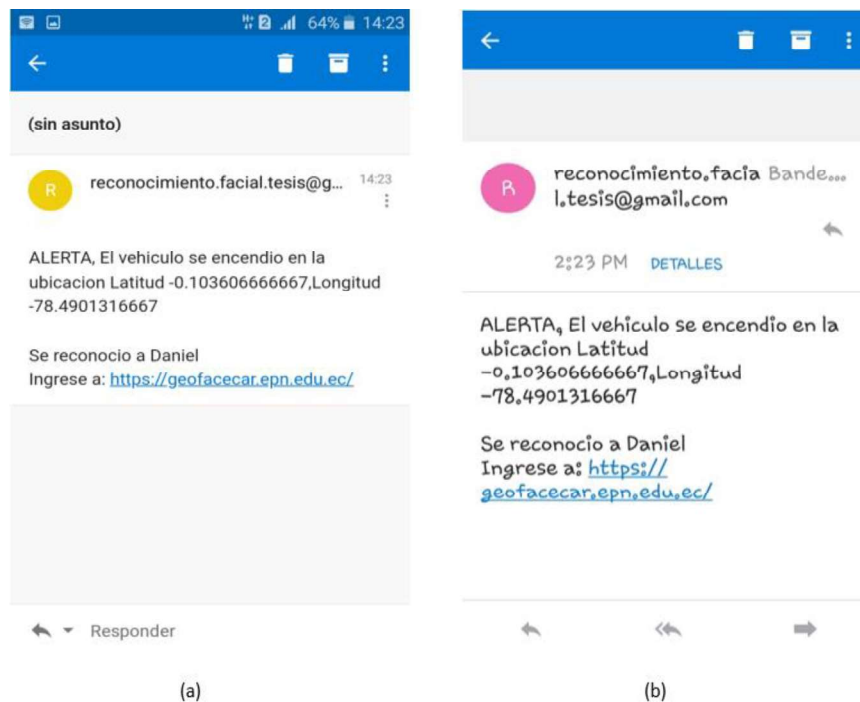


Figura 3.38 Recepción en los correos destinatarios.

Posteriormente, se puede acceder a la URL de <https://geofacecar.epn.edu.ec/> para realizar el rastreo satelital mediante la API de Google Maps. Previo un login,

donde usuario y contraseña es “admin” para ambos casos, como se observa en la Figura 3.39.

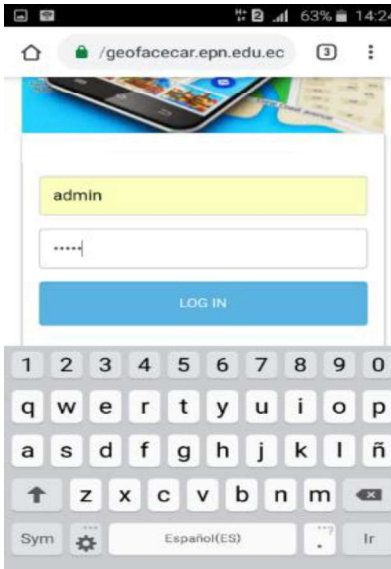


Figura 3.39 Ingreso a la aplicación.

Para la navegación en el mapa de *Google Maps*, se ha definido que cada 15 segundos se refresque la pantalla con la ubicación. Esto por motivo de las restricciones que presenta la *API* de *Google*, por tratarse netamente, como propósito de desarrollo. A continuación, se muestra en la Figura 3.40 la ubicación normal y con zoom.

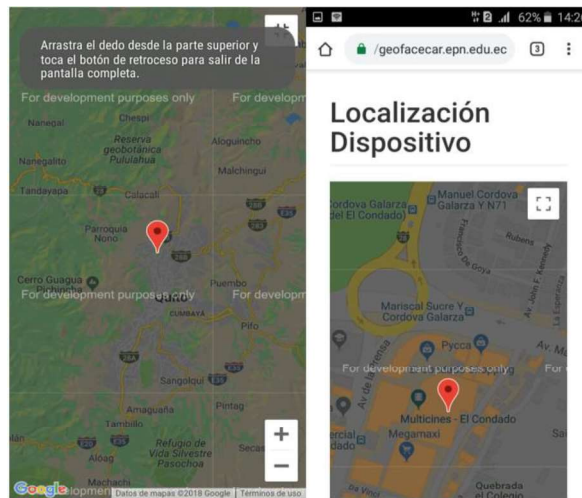


Figura 3.40 Ubicación (a) normal, (b) con zoom.

### 3.5 COSTOS DE IMPLEMENTACIÓN

Tabla 3.2 Costos del Proyecto.

DETALLE	PRECIO UNITARIO	CANTIDAD	TOTAL
Tarjeta Raspberry Pi 3 modelo B+	\$80,00	1	\$80,00
Power Bank de 5v (3,4A)	\$50,00	1	\$50,00
GPS Ublox NEO 6M	\$25,00	1	\$25,00
SD Card 32 GB	\$22,00	2	\$44,00
Adaptador HDMI a VGA	\$15,00	1	\$15,00
Cámara Pi Noir V2	\$65,00	1	\$65,00
LCD TOUCH SCREEN DISPLAY 3.5"	\$200,00	1	\$200,00
Switch	\$1,00	1	\$1,00
Led	\$0,25	1	\$0,25
Resistores 330 $\Omega$ , 10 k $\Omega$	\$0,15	2	\$0,30
Relé 12v	\$8,00	2	\$16,00
2N3904	\$0,20	1	\$0,20
Diodo 1N4007	\$0,10	2	\$0,20
Borneras	\$0,50	2	\$1,00
Par trenzado categoría 5E/100MHZ	\$0,80	1	\$0,80
Papel fotográfico	\$0,75	1	\$0,75
Baquelita PCB 10X15 cm	\$1,50	1	\$1,50
Soporte celulares auto	\$8,00	1	\$8,00
Caja de madera	\$3,00	1	\$3,00
	<b>COSTO TOTAL</b>		<b>\$512,00</b>

Nota: en el presente presupuesto referencial no se han considerado rubros del total de horas efectivas de trabajo o de desarrollo intelectual.

## 4. CONCLUSIONES Y RECOMENDACIONES

### 4.1 CONCLUSIONES

- El desarrollo del sistema de seguridad electrónico para el encendido vehicular, aparte de ser un proyecto innovador, incentiva a otros estudiantes y público en general a continuar investigando, mejorando e implementando nuevos métodos de seguridad que contribuyan a solucionar esta oleada de delincuencia radicada en la actualidad en nuestra sociedad.
- Implementar un sistema de reconocimiento facial para el encendido automático de un vehículo, ahora es posible gracias a la enorme cantidad de información existente en internet, para el manejo de la *Raspberry Pi 3 B+*, la información se encuentra en repositorios, donde comunidades virtuales de programación, permiten el libre acceso a los códigos fuente, que son editables y no tienen ningún costo.
- La tarjeta Raspberry Pi 3 B+, claramente ha demostrado ser un sistema embebido con potencialidades asombrosas, tales como ser denominada como un micro-ordenador. Posee múltiples características iguales a un ordenador convencional de escritorio. Por ejemplo, conexión de internet a través de ethernet, wifi y salidas de audio y video.
- Hace pocos años atrás, el aprendizaje automático de máquinas no era tan viable, considerando las limitaciones de procesamiento y de funcionamiento que presentaban los ordenadores convencionales. Pero con el avance acelerado de estos microprocesadores, como es el caso de la familia *Raspberry*, permite que cualquier persona pueda desarrollar sus propias aplicaciones y proyectos de investigación. Gracias a que la plataforma presenta repositorios de código abierto e interfaces interactivas, sin ninguna limitación, si al momento de programar y acoplar módulos periféricos se trata.
- La cámara PiNoir V2, sin duda alguna, es una de las herramientas más impresionantes, dado que este dispositivo permite capturar y grabar

imágenes en tiempo real, con gran precisión y una elevada calidad de imagen. Adicionalmente, la versatilidad que posee para trabajar en distintos ambientes, pudiendo entrenar imágenes con absoluta obscuridad hasta ambientes con una elevada intensidad de luz, únicamente se dispondrá de una lámpara, integrada de leds infrarrojos, para trabajar en todos los ambientes.

- Lo más sobresaliente de JavaScript es que no necesita de compilación, ya que se trata de un lenguaje interpretado; esto es realizado por los navegadores web, y funciona del lado del cliente, más no del lado del servidor. De esta manera, permite manejar grandes números de usuarios en línea, eliminando la posibilidad de colapsar la página web.
- Tanto los receptores GPS de mano, como los instalados en vehículos con antena exterior fija, necesitan abarcar el campo visual de los satélites. Generalmente, estos dispositivos no funcionan bajo techo ni debajo de las copas de los árboles; para que trabajen con precisión, hay que situarlos en el exterior, preferiblemente donde no existan obstáculos que impidan la visibilidad, y reduzcan su capacidad de captar las señales que los satélites envían a la Tierra.
- La modalidad de conexión (contacto hacia masa) del transistor 2N3904, tiene la ventaja de permitir el uso de relés con intensidades elevadas en sus salidas, lo que facilita el acoplamiento de dispositivos que usan una mínima cantidad de corriente, como miliamperios; pero, en el caso del motor de arranque de un automóvil, el rango de trabajo de corriente oscila entre los 30 a 60 amperios, dependiendo del modelo y la marca del mismo.
- Como la alimentación de control es totalmente independiente a la alimentación de los relés, se puede trabajar con tensiones separadas para la parte de control y para los relés. Esto es muy útil, porque da libertad de conectar distintos tipos de relé, sin necesidad de modificar la parte de control.

Además, se puede evitar interferencias generadas por la bobina del relé, sobre el circuito de control (las bobinas generan picos de tensión cuando son conmutadas).

- Para establecer comunicación entre la Raspberry y el servidor, primero se debe conectar a internet las dos partes; la Raspberry se conecta a internet por medio de una red privada suministrada desde un Smart Phone mediante WIFI, y el servidor se enlaza a internet por medio de un cable de red a través del puerto Ethernet.

## 4.2 RECOMENDACIONES

- Es importante estudiar a fondo el funcionamiento de la Raspberry Pi 3 B+, cámara *PiNoir*, *GPS* y *LCD touch screen display*, minimizando errores y posibles daños al momento de su conexión, y manipulación. Por su elevada sensibilidad y vulnerabilidad a cargas electrostáticas, pueden presentar errores e incluso, peligrar la integridad física de los mismos.
- Es importante considerar que antes de proceder a la energización completa del módulo, ya sea mediante vía USB o PoE, previamente se debe conectar todos los periféricos externos que se vayan a utilizar. Caso contrario, esto provocará fallos en la ejecución de los programas y posibles daños en el módulo *Raspberry*.
- Es estrictamente necesario usar una tarjeta SD clase 10, con capacidad mínima de 32 GB, para tener una lectura y escritura rápida. En esta SD se aloja todo el sistema operativo y la base de datos. Además, se debe contar con disipadores de calor en la tarjeta Raspberry, ya que el sistema opera en modo turbo a una velocidad de 1 Ghz. Adicionalmente, los puertos GPIO y USB trabajan a 3.3 VDC, con un límite de corriente de 16 mA. En vista de esto, es importante contar con una fuente de polarización superior a 2.5 A; caso contrario, el sistema se resetea constantemente.



- En la librería OpenCV, una gran parte de los algoritmos trabajan en escala de grises. Por lo tanto, es recomendable emplear una cámara PiNoir V2, ya que esta trabaja en escala de grises; además, a esta cámara se le debe incorporar una lámpara de LEDs infrarrojos, que permitirán capturar y grabar imágenes en absoluta oscuridad.
- De acuerdo a múltiples investigaciones de tipo experimental, se sugiere trabajar con el algoritmo LBP, mismo que realiza la identificación de rasgos faciales más rápido que *Haar*, *EigenFaces*, *FisherFaces* y otros que son propios de la librería *OpenCV*.

## 5. REFERENCIAS BIBLIOGRÁFICAS

- [1] R. Justicia, «4.195 vehículos fueron robados en el país de enero a noviembre de 2018,» El Telégrafo, 03 01 2019. [En línea]. Available: <https://www.eltelegrafo.com.ec/noticias/judicial/12/cifras-vehiculos-robados-ecuador2018>. [Último acceso: 04 01 2019].
- [2] A. Ibañez, «Un problema de seguridad afecta a las llaves de millones de coches,» El País, Tecnología, 18 08 2016. [En línea]. Available: [https://elpais.com/tecnologia/2016/08/17/actualidad/1471425958\\_156523.html](https://elpais.com/tecnologia/2016/08/17/actualidad/1471425958_156523.html). [Último acceso: 04 01 2019].
- [3] E. Metro, «Más de 13 mil robos de bienes, accesorios, autopartes y vehículos,» 22 12 2017. [En línea]. Available: <https://www.metroecuador.com.ec/ec/noticias/2017/12/22/robo-autos.html>. [Último acceso: 21 12 2018].
- [4] D. Bravo, «Bandas delictivas utilizan controles universales de alarmas para robar vehículos en Quito, según la Policía,» El Comercio, 10 05 2018. [En línea]. Available: <https://www.elcomercio.com/actualidad/quito-bandasdelictivas-controlesuniversales-robo-vehiculos.html>. [Último acceso: 04 01 2019].
- [5] H. Herrero, «Raspberry Pi – Reconocimiento facial,» Raspberry Pi, 17 09 2016. [En línea]. Available: <http://www.bujarra.com/raspberry-pi-reconocimiento-facial/>. [Último acceso: 04 01 2019].
- [6] R. Velasco , «Review: Raspberry Pi 3 Modelo B+,» HARDZONE, 02 07 2018. [En línea]. Available: <https://hardzone.es/2018/07/02/review-raspberry-pi-3-modelo-b/>. [Último acceso: 05 01 2019].
- [7] N. Heath, «Raspberry Pi 3 Model B+: See all the new features on the board,» TechRepublic, 14 03 2018. [En línea]. Available: <https://www.techrepublic.com/pictures/raspberry-pi-3-model-b-see-all-the-new-features-on-the-board/>. [Último acceso: 05 01 2019].
- [8] J. Pastor, «Raspberry Pi 3 Model B+, análisis: más potencia y mejor WiFi para un miniPC que sigue asombrando,» Xataka, 16 04 2018. [En línea]. Available: Raspberry Pi 3 Model B+, análisis: más potencia y mejor WiFi para un miniPC que sigue asombrando. [Último acceso: 05 01 2019].
- [9] «Cortex-A53,» armDeveloper, 23 03 2018. [En línea]. Available: <https://developer.arm.com/products/processors/cortex-a/cortex-a53>. [Último acceso: 06 01 2019].
- [10] J. Jacques, «Acceder a un puerto GPIO con Python en su Raspberry Pi 3,» Arrow Electronics, 07 06 2018. [En línea]. Available: <https://www.arrow.com/es-mx/research-and-events/articles/raspberry-pi-gpio>. [Último acceso: 06 01 2019].
- [11] M. Contreras, «Raspberry Pi 3 B+, el ordenador más pequeño y ahora mucho más rápido,» Clipset, 05 03 2018. [En línea]. Available: <https://clipset.20minutos.es/raspberry-pi-3-b-plus/>. [Último acceso: 06 01 2019].

- [12] makerlab electronics, «makerlab electronics,» [En línea]. Available: <https://www.makerlab-electronics.com/product/raspberry-pi-noir-infrared-camera/>. [Último acceso: 7 Enero 2019].
- [13] sparkfun, «sparkfun,» [En línea]. Available: <https://www.sparkfun.com/products/14776>. [Último acceso: 8 Enero 2019].
- [14] A. Novikova, «Blog Electrónica Radical,» Nexgendesing, 05 06 2017. [En línea]. Available: <https://electronicaradical.blogspot.com/2014/08/sistema-de-posicionamiento-global-gps.html>. [Último acceso: 06 01 2019].
- [15] «Ublox NEO-6M GPS Module,» Sunfounder, 17 03 2017. [En línea]. Available: [http://wiki.sunfounder.cc/index.php?title=Ublox\\_NEO-6M\\_GPS\\_Module](http://wiki.sunfounder.cc/index.php?title=Ublox_NEO-6M_GPS_Module). [Último acceso: 06 01 2019].
- [16] «Tipos de Diodos Led,» FactorLed, 23 01 2018. [En línea]. Available: <https://www.factorled.com/blog/es/tipos-de-diodos-led-caracteristicas-tecnicas-y-formatos/>. [Último acceso: 06 01 2019].
- [17] B. Solarte, «Resistencia Eléctrica,» 13 08 2015. [En línea]. Available: <http://mairaresistenciaelectrica.blogspot.com/>. [Último acceso: 06 01 2019].
- [18] V. García , «Electrónica Aplicada,» 13 11 2010. [En línea]. Available: <https://www.diarioelectronicohoy.com/blog/pulsadores-sin-rebotes>. [Último acceso: 06 01 2019].
- [19] A. Ahumada, «Características principales de un Relé,» Electricidad de Alejandro, 13 07 2008. [En línea]. Available: <http://electroalejo.blogspot.com/2008/07/caracteristicas-principales-de-un-rele.html>. [Último acceso: 06 01 2019].
- [20] «El Transistor NPN – estructura y aplicaciones,» Hetpro, 20 10 2017. [En línea]. Available: <https://hetpro-store.com/TUTORIALES/transistor-npn/>. [Último acceso: 06 01 2019].
- [21] J. Gomar , «Samsung EVO 32 GB Review,» Profesional Review, 15 12 2015. [En línea]. Available: <https://www.profesionalreview.com/2015/12/22/samsung-evo-micro-sd-32-gb-review/>. [Último acceso: 06 01 2019].
- [22] J. Lopez, «Los mejores sistemas operativos para instalar en Raspberry Pi,» Hipertextual, 25 05 2018. [En línea]. Available: <https://hipertextual.com/2018/05/sistemas-operativos-instalar-raspberry-pi>. [Último acceso: 06 01 2019].
- [23] A. Rodríguez, «OpenCV: Librería de Visión por Computador,» Oficina de software libre, 15 08 2015. [En línea]. Available: <https://osl.ull.es/software-libre/opencv-libreria-vision-computador/>. [Último acceso: 06 01 2019].
- [24] O. Org, «OpenCV,» 2018. [En línea]. Available: <https://opencv.org/about.html>. [Último acceso: 10 01 2019].
- [25] A. Fernández, «Principales Características,» de *Python al descubirto*, España, RC Libros, 2013, pp. 3-6.

- [26] R. H. E. L. 4, «Protocolo SSH,» [En línea]. Available: <https://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ssh.html>. [Último acceso: 11 01 2019].
- [27] Raspberrypi.org, «VNC,» [En línea]. Available: <https://www.raspberrypi.org/documentation/remote-access/vnc/>. [Último acceso: 11 01 2019].
- [28] Centos.org, «Planeta centos,» 08 01 2019. [En línea]. Available: <https://planet.centos.org/>. [Último acceso: 11 01 2019].
- [29] Internetya, «Servidores Linux,» 21 05 2018. [En línea]. Available: <https://www.internetya.co/servidores-linux-ventajas-del-sistema-operativo-centos-7/>. [Último acceso: 11 01 2019].
- [30] Ionos.es, «Servidor LAMP: una solución económica para webs dinámicas,» Digital Guide, [En línea]. Available: <https://www.ionos.es/digitalguide/servidores/know-how/servidor-lamp-la-solucion-para-webs-dinamicas/>. [Último acceso: 11 01 2019].
- [31] Ymant, «LAMP,» [En línea]. Available: <https://www.ymant.com/blog/lamp>. [Último acceso: 11 01 2019].
- [32] R. Alfonso, «¿Qué es un Hosting?,» Mi posicionamiento web, 05 11 2018. [En línea]. Available: <https://miposicionamientoweb.es/que-es-un-hosting/>. [Último acceso: 11 01 2019].
- [33] phpMyAdmin, «phpMyAdmin,» [En línea]. Available: <https://www.phpmyadmin.net/>. [Último acceso: 11 01 2019].
- [34] C. L, «HTML,» Ciber Corresponsales, 26 05 2018. [En línea]. Available: <https://www.cibercorresponsales.org/perfiles/labalsa02/blogs/1--html-%C2%BFqu%C3%A9-es-y-para-que-sirve>. [Último acceso: 11 01 2019].
- [35] D. Pérez, «¿Qué es Javascript?,» Maestros del web, 03 06 2007. [En línea]. Available: <http://www.maestrosdelweb.com/que-es-javascript/>. [Último acceso: 15 01 2019].
- [36] M. Calatayud, «Direcciones IP Públicas y privadas,» Informática, 24 01 2012. [En línea]. Available: <http://informaticaredes123.blogspot.com/2012/01/direcciones-ip-publicas-y-privadas.html>. [Último acceso: 15 01 2019].
- [37] J. Lopez, «Los servidores DNS,» 30 06 2016. [En línea]. Available: <https://prezi.com/lpzft6xs2qd7/los-servidores-dns-usos-caracteristicas-y-configuracion/>. [Último acceso: 15 01 2019].
- [38] gasexpress, «El reconocimiento facial llega al automóvil,» 17 01 2017. [En línea]. Available: <http://gasexpress.es/blog/el-reconocimiento-facial-llega-al-automovil/>. [Último acceso: 16 01 2019].
- [39] E. Marín, «reconocimiento facial,» 18 03 2018. [En línea]. Available: <https://es.gizmodo.com/el-nuevo-coche-de-subaru-tiene-reconocimiento-facial-pa-1824160811>. [Último acceso: 16 01 2019].

- [40] A. Lopez, «Local Binary Patterns,» 2019. [En línea]. Available: <https://es.coursera.org/lecture/deteccion-objetos/l2-2-a-local-binary-patterns-lzZui?fbclid=IwAR3rDp0Hw9I-508PBLWd6ghyGwjGi78XID7m9KLGPs7kvKQwqq1-b9TXiNI>. [Último acceso: 16 01 2019].
- [41] C. Troya , «Local Binary Patterns,» 26 02 2016. [En línea]. Available: [https://cesartroyasherdek.wordpress.com/2016/02/26/ejercicio-local-binary-patterns/?fbclid=IwAR3F0lc8x0b\\_j0miNO0if0cVxNrgWlhB2xvihk1AGAMeLmLAS0Uij-yDang](https://cesartroyasherdek.wordpress.com/2016/02/26/ejercicio-local-binary-patterns/?fbclid=IwAR3F0lc8x0b_j0miNO0if0cVxNrgWlhB2xvihk1AGAMeLmLAS0Uij-yDang). [Último acceso: 16 01 2019].
- [42] D. Acosta, «Instrucciones para actualizar Raspbian,» 03 02 2018. [En línea]. Available: <https://www.deacosta.com/instrucciones-para-actualizar-raspbian-8-jessie-raspbian-9-stretch-en-raspberry-pi/>. [Último acceso: 28 01 2019].
- [43] «Pimoroni,» [En línea]. Available: <https://shop.pimoroni.com/products/raspberry-pi-3-b-plus>. [Último acceso: 04 01 2019].
- [44] A. Navarro, «Todo sobre el GPIO de la Raspberry Pi,» Comohacer.eu, 18 12 2018. [En línea]. Available: <https://comohacer.eu/gpio-raspberry-pi/>. [Último acceso: 06 01 2019].
- [45] «Memoria Micro SD 32GB,» Electronilab, [En línea]. Available: <https://electronilab.co/tienda/memoria-micro-sd-32gb-sandisk-ultra-uh-iclasa-10-80mbs-adaptador/>. [Último acceso: 06 01 2019].
- [46] «Ya está disponible para descargar la nueva actualización de Raspbian,» Desde Linux, 30 06 2018. [En línea]. Available: <https://blog.desdelinux.net/ya-esta-disponible-para-descargar-la-nueva-actualizacion-de-raspbian/>. [Último acceso: 06 01 2019].
- [47] Euronews, «Diseñando los vehículos del futuro,» 29 04 2014. [En línea]. Available: <https://es.euronews.com/2014/04/29/disenando-los-vehiculos-del-futuro>. [Último acceso: 16 01 2019].
- [48] random nerd tutorials, «random nerd tutorials,» [En línea]. Available: <https://randomnerdtutorials.com/guide-to-raspberry-pi-camera-v2-module/>. [Último acceso: 7 Enero 2019].
- [49] Garmin, «Garmin,» [En línea]. Available: <https://www8.garmin.com/aboutGPS/>. [Último acceso: 7 Enero 2019].
- [50] Bot Science , «Bot Science,» [En línea]. Available: [http://botscience.net/store/index.php?route=product/product&product\\_id=73](http://botscience.net/store/index.php?route=product/product&product_id=73). [Último acceso: 7 Enero 2019].
- [51] sunfounder, «sunfounder,» [En línea]. Available: [http://wiki.sunfounder.cc/index.php?title=Ublox\\_NEO-6M\\_GPS\\_Module](http://wiki.sunfounder.cc/index.php?title=Ublox_NEO-6M_GPS_Module). [Último acceso: 7 Enero 2019].
- [52] «Raspberry Pi NoIR Infrared Camera Board V2 (8MP,1080p),» Makerlab Electronic, 03 06 2018. [En línea]. Available: <https://www.makerlab-electronics.com/product/raspberry-pi-noir-infrared-camera/>. [Último acceso: 06 01 2019].

- [53] «LCD Touchscreen HAT for Raspberry Pi,» Sparkfun, 14 05 2018. [En línea]. Available: <https://www.sparkfun.com/products/14776>. [Último acceso: 06 01 2019].
- [54] HARDZONE, «HARDZONE,» [En línea]. Available: <https://hardzone.es/reviews/perifericos/analisis-raspberry-pi-3-modelo-b/>. [Último acceso: 5 Febrero 2019].

# **ANEXOS**

# **ANEXOS**

**ANEXO A: TEORÍA COMPLEMENTARIA**

**ANEXO B: MANUAL DE USUARIO**

**ANEXO C: MANUAL DE MANTENIMIENTO**



## ANEXO A

### TEORIA COMPLEMENTARIA

#### ANEXO A1: Chips de la Raspberry Pi 3 Modelo B+

Broadcom BCM2837B0 1.4Hz – 64bit (Figura A1.1), en el cual se encuentra un disipador y una chapa con el logo de Raspberry Pi, abajo de la cual está ubicado el chip red de Adafruit. Este chip proporciona:

- Wi-Fi de doble banda
- Bluetooth 4.2
- LAND7515 (es el chip encargado de controlar la tarjeta de red, así como los puertos USB)



Figura A1.1 Procesador BCM2837B0.

Elpida B8132B4PB-8D-F (Figura A1.2), es el chip encargado de suministrar 1GB de RAM al módulo Raspberry Pi.



Figura A1.2 Elpida B8132B4PB-8D-F.

En la tabla A1.1, se presenta un cuadro de las características de hardware de la que está integrada la Raspberry Pi 3 B+

Tabla A1.1 Características de hardware de Raspberry Pi3 B+. [42]

SoC	Broadcom BCM2837B0
CPU	4× Cortex-A53 1.4 GHz
Set de instrucciones	ARMv8-A (64/32-bit)
RAM	1 GB SDRAM
Almacenamiento	Micro SD
Ethernet	10/100/1000 Mbit/s
USB	4 puertos
Inalámbrico	802.11b/g/n/ac - Bluetooth 4.2
Salida de Video	HDMI
Salida de Audio	HDMI/ Auriculares
GPIO	40 Pines

## ANEXO A2: Instalación del módulo GPS

Para obtener los datos tanto de longitud como de latitud de cualquier punto de la Tierra por medio del módulo GPS a la Raspberry Pi, es necesario además de instalar la biblioteca GPS Daemon (DSGP) y conexiones del GPS a la Raspberry Pi, el uso de las siguientes configuraciones en la terminal.

En primera instancia, se actualiza la Raspberry Pi y luego se reinicia el sistema con la siguiente configuración.

```
apt-get update sudo
sudo apt-get upgrade
sudo reboot
```

Para la configuración de la UART en la Raspberry Pi, se edita el archivo config.txt y se añade las siguientes líneas a dicho archivo.

```
sudo nano /boot/config.txt
dtparam = spi = on
dtoverlay = PI3-disable-bt
core_freq = 250
enable_uart = 1
force_turbo=1
```

El comando, `force_turbo` habilita UART para utilizar la frecuencia de la base máxima, que en este caso es 250 Hz, así se asegura la coherencia e integridad de los datos recibidos.

El comando `dtoverlay = PI3-disable-bt` desconecta el bluetooth del `ttyAMA0` para así permitir el acceso completo a UART. Es importante antes de guardar el archivo editado, este sea revisado por los posibles errores tipográficos que puedan existir. Para terminar la configuración de UART, se edita el archivo `cmdline.txt`, sustituyendo el contenido de este archivo por lo siguiente.

```
dwc_otg.lpm_enable = 0 = raíz de la consola tty1 = / dev / mmcblk0p2
rootfstype = ext4 ascensor = plazo fsck.repair = sí rootwait tranquila
salpicaduras plymouth.ignore serie-consolas
```

Antes de reiniciar el sistema, es necesario desactivar el servicio Getty; se usó la siguiente configuración.

```
sudo systemctl serial-getty@ttyS0.service parada
desactivar sudo systemctl serial-getty@ttyS0.service
```

Finalmente, se debe reiniciar el sistema.

## **Instalación de Minicom y Pynmea2**

Antes de proceder a instalar estos paquetes, se realiza la activación `ttyAMA0` con la siguiente configuración.

```
sudo systemctl permiten serial-getty@ttyAMA0.service
```

La instalación de Minicom y Pynmea2 se usa para probar y analizar los datos recibidos por el módulo GPS, verificando que esté trabajando correctamente; para su instalación se usaron los siguientes comandos.

```
sudo apt-get install minicom
sudo PIP instalar pynmea2
```

Es importante que la velocidad de transmisión esté regulada a 96000 baudios para lograr la comunicación del módulo GPS, usando la siguiente configuración.

```
sudo minicom -D / dev / ttyAMA0 -b9600
```

Finalmente, se reinicia el sistema.

## ANEXO B: MANUAL DE USUARIO

El proyecto consta de un módulo que contiene la cámara PiNoir. Dicha cámara tiene la ventaja que permite capturar imágenes en el día y en la noche, dado que es una cámara que toma imágenes infrarrojas, con la ayuda de la lámpara de LEDs infrarrojos, esto se puede observar en la Figura B.1.



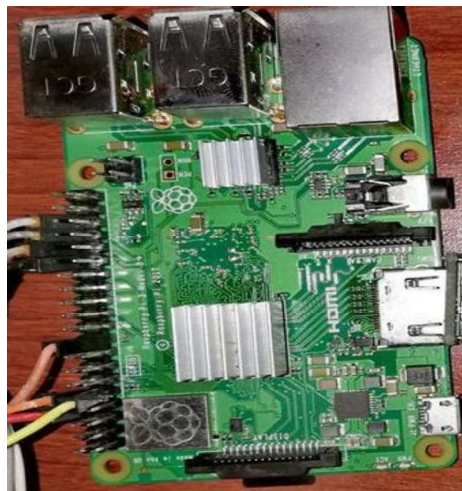
*Figura B.1 Módulo del sistema.*

## COMPONENTES

El módulo consta de los siguientes componentes.

- Raspberry Pi 3B+

Tarjeta de desarrollo (Figura B.2).



*Figura B.2 Raspberry Pi 3B+.*

- Tarjeta SD de 32 GB clase 10

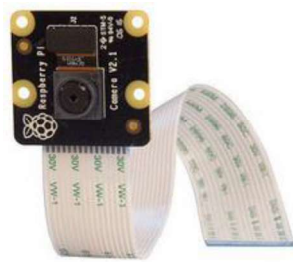
Esta tarjeta sirve como disco duro para la Raspberry (Figura B.3).



*Figura B.3 Micro SD clase 10. [24]*

- Cámara PiNoir

Se utiliza para capturar imágenes y se conecta al módulo mediante el puerto CSI (Figura B.4).



*Figura B.4 Cámara PiNoir. [13]*

- Lámpara LED

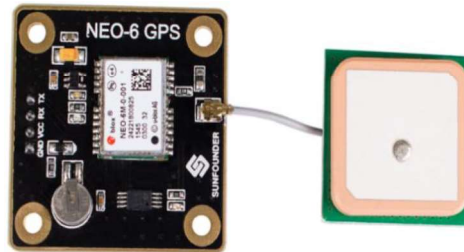
Proporciona luz para obtener una mejor iluminación del rostro enfocado en la cámara (Figura B.5). Para su alimentación se la conecta al banco de baterías por medio de su terminal USB.



*Figura B.5 Lámpara LED.*

- Receptor GPS UBLOX NEO-6M

Permite obtener los datos emitidos por los satélites de posicionamiento global (Figura B.6).



*Figura B.6 GPS. [15]*

- Power Bank

Suministra el voltaje de 5v y una corriente de 3,4A, que sirve para alimentar al módulo del sistema (Figura B.7).



*Figura B.7 Banco de baterías.*

- LED indicador

Diodo emisor de luz de color verde, indica que el reconocimiento al usuario ha sido exitoso (Figura B.8).



*Figura B.8 LED indicador.*

- Interruptor eléctrico o Switch

Permite el inicio y finalización del sistema de reconocimiento facial (Figura B.9).



*Figura B.9 Switch.*

- Soporte de celular para auto

Permite sostener y ajustar el módulo al parabrisas del auto. (Figura B.10)



*Figura B.10 Soporte de celular.*

- Circuito electrónico de control

La placa (Figura B.11) permite activar el motor de arranque del vehículo a través del reconocimiento facial del usuario y así encender el auto.



*Figura B.11 Placa de circuito de control electrónico.*

## CONEXIONES

El módulo funciona con 5 VDC/2.4 A, este voltaje está presente en el banco de baterías (Figura B.12) y se debe conectar a la Raspberry.



*Figura B.12 Conexiones del banco de baterías.*

La tarjeta micro SD, se coloca en el orificio posterior del módulo (Figura B.13).



*Figura B.13 Conexión Micro SD.*

La cámara va conectada al puerto CSI de la Raspberry, a través del cable plano de 15 pines (Figura B.14).



*Figura B.14 Conexión cámara.*



La lámpara de LEDs está conectada al banco de baterías al puerto USB de 5VDC/1A (Figura B.15).



Figura B.15 Puertos del banco de baterías.

El módulo GPS se coloca directamente en el puerto UART de la Raspberry (Figura B.16).

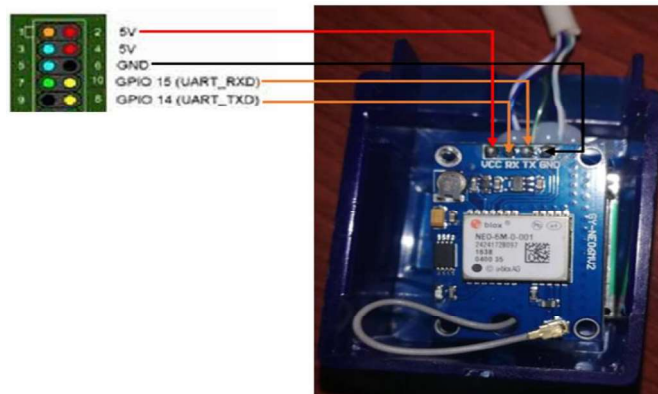


Figura B.16 Conexión módulo GPS.

La placa del circuito electrónico de control (Figura B.17) es alimentada por 12 VDC, se conecta al puerto GPIO 12 (pin 32) de la Raspberry a la entrada de dicha placa, y su salida al terminal positivo del motor de arranque del vehículo.

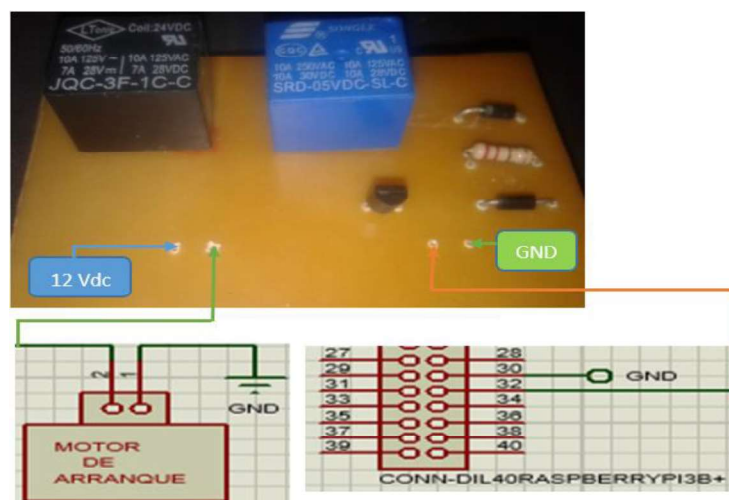


Figura B.17 Conexiones placa de circuito electrónico de control.

El Switch (Figura B.18) que inicia y finaliza el sistema de reconocimiento facial, se conecta al puerto GPIO 16 (pin 36).

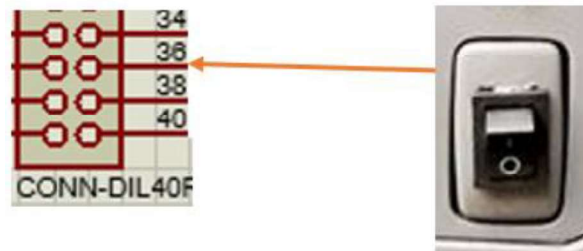


Figura B.18 Conexión Switch.

## FUNCIONAMIENTO

### Inicialización y conexión del módulo a la red móvil

Una vez el usuario haya ingresado al vehículo, se procederá a encender la fuente de alimentación (*Power Bank*) del módulo, este se enciende y se debe esperar que el *Smart Phone* establezca una conexión con la red móvil; en promedio, esto tarda 1 minuto aproximadamente. Cuando se haya establecido la conexión de la red móvil con el módulo, se indicará en la pantalla del *Smart Phone* el nombre de la red "*Raspberry*". Esto se observa en la Figura B.19.

Ya establecida la conexión, el sistema está listo para funcionar.



Figura B.19 Red Privada.

## Arranque del sistema

Para el arranque del sistema, una vez que se haya establecido la conexión del módulo con la red móvil, se procederá a colocar la llave en la posición de contacto (Figura B.20).



*Figura B.20 Llave del auto.*

Posteriormente, como se observa en la Figura B.21, el *Switch* conectado al módulo, encenderá la cámara y pantalla, dando inicio al reconocimiento facial siempre y cuando este sea accionado.



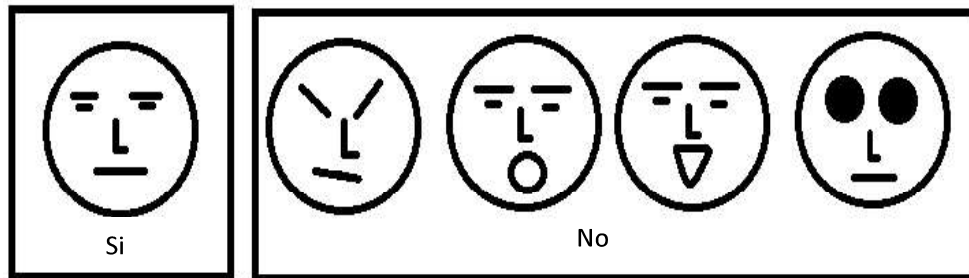
*Figura B.21 Switch adaptado al auto.*

## La distancia facial

La distancia recomendada entre la persona y la cámara del módulo es de 0.3m a 0.6m. Si la imagen se encuentra muy brillante o muy oscura se deberá moverse hasta que se pueda enfocar correctamente.

## Expresión facial

En la Figura B.22 se muestra la expresión facial recomendada y las expresiones faciales inapropiadas para el uso del sistema. Tanto en la etapa de captura, y reconocimiento es necesario mantener una expresión facial normal.



*Figura B.22 expresiones faciales*

## Reconocimiento facial

Una vez accionado el Switch, se encenderá la cámara y pantalla del módulo; de esta forma se podrá visualizar al usuario, e iniciará el reconocimiento facial. Como se puede observar en la Figura B.23, en la pantalla se despliega un temporizador de 15 segundos puesto que el reconocimiento se realizará cada 1 segundo.



*Figura B.23 Temporizador en pantalla.*

En el caso que se tome un rostro por 15 veces, de las cuales 10 de estas hayan sido correctas, el LED se encenderá por 1 segundo indicando un reconocimiento facial exitoso, y en la pantalla se visualizará el mensaje "Bienvenido Daniel"; por consiguiente, se encenderá el auto.

Esto se puede observar en la Figura B.24



*Figura B.24 Reconocimiento facial correcto.*

De lo contrario, si el rostro no es identificado por 15 veces, dicho contador se reiniciará automáticamente en un ciclo cerrado indefinidamente, hasta que, haciendo uso del Switch se detenga el sistema de reconocimiento facial.

### **Ubicación GPS**

La ubicación GPS se dividió en dos etapas, la primera consiste, en enviar una notificación de alerta por correo electrónico al usuario, indicando que, el auto ha sido encendido; simultáneamente, se adjuntan las coordenadas actuales del auto. La segunda etapa es la del rastreo continuo del auto, por medio de la página web: <https://geofacecar.epn.edu.ec/>.

### **Notificaciones de alertas**

Las alertas notificarán la ubicación actual del auto; adicionalmente, señalan que el mismo ha sido encendido. Estas alertas se remitirán mediante Gmail al correo electrónico del usuario, como se puede observar en la Figura B.25.

- Correo electrónico remitente

Correo electrónico: [reconocimiento.facial.tesis@gmail.com](mailto:reconocimiento.facial.tesis@gmail.com)

Contraseña: esfot2018

- Correos electrónicos destinatarios

Correo electrónico: danielperez@live.com, luis\_car@hotmail.com

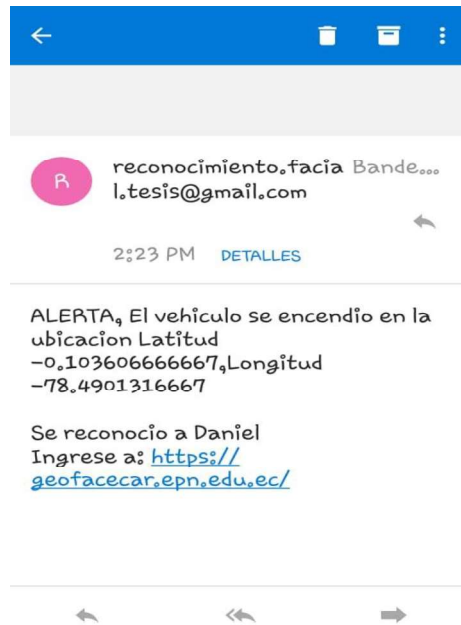


Figura B.25 Mensaje de correo electrónico al usuario.

### Página web

La dirección de la página web es: <https://geofacecar.epn.edu.ec/>, para acceder a ella es necesario ingresar el usuario "admin" y la contraseña "admin", para así, visualizar la ubicación actual del auto en la API de Google Maps. En la Figura B.26. Se observa la página para iniciar sesión.

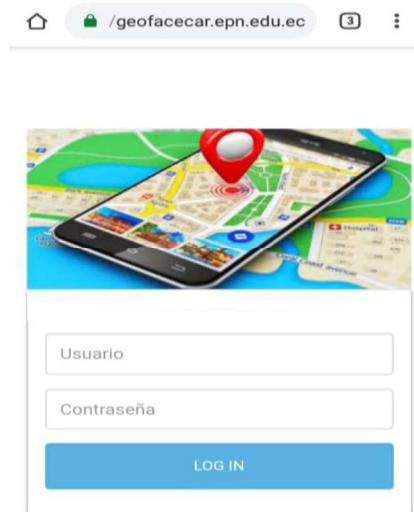


Figura B.26 Página de login.

Se observa en la Figura B.27. La página de inicio en la cual se visualiza el logotipo de “Localización dispositivo”.



## Localización Dispositivo



Figura B.27 Página de inicio.

En el menú opciones (Figura B.28) se puede acceder a la opción “Localización” donde se encuentra “Mapa”.

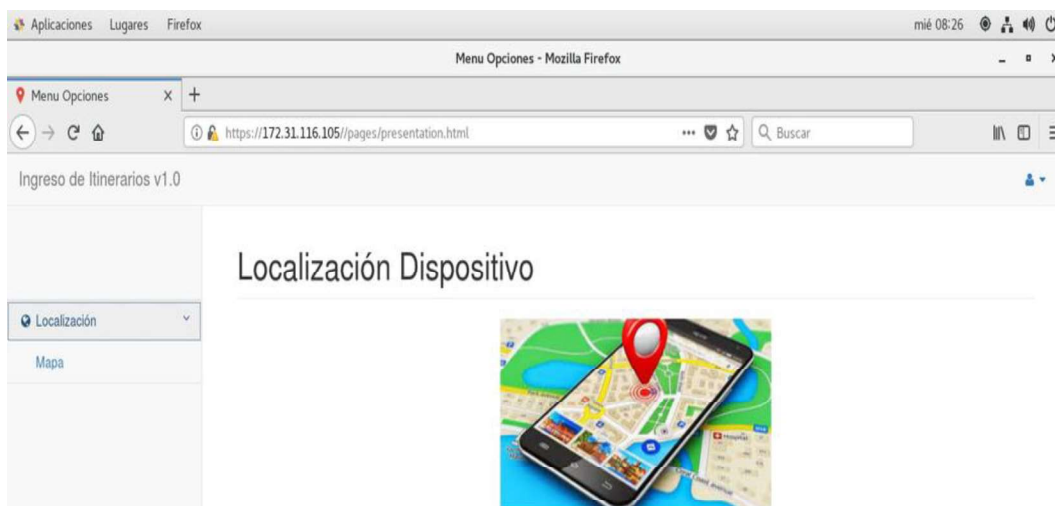


Figura B.28 Menú opciones.

Ingresando a la opción “Mapa” con la ayuda de la aplicación Google Maps API, se visualizará la ubicación actual del auto (Figura B.29), esta ubicación se actualizará cada 15 segundos, para poder navegar en el mapa.

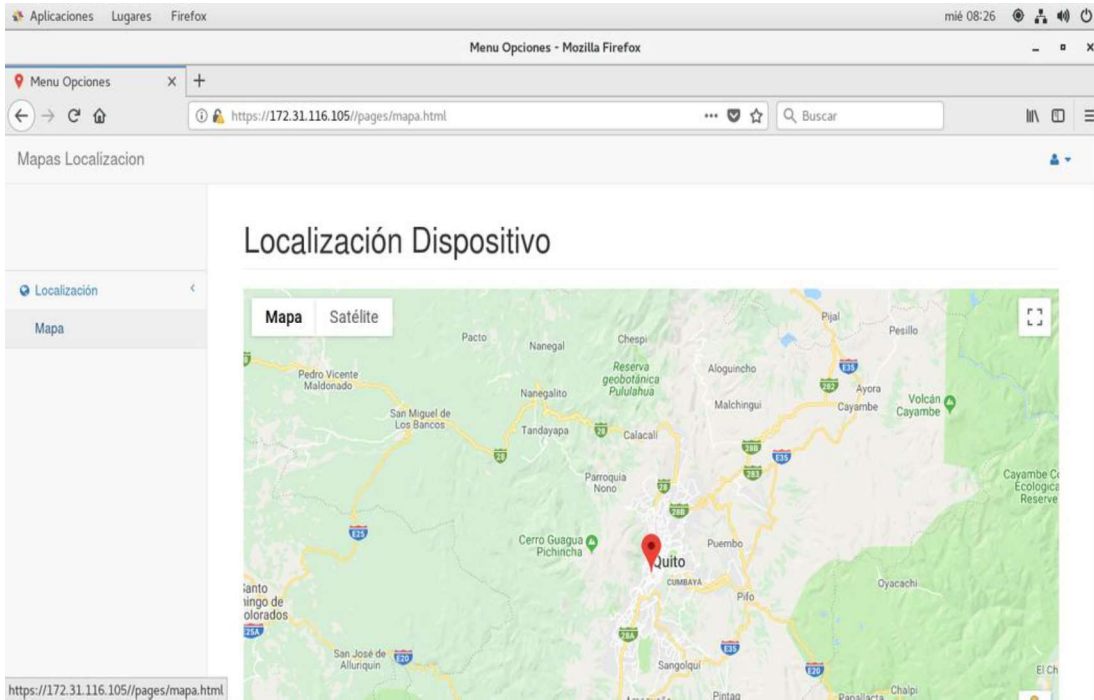


Figura B.29 Visualización del mapa.



## **ANEXO C: MANUAL BÁSICO DE MANTENIMIENTO**

### **ANEXO C1: MANTENIMIENTO DE HARDWARE**

#### **RASPBERRY PI 3 B+**

- Elementos de conexión
- Disipadores de calor
- Cables de conexión

#### **SERVIDOR WEB**

- Mantenimiento de Hardware del servidor web
- Limpieza externa del servidor web
- Limpieza interna del servidor web

### **ANEXO C2: MANTENIMIENTO DE SOFTWARE**

#### **RASPBERRY PI 3 B+**

- Actualizaciones mensuales del sistema operativo

#### **SERVIDOR LAMP**

- Actualizaciones semestrales del sistema operativo
- Copias de seguridad
- Asegurarse si se actualizó el sistema
- Comprobar el consumo de espacio del CPU
- Cambio de contraseñas

## ANEXO C1: MANTENIMIENTO DE HARDWARE

### RASPBERRY PI 3 B+

#### Elementos de conexión:

#### Limpieza física del módulo

Para empezar, se debe encontrar un espacio cómodo, amplio e iluminado para trabajar, así mismo contar con las herramientas necesarias para llevar a cabo la limpieza. Entre las herramientas necesarias son: brocha mediana, franela, una pulsera antiestática y una botella de aire comprimido o secador de cabello de aire frío.

Ahora bien, se debe apagar y desconectar el módulo del banco de baterías; posteriormente, se desconecta la pantalla LCD y cámara PiNoir de la Raspberry Pi. Esto se observa en la Figura C1.1.



*Figura C1.1 Desconexión de cámara PiNoir y pantalla LCD.*

Lo primero será descargar aire a presión o comprimido sobre los periféricos de la Raspberry Pi, y con una brocha mediana se debe limpiar los bordes exteriores para eliminar todo tipo de partículas de suciedad presentes en el ambiente; esto removerá una buena cantidad de polvo acumulado, en la Figura C1.2 se observa la acumulación de residuos en dichos puertos.



*Figura C1.2 Puertos de Raspberry Pi.*

Es recomendable no soplar con la boca ya que pueden expedirse partículas de saliva y así causar un corto circuito debido a su conductividad.

Para evitar daños en algunos de los componentes debido a descargas o corrientes de arco, se debe usar una pulsera antiestática; en caso de no contar con una, se puede descargar, tocando una pieza metálica que esté conectada a tierra por lo menos 5 segundos.

Es necesario inspeccionar las conexiones y tener un bosquejo, fotografía o diagrama de las mismas. En caso de notar algún deterioro en el cable, este debe ser cambiado por uno en mejor estado.

### **Disipadores de calor**

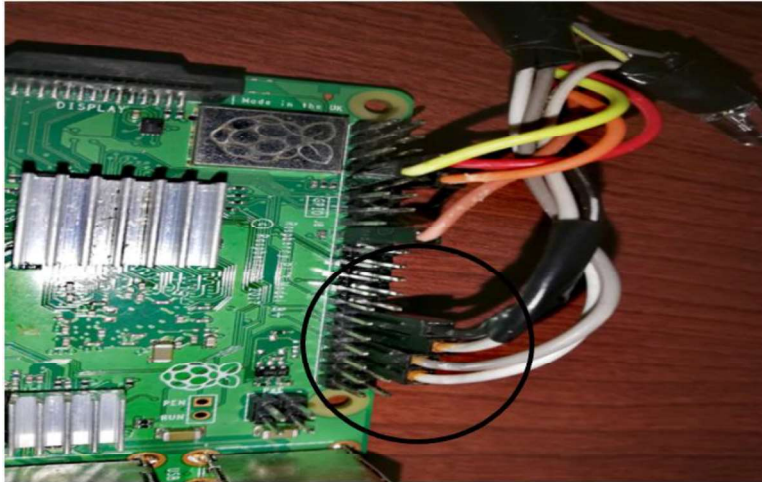
Es necesario que se verifique que los disipadores de calor se encuentren pegados a los chips, de lo contrario se deben reemplazar por unos en mejores condiciones. En la figura C1.3 se observa los disipadores de calor en buen estado.



*Figura C1.3 disipadores de calor.*

### **Cables de conexión**

Debido a la manipulación de los cables, estos pueden presentar fisuras internas. Es recomendable usar un multímetro para verificar el estado de los mismos. Si estos presentan irregularidades, deben ser reemplazados por cables de las mismas especificaciones técnicas, en la Figura C1.4 se observa los cables en mal estado.



*Figura C1.4 cables en mal estado.*

## **SERVIDOR WEB**

### **Mantenimiento de Hardware del servidor web**

Para empezar, se debe encontrar un espacio cómodo, amplio e iluminado para trabajar.

El mantenimiento consiste en dar una limpieza a la PC. Cuando se trabaje con este equipo electrónico se debe cuidar la carga estática, ya que puede causar daños al procesador; para evitar este tipo de inconvenientes se debe contar con las siguientes herramientas: manilla antiestática, tapa bocas, destornillador estrella, una brocha mediana, spray limpiador de pantallas, limpiadores, un borrador y una botella de aire comprimido o secador de cabello de aire frío.

### **Limpieza externa del servidor web**

Se procede a desconectar los cables de alimentación eléctrica de la PC, así como también los periféricos de salida como son: teclado, mouse, parlantes, entre otros. Se utiliza un limpión para retirar el polvo acumulado de la carcasa del equipo y elementos periféricos. Se puede usar el secador de cabello de aire frío en el teclado para así eliminar el polvo acumulado en las teclas.

### **Limpieza interna del servidor web**

Para realizar esta etapa hay que tener la seguridad de descargar la energía estática, se puede tocar 5 segundos un objeto metálico que esté conectado a tierra o usando la manilla antiestática.

Con la ayuda de un destornillador estrella, se destapa la tapa del CPU; una vez destapada, se debe retirar la batería de la tarjeta madre y continuar con la desconexión de los componentes.

Se utiliza una brocha mediana para retirar el polvo acumulado en las tarjetas electrónicas y luego rociar con spray para limpiar los contactos.

En la Figura C1.5, se puede observar que una de las partes donde más se acumula polvo es en la fuente de poder, específicamente en su ventilador. Para limpiarlo, se usa una botella de aire comprimido o secador de cabello de aire frío.



*Figura C1.5 Acumulación de polvo en fuente de poder.*

Los puertos y ranuras se barren con una brocha y se rocía spray limpia contactos, así como también a los componentes electrónicos. Tener precaución de no doblarlos o desprenderlos de su base.

Finalmente, una vez limpias todas las piezas internas del servidor web se procede con el ensamble. Conectar el cableado y colocar todas las piezas en su lugar. Cuando se termina de ensamblar, antes de cerrar la cubierta del CPU, es preferible realizar una prueba de funcionamiento.

Una vez terminada la prueba de funcionamiento y esta es satisfactoria, se cierra la cubierta y enciende la PC.

## ANEXO C2: MANTENIMIENTO DE SOFTWARE

### RASPBERRY PI 3 B+

#### Actualizaciones trimestrales del sistema operativo

Para la actualización del sistema operativo Raspbian es necesario usar los siguientes comandos mostrados en la Tabla C2.1. Dichos comandos se ejecutan en la terminal de la Raspberry Pi.

Tabla C2.1 Lista de comando para la actualización de paquetes en Raspbian. [42]

<code>sudo apt-get update</code>	Actualiza el listado de paquetes disponibles, buscando en los repositorios instalados en Raspbian.
<code>sudo apt-get upgrade</code>	Actualiza los paquetes ya instalados en el sistema.
<code>sudo apt-get dist-upgrade</code>	Usado para resolver algunas dependencias que no se instalan con upgrade.
<code>sudo apt-get install &lt;paquete&gt;</code>	Instala el paquete que se indique.
<code>sudo apt-get remove</code>	Desinstala el paquete que se indique.
<code>sudo apt-show-versions (-u)</code>	Enlista paquetes ya instalados.
<code>sudo apt-get autoremove</code>	Limpieza de paquetes inútiles.
<code>sudo rpi-update</code>	Actualiza el kernel o núcleo de la Raspberry.

### SERVIDOR LAMP

#### Actualizaciones semestrales del sistema operativo

Un servidor necesita mantenimiento, así como también revisiones constantes, es recomendable realizar actualizaciones semestrales. Esto se realiza en previsión de cualquier problema que puede surgir en un futuro. Las configuraciones que se realicen se deben ejecutar en la terminal del servidor.

## **Copias de seguridad**

Como prevención, se sugiere contar con registros de copia de seguridad. Es importante hacer copias de seguridad de los archivos del sistema, esto se puede realizar con la siguiente configuración ejecutada en la terminal:

```
sudo hg clone <nombre del archivo>
```

## **Asegurarse si se actualizó el sistema**

Las actualizaciones deben ser realizadas para responder adecuadamente a un problema específico, en la mayoría de casos suelen ser errores o virus. Para realizar este proceso se presenta la configuración a seguir:

```
yum update <nombre del paquete a actualizar>
```

## **Comprobar el consumo de espacio del CPU**

Es importante estar informado de los límites de recursos del servidor antes de que este empiece a hacer que el sistema se cuelgue o se bloquee en medio de las operaciones realizadas. Se recomienda utilizar programas en línea que permiten verificar el consumo, almacenamiento y datos de la red o memoria RAM.

## **Cambio de contraseñas**

Cambiar las contraseñas regularmente es altamente recomendable, especialmente si muchas personas tienen acceso al servidor.