

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO.**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES**

**INDUSTRIALES**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE SEGURIDAD PARA UN AUTOMÓVIL CON AUTENTICACIÓN POR RECONOCIMIENTO FACIAL UTILIZANDO TÉCNICAS DE VISIÓN ARTIFICIAL**

Trabajo de Titulación presentado para optar al grado académico de:

**INGENIERO EN ELECTRÓNICA, CONTROL Y REDES INDUSTRIALES**

**AUTOR:** CHRISTIAN FERNANDO SALAZAR ESPINOZA

**TUTOR:** ING. MSC. FERNANDO MEJIA

Riobamba – Ecuador.

2016

**FACULTAD DE INFORMÀTICA Y ELECTRÒNICA   
ESCUELA DE INGENIERIA EN ELECTRÒNICA EN CONTROL Y REDES INDUSTRIALES**

El Tribunal del Trabajo de Titulación certifica que: El trabajo de Titulación: “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE SEGURIDAD PARA UN AUTOMÓVIL CON AUTENTICACIÓN POR RECONOCIMIENTO FACIAL UTILIZANDO TÉCNICAS DE VISIÓN ARTIFICIAL”, de responsabilidad del Señor Christian Fernando Salazar Espinoza, ha sido minuciosamente revisado por los Miembros del Tribunal de Trabajo de Titulación, quedando autorizada su presentación.

Dr. Miguel Tasambay. P.H.D.

**DECANO DE LA FACULTAD**

**DE INFORMÁTICA Y ELECTRÓNICA. ………………… …………………..**

Ing. Msc. Wilson Zuñiga.

**DIRECTOR DE LA ESCUELA DE**

**INGENIERÍA ELECTRÓNICA EN**

**CONTROL Y REDES INDUSTRIALES ………………… …………………..**

Ing. Msc. Fernando Mejía

**DIRECTOR DE TRABAJO**

**DE TITULACIÓN ………………… …………………..**

Ing. José Luis Morales.

**MIEMBRO DE TRIBUNAL ………………… …………………..**

Yo, **CHRISTIAN FERNANDO SALAZAR ESPINOZA,** soy responsable de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación, y el patrimonio intelectual pertenece a la

**ESCUELASUPERIOR POLITÈCNICA DE CHIMBORAZO.**

CHRISTIAN FERNANDO SALAZAR ESPINOZA

El presente trabajo está dedicado a Dios, mi madre, mi abuelita, mis familiares y todas las personas que de varias maneras han influenciado de forma positiva en mi vida.

**ÍNDICE DE ABREVIATURAS**

**A** Amperios.

**A/D** Analogic - Digital

**ADC** Analogic – Digital Converter

**ALU** Arithmetic Logic Unit

**ARM** Advanced RISC Machine

**ARMv7** Advanced RISC Machine version 7

**AT** ATtention

**AUV**  Unmanned Aircraft Vehicle

**cm** Centímetro

**CMSIS** Cortex Microcontroller Software Interface Standard

**COM** Puerto de Comunicación Serie

**CPU** Central Unit of Process

**CSI** Camera Serial Interface

**dBm** Decibelio-Milivatio

**DMA** Direct to Memory Access

**DSI** Display Serial Interface

**DSP** Digital Signal Processor

**E/S** Entrada - Salida

**EGSM** Extension of GSM

**EN** Enable

**FLD** Fisher Linear Discriminant

**GB** Giga Byte

**GND** Ground

**GNU** GNU's Not Unix

**GPIO** General Port In / Out

**GPRS** General Packet Radio Service

**GPS** Geo

**GSM** Global System for Mobile communications

**HAL** [Hardware abstraction](https://en.wikipedia.org/wiki/Hardware_abstraction) Layer

**HDMI** High Definition Multimedia Interface

**HP** High Power

**Hz** Hertz

**I2C** Inter-Integrated Circuit

**IDE** Integrated Development Environment

**IEEE** Institute of Electrical and Electronics Engineers

**IP** Internet Protocol

**ISM** Industrial, Scientific and Medical

**ISP** In-System Programming

**KB** Kilo Bytes

**LBP** [Local binary patterns](https://en.wikipedia.org/wiki/Local_binary_patterns)

**LBPH**  [Local binary patterns](https://en.wikipedia.org/wiki/Local_binary_patterns) Histogram

**LDR** Light Dependent Resistor

**LQFP** Low-profile Quad Flat Package

**LUT** Look Up Table

**mA** Mili Amperios

**MCU** Multipoint Control Unit

**MHz** Mega Hertz

**MP** Mega Pixel

**NMEA** National Marine Electronics Association

**NMOS** Negative-channel Metal-Oxide Semiconductor

**NVIC** Nested Vectored Interrupt Controller

**ºC** Grado Celsius

**PC** Personal Computer

**PCA** Análisis de Componentes Principales

**PCB** Printed Circuit Board

**PWM** Pulse width Modulation

**RAM** Random Access Memory

**RGB** Red, Green and Blue

**RISC** Reduced Instruction Set Computer

**RX** Receiver

**SATA** Serial Advanced Technology Attachment

**SCCB** Serial Camera Control Bus

**SD** Secure Digital

**SMD** Surface Mounted Device

**SMS** Short Message Service

**SMT** Surface Mounting Technology

**SPI** Serial Protocol Interface

**SRAM** Static Random Access Memory

**TCP** Protocolo de Control de Transmisión

**TH** Thousand of an inches

**TTFF** Time To First Fix

**TTL** Logical Transistor to Transistor

**TX** Transmitter

**UART** Universal Asynchronous Receiver-Transmitter

**USART** Universal Synchronous Asynchronous Receiver-Transmitter

**USB** Universal Serial Bus

**V** Voltios

**VCC** Voltaje de Corriente Continua

**VGA** Video Graphics Array

**VLAN** Virtual LAN

TABLA DE CONTENIDO

[RESUMEN 14](#_Toc447117723)

[ABSTRACT 15](#_Toc447117724)

[INTRODUCCIÓN 16](#_Toc447117725)

[Capítulo 1 17](#_Toc447117726)

[1. MARCO REFERENCIAL. 17](#_Toc447117727)

[1.1. Antecedentes. 17](#_Toc447117728)

[1.2. Justificación del Trabajo de Titulación. 17](#_Toc447117729)

[1.3. Objetivos. 18](#_Toc447117730)

[1.3.1. Objetivo general. 18](#_Toc447117731)

[1.3.2. Objetivos específicos. 18](#_Toc447117732)

[1.4. Hipótesis. 18](#_Toc447117733)

[Capítulo 2 19](#_Toc447117734)

[2. MARCO TEÓRICO. 19](#_Toc447117735)

[2.1. Sistemas de seguridad de automóviles 19](#_Toc447117736)

[2.2. Sistemas inteligentes 19](#_Toc447117737)

[2.2.1. Control e interacción remota 19](#_Toc447117738)

[2.3. Seguridad y comodidad 19](#_Toc447117739)

[2.4. Comunicación inalámbrica 20](#_Toc447117740)

[2.4.1. Tecnología GSM 20](#_Toc447117741)

[2.4.2. Tecnología Bluetooth 20](#_Toc447117742)

[2.5. Sistemas embebidos 20](#_Toc447117743)

[2.5.1. Raspberry PI 2 20](#_Toc447117744)

[2.5.3. Wandboard 21](#_Toc447117745)

[2.6. Visión artificial 21](#_Toc447117746)

[2.6.1. Definición de visión artificial 21](#_Toc447117747)

[2.7 OpenCV 22](#_Toc447117748)

[2.7.1 Introducción a OpenCV 22](#_Toc447117749)

[2.7.2. Instalación de OpenCV en Raspbian 22](#_Toc447117750)

[2.7.3. Manipulación de imágenes 22](#_Toc447117751)

[2.7.4. Acceso a la cámara y obtención de imágenes 24](#_Toc447117752)

[2.7.4.1. Cámaras Web USB 24](#_Toc447117753)

[2.7.4.2. Cámaras IP 24](#_Toc447117754)

[2.7.4.3. Cámaras CSI 24](#_Toc447117755)

[2.7.5. Cámaras disponibles 24](#_Toc447117756)

[2.7.5.1. Cámaras Web USB normales 24](#_Toc447117757)

[2.7.5.2. Cámaras Web USB estéreo: 25](#_Toc447117758)

[2.7.5.3. Cámaras de un Smartphone como cámara USB 25](#_Toc447117759)

[2.7.5.4. Cámaras IP 25](#_Toc447117760)

[2.7.5.5. Cámaras de un Smartphone como cámara IP 25](#_Toc447117761)

[2.7.5.6. Cámaras CSI 25](#_Toc447117762)

[2.7.6. Detectar ojos y rostro 25](#_Toc447117763)

[2.7.7. Tipos de filtros de imágenes 27](#_Toc447117764)

[2.7.7.1. Reducción de la imagen 27](#_Toc447117765)

[2.7.7.2. Corte 27](#_Toc447117766)

[2.7.7.3. Canny 28](#_Toc447117767)

[2.7.7.4. Escala de grises 28](#_Toc447117768)

[2.8. Reconocimiento facial 28](#_Toc447117769)

[2.8.1. Algoritmo Eigenface 28](#_Toc447117770)

[2.8.2. Algoritmo Fisherface 29](#_Toc447117771)

[2.8.3. Algoritmo LBPH 31](#_Toc447117772)

[2.9. Microcontroladores con arquitectura ARM 33](#_Toc447117773)

[2.9.1. Introducción a la arquitectura ARM 33](#_Toc447117774)

[2.9.2. Características de la arquitectura ARM 34](#_Toc447117775)

[2.9.3. Microcontroladores STM32 34](#_Toc447117776)

[2.9.4. STM32CubeMX 35](#_Toc447117777)

[2.9.5. MDK V5 (KEIL) 35](#_Toc447117778)

[2.9.5.1. MDK Core 36](#_Toc447117779)

[2.9.5.2. Paquetes de Software 36](#_Toc447117780)

[2.9.6. STM Studio 36](#_Toc447117781)

[2.9.7. Programación de comunicación universal serial asincrónica de un microcontrolador 36](#_Toc447117782)

[2.9.7.1. Comunicación UART 36](#_Toc447117783)

[2.9.7.2. Comunicación UART por sondeo 37](#_Toc447117784)

[2.9.7.3. Comunicación UART por interrupciones 37](#_Toc447117785)

[2.9.8. MIT APP Inventor y Bluetooth 37](#_Toc447117786)

[2.9.9. Comandos AT para para uso de red GSM y GPS 38](#_Toc447117787)

[2.10. Dispositivos y componentes del sistema. 38](#_Toc447117788)

[2.10.1. Tarjeta Discovery STM32F3. 38](#_Toc447117789)

[2.10.2. Módulo USART MAX232. 40](#_Toc447117790)

[2.10.3. Módulo Bluetooth HC-05. 40](#_Toc447117791)

[Características 40](#_Toc447117792)

[2.10.4. Módulo GSM/GPRS/GPS SIM908. 41](#_Toc447117793)

[2.10.5. Raspberry PI 2. 42](#_Toc447117794)

[2.10.6. Cámara Webcam 43](#_Toc447117795)

[2.10.7. Atmega 328. 43](#_Toc447117796)

[2.10.8. Convertidor STEP UP STEP DOWN XI 6009 44](#_Toc447117797)

[2.10.9. Sensor de temperatura LM35. 44](#_Toc447117798)

[2.10.10. Sensor de distancia SHARP GP2Y0D340K. 45](#_Toc447117799)

[2.10.11. Sensores switch y/o pulsadores. 45](#_Toc447117800)

[2.10.12. Salidas a módulos relés mecánicos de potencia 46](#_Toc447117801)

[2.10.13. Sabertooth 2x5. 46](#_Toc447117802)

[2.10.14. Sensor de alcohol MQ-3 47](#_Toc447117803)

[2.10.15. Sensor de luz LDR 47](#_Toc447117804)

[2.10.16. Joystick 48](#_Toc447117805)

[2.10.17. Potenciómetros 48](#_Toc447117806)

[2.10.18. LED SMD 5050 49](#_Toc447117807)

[2.10.19. Resistencias SMD 49](#_Toc447117808)

[2.10.20. Zumbador 49](#_Toc447117809)

[2.10.21. Micro motor POLOLU 50](#_Toc447117810)

[2.10.22. Motor POLOLU A 12V 50](#_Toc447117811)

[2.10.23. Tarjeta Discovery STM32F4 51](#_Toc447117812)

[2.10.24. Wandboard Dual FREESCALE i.MX6 CORTEX A9 51](#_Toc447117813)

[Capítulo 3 53](#_Toc447117814)

[3. MARCO APLICATIVO 53](#_Toc447117815)

[3.1. Diseño del sistema 53](#_Toc447117816)

[3.1.1. Estructura Base 53](#_Toc447117817)

[3.1.2. Etapa de control 53](#_Toc447117818)

[3.1.3. Etapa de adquisición de datos 54](#_Toc447117819)

[3.1.4. Etapa de potencia 55](#_Toc447117820)

[3.2. Selección del Sistema Embebido 55](#_Toc447117821)

[3.3. Comunicaciones del sistema 57](#_Toc447117822)

[3.3.1. Comunicación STM32F3-RASPBERRY 57](#_Toc447117823)

[3.3.2. Comunicación STM32F3-Módulo Bluetooth 57](#_Toc447117824)

[3.3.3. Comunicación ATMEGA328p-Modulo GSM 57](#_Toc447117825)

[3.4. Etapa de potencia y alimentación del sistema 57](#_Toc447117826)

[3.5. Interfaz de control con Android 58](#_Toc447117827)

[3.6. Reconocimiento facial 59](#_Toc447117828)

[3.6.1. Selección de cámara 59](#_Toc447117829)

[3.6.2. Autenticación en la noche 60](#_Toc447117830)

[3.6.3. Filtrado de la imagen 61](#_Toc447117831)

[3.6.3.1. Reducción de imagen 61](#_Toc447117832)

[3.6.3.2. Corte 62](#_Toc447117833)

[3.6.3.3. Escala de Grises 62](#_Toc447117834)

[3.6.4. Selección de algoritmo para reconocimiento facial 62](#_Toc447117835)

[3.6.5. Breve descripción del algoritmo para reconocimiento facial 63](#_Toc447117836)

[3.7. Método de programación del microcontrolador ARM 63](#_Toc447117837)

[3.7.1. Programación por sondeo 63](#_Toc447117838)

[3.7.2. Programación por interrupciones 63](#_Toc447117839)

[3.7.3. Programación por DMA 64](#_Toc447117840)

[3.8. Control del sistema 64](#_Toc447117841)

[3.8.1. Modo seguro 65](#_Toc447117842)

[3.8.2. Modo conducción 65](#_Toc447117843)

[3.9. Construcción del sistema 65](#_Toc447117844)

[3.9.1. Estructura base 66](#_Toc447117845)

[3.9.2. Etapa de control 66](#_Toc447117846)

[3.9.3. Etapa de adquisición de datos 68](#_Toc447117847)

[3.9.4. Etapa de potencia 69](#_Toc447117848)

[3.10. OpenCV sobre Raspberry PI 2 con Raspbian 70](#_Toc447117849)

[3.11. Funcionamiento y algoritmos del sistema 70](#_Toc447117850)

[Capítulo 4 76](#_Toc447117851)

[4. COSTOS, PRUEBAS Y RESULTADOS 76](#_Toc447117852)

[4.1. Costo de los materiales 76](#_Toc447117853)

[4.2. Pruebas realizadas 77](#_Toc447117854)

[4.3. Análisis estadístico 77](#_Toc447117855)

[4.3.1. Tipos de cámara 77](#_Toc447117856)

[4.3.2. Métodos de reconocimiento facial 81](#_Toc447117857)

[4.4. Resultados 84](#_Toc447117858)

[CONCLUSIONES. 85](#_Toc447117859)

[RECOMENDACIONES. 87](#_Toc447117860)

[BIBLIOGRAFÍA. 89](#_Toc447117861)

**ÍNDICE DE FIGURAS**

[**Figura 2‑1** Función GasussianBlur](#_Toc434272710) 8

[**Figura 2‑2** Función Erode](#_Toc434272711) 8

[**Figura 2‑3** Función Dilate](#_Toc434272712) 8

[**Figura 2‑4** Filtro Canny 1](#_Toc434272713)1

[**Figura 2‑5** Coordenadas para buscar ojos 1](#_Toc434272714)1

[**Figura 2‑6** Parámetro para reducción de imagen 1](#_Toc434272715)1

[**Figura 2‑7** Detección de ojos 1](#_Toc434272716)2

[**Figura 2‑8** Código para reducción de imágenes 1](#_Toc434272717)2

[**Figura 2‑9** Código para filtro de Canny 1](#_Toc434272718)3

[**Figura 2‑10** Código para filtro a escala de grises 1](#_Toc434272719)3

[**Figura 2‑11** Eigenfaces de rostros 1](#_Toc434272720)4

[**Figura 2‑12** Fisherfaces de rostros 1](#_Toc434272721)5

[**Figura 2‑13** Operadores LBP 1](#_Toc434272722)6

[**Figura 2‑14** Distribución de la muestra de Chi cuadrado distancia](#_Toc434272723) 18

[**Figura 2‑15** STM32F3DISCOVERY 2](file:///C:\Users\PC\Downloads\TESIS_CRISTIAN.odt#_Toc434272724)4

[**Figura 2‑16** USB 2.0 Serial RS232 UART TTL CP2102. 2](#_Toc434272725)5

[**Figura 2‑17** Módulo Bluetooth HC-05.](file:///C:\Users\PC\Downloads\TESIS_CRISTIAN.odt#_Toc434272726) 26

[**Figura 2‑18** Módulo SIM908 GSM GPRS GPS 27](file:///C:\Users\PC\Downloads\TESIS_CRISTIAN.odt#_Toc434272727)

[**Figura 2‑19** Raspberry PI 2 modelo B](#_Toc434272728) 27

[**Figura 2‑20** Cámara Webcam](#_Toc434272729) 28

[**Figura 2‑21** IC Atmega328P](#_Toc434272730) 28

[**Figura 2‑22** Convertidor step up step down XI 6009](#_Toc434272731) 29

[**Figura 2‑23** Sensor de temperatura LM35](#_Toc434272732) 29

[**Figura 2‑24** Sensor Sharp GP2Y0D340K 0cm a 40cm 3](#_Toc434272733)0

[**Figura 2‑25** Sensor switch 3](#_Toc434272734)0

[**Figura 2‑26** Módulo de relés a 5V 3](#_Toc434272735)1

[**Figura 2‑27** Sabertooth 2x5 3](#_Toc434272736)1

[**Figura 2‑28** Sensor de alcohol MQ-3. 3](#_Toc434272737)2

[**Figura 2‑29** Sensor de luz LDR 3](#_Toc434272738)2

[**Figura 2‑30** Joystick 3](#_Toc434272739)2

[**Figura 2‑31** Potenciómetros 3](#_Toc434272740)3

[**Figura 2‑32** Led SMD 5050 3](#_Toc434272741)3

[**Figura 2‑33** Resistencias SMD 3](#_Toc434272742)4

[**Figura 2‑34** Zumbador 3](#_Toc434272743)4

[**Figura 2‑35** Micro motor Pololu 3](#_Toc434272744)4

[**Figura 2‑36** Motor Pololu a 12V 35](#_Toc434272745)

[**Figura 2‑37** Tarjeta DISCOVERY STM32F4](#_Toc434272746) 35

[**Figura 2‑38** Wandboard dual Freescale i.MX6 Cortex A9.](#_Toc434272747) 36

[**Figura 3‑1**  Diagrama de bloques del abstracto del sistema completo](#_Toc434272748) 36

[**Figura 3‑2** Diagrama de bloques de etapa de control](#_Toc434272749) 38

[**Figura 3‑3**  Diagrama de bloques de etapa de adquisición de datos](#_Toc434272750) 38

[**Figura 3‑4** Diagrama de bloques de etapa de potencia](#_Toc434272751) 39

[**Figura 3‑5** Comunicación Sistema Embebido - Micro controlador](#_Toc434272752) 41

[**Figura 3‑6** Interfaz de control con Android 4](#_Toc434272753)2

[**Figura 3‑7**  Funciones de la aplicación de Android 4](#_Toc434272754)3

[**Figura 3‑8**  Cámara de visión nocturna](#_Toc434272755) 44

[**Figura 3‑9** Led infrarrojo 45](#_Toc434272756)

[**Figura 3‑10** Flash de led 45](#_Toc434272757)

[**Figura 3‑11** Funciones en C++ para filtro de imágenes](#_Toc434272758) 46

[**Figura 3‑12** Reconocimiento facial 4](#_Toc434272759)7

[**Figura 3‑13** Diagrama de bloques secuencial de procesos](#_Toc434272760) 49

[**Figura 3‑14** Diagrama de bloques detallado del sistema](#_Toc434272761) 49

[**Figura 3‑15** Diagrama electrónico de la etapa de control 5](#_Toc434272762)0

[**Figura 3‑16** PCB del diagrama electrónico de la etapa de control 5](#_Toc434272763)1

[**Figura 3‑17** Imagen 3D del PCB del diagrama de electrónico de la etapa de control 5](#_Toc434272764)2

[**Figura 3‑18** Diagrama electrónico de la etapa de adquisición de datos](#_Toc434272765) 52

[**Figura 3‑19** PCB del diagrama electrónico de la etapa de adquisición de datos](#_Toc434272766) 53

[**Figura 3‑20** Algoritmo de la tarjeta STM32F303](#_Toc434272767) 55

[**Figura 3‑21** Algoritmo del programa de reconocimiento facial 56](#_Toc434272768)

[**Figura 3‑22** Algoritmo del Atmega328 con SIM908](#_Toc434272769) 57

[**Figura 3‑23** Algoritmo de la aplicación de Android](#_Toc434272770) 58

[**Figura 4‑1**  Gráfico porcentual de las Sumas de la tabla de Baremo de tipo de cámaras](#_Toc434272771) 63

[**Figura 4‑2** Gráfico porcentual de las Sumas de la tabla de Baremo de tipo de métodos de reconocimiento facial](#_Toc434272772) 67

**ÍNDICE DE TABLAS**

[**Tabla 2‑1**Lista de comandos AT usados para comunicación con SIM908 2](#_Toc434272355)3

[**Tabla 3‑1**Tabla comparativa entre Raspberry Pi 2 y Wandboard iMX6 Dual 39](#_Toc434272356)

[**Tabla 3‑2** Tabla comparativa entre los tres tipos de cámara](#_Toc434272357) 43

[**Tabla 4‑1**Lista de precios](#_Toc434272358) 60

[**Tabla 4‑2** Datos recolectados de los tres tipos de cámara funcionando en el Sistema Embebido](#_Toc434272359) 61

[**Tabla 4‑3**Tabla de Baremo de los datos de la Tabla 2-4](#_Toc434272360) 61

[**Tabla 4‑4**Resultados del análisis estadístico de Tabla 3-4 en software SPSS](#_Toc434272361) 62

[**Tabla 4‑5**Datos recolectados de los tres métodos de reconocimiento facial funcionando en el Sistema Embebido](#_Toc434272362) 64

[**Tabla 4‑6**Tabla de Baremo de los datos de la Tabla 5-4](#_Toc434272363) 65

[**Tabla 4‑7**Resultados del análisis estadístico de Tabla 6-4 en software SPSS](#_Toc434272364) 65

## RESUMEN

Con el objetivo de determinar una adecuada cámara, un adecuado algoritmo de reconocimiento facial y adecuado sistema embebido, en la ciudad de Riobamba se diseñó y ensamblo un sistema de seguridad para automóviles con autenticación por reconocimiento facial con un micro controlador con arquitectura ARM de 32 bits, software libre OPENCV que es usado para analizar imágenes y en si para dotar al sistema de visión artificial sobre un pequeño computador Raspberry Pi 2 que cuenta con el sistema operativo Raspbian, el sistema cuenta con diferentes partes o etapas, teniendo así un módulo para geo localización constituido por un Atmega 328p junto con un Módulo GSM/GPRS/GPS SIM908 para manejar redes inalámbricas como GSM y GPS, el sistema tiene la capacidad de mantener sin arranque al vehículo si no se supera la autenticación, puede avisarnos de anomalías con una llamada perdida, es capaz de darnos su localización aproximada por mensajes de texto cortos, cuenta con una aplicación Android para interactuar con el usuario la cual cuenta con la opción para abrir el seguro de puertas del automóvil por medio de ordenes enviadas a través de Bluetooth, además tiene clave de súper usuario para casos de emergencia, la cámara de tipo webcam USB obtuvo resultados de un 74% en contraste al tipo de mayor requerimiento, no así las cámaras de Protocolo de Internet (IP) y la con Conector de Interfaz Serial(CSI)con mayores porcentajes, así mismo con el método [Local Binary Patterns](https://en.wikipedia.org/wiki/Local_binary_patterns) Histogram (LBPH) obtuvo menores requerimientos con un 88.57% . Concluyendo así que la mejor cámara y mejor método de reconocimiento facial resultaron ser la webcam USB y LBPH respectivamente, también analizando que el sistema embebido más adecuado fue la antes mencionada Raspberry Pi 2, por estos resultados obtenidos se recomienda usar las usadas en este caso para con ello no caer en el sobredimensionamiento innecesario en la aplicación.

<RIOBAMBA [CANTÓN]><SEGURIDAD VEHICULAR><VISIÓN ARTIFICIAL [RAMA DE LA INTELIGENCIA ARTIFICIAL]><METODO DE RECONOCIMIENTO FACIAL [LBPH]><SOFTWARE DE PROGRAMACION DE MICROCONTROLADORES STM32 [STM32CUBE MX]><TARJETA DE DESARROLLO [STM32F3 DISCOVERY]><MÓDULO DE COMUNICACIÓN GSM/GPRS/GSP [SIM908]><MINI COMPUTADOR [RASPBERRY PI 2]><SISTEMA OPERATIVO [RASPBIAN]><ARQUITECTURA DE COMPUTADORES [ARM]>.

## 

## ABSTRACT

In Riobamba city to determine an appropriate camera, a facial recognition algorithm, and a proper embedded system, the researcher designed and assembled a car security system with facial recognition authentication a microcontroller with ARM 32-bit architecture free software OPENCV that is used to analyze images and whether to provide the system of artificial vision on a small computer Raspberry Pi 2 has the Raspbian operating system. The system has different parts or stages; it has a module for geo location consisting of an Atmega 328p with a module GSM / GPRS / GPS SIM 908 to handle wireless networks such as GSM and GPS. The system has the ability to maintain without starting the vehicle if authentication does not overcome, it can warn us of anomalies with a missed call; it is able to give us your approximate location for short text messages. The system has an Android application to interact with the user which has the option to open the door lock of the automobile through commands sent via Bluetooth, also it has super user key for emergency cases. The camera USB webcam type got results from 74% in contrast to the type of higher demand, it isn`t the same with the Internet Protocol cameras (IP) and with Serial Interface Connector (CSI) with higher percentages; likewise, the Local Binary Patterns Histogram (LBPH) method, it obtained minor requirements with a 85.57%.

The conclusion is that the best camera and the best method of face recognition proved to the USB webcam and LBPH respectively, also considering that the most appropriate embedded system was the aforementioned Raspberry Pi 2. These results recommend applying those used in this case, in order they don`t fall into unnecessary oversizing in the application.

<RIOBAMBA [CANTON] >< VEHICLE SAFETY >< ARTIFICIAL VISION [BRANCH OF THE ARTIFICIAL INTELIGENCE] >< FACIAL RECOGNITION METHOD [LBPH] >< STM32 MICROCONTROLLER PROGRAMMING SOFTWARE [STM32CUBEMX]><DEVELOPMENT BOARD [STM32F3 DISCOVERY]><COMUNICATION MODULE GSM/GPRS/GSP [SIM908]>< MINI COMPUTER [RASPBERRY PI 2]><OPERATING SYSTEM [RASPBIAN]><COMPUTER ARCHITECTURE [ARM]>.

## INTRODUCCIÓN

Debido a los altos índices delincuenciales en contra de la integridad y seguridad de los automóviles se pretende combatir este problema social diseñando un sistema de seguridad que contara con: un proceso de reconocimiento facial, redes de comunicación para alertar e interactuar de forma remota y algunas características generales de los sistemas de seguridad para vehículos existentes en el mercado.

La visión artificial es una rama derivada de la Inteligencia Artificial que estudia el análisis de las imágenes por procesos computacionales para la toma de decisiones y obtención de resultados en diferentes campos de la industria y la investigación.

A pesar de ser una ciencia muy joven tiene gran acogida a nivel mundial con una gran tendencia a seguir aumentando su estudio y desarrollo, lo cual van abriendo muchas más opciones con sus respectivas aplicaciones que pueden ayudar al ser humano a desarrollar actividades de cierta complejidad.

Hoy en día la visión artificial se ve involucrada en muchas ramas de la ciencia como en la medicina, geología, arquitectura, agricultura, industria, etc.

Con el avance de la tecnología y la creación de procesadores más pequeños, potentes y ahora muy portátiles hacen que la aplicación de algoritmos de visión artificial sea cada vez más frecuente ya en procesos que ayudan a resolver problemas de mejor de manera.

En el presente trabajo se ha seleccionado el más óptimo y aplicable de los métodos y técnicas de visión artificial para reconocimiento facial con el fin de hacer autenticación de un sistema de seguridad para un automóvil a más de determinar el grado de aplicabilidad en los diferentes casos para que el sistema sea una ayuda, para lo cual se hizo una selección de una adecuada cámara, fiables protocolos de comunicación entre los diferentes bloques del sistema, opción para adquisición de imágenes en la noche.

Se optó por usar una metodología científico inductivo por la necesidad de realizar pruebas para obtener resultados y llegar a las respectivas conclusiones de los objetivos planteados.

# 

## 1. MARCO REFERENCIAL.

### 1.1. Antecedentes.

Se ha observado que últimamente ha surgido un gran problema de carácter social, tal es que se ha estado vulnerando la seguridad de los automóviles cada vez con más frecuencia para usarlos con diferentes fines, donde el dueño del vehículo puede salir muy afectado por diferentes causas, existen varios sistemas de seguridad para los vehículos, unos más sofisticados que otros, pero lastimosamente siempre se las ingenian para vulnerarlos, llevando con esto a una competencia entre quienes diseñan sistemas cada vez más seguros y quienes lo burlan por lo cual se ha pensado con la presente propuesta, poner una gran dificultad al momento de llevar a cabo su cometido.

En si se puede deducir a breves rasgos la acogida del proyecto tomando en cuenta las estadísticas del incremento de este tipo de delitos según el Observatorio Metropolitano de Seguridad en Quito, se registró un incremento de estos desde el 2012, también indica en el informe de la seguridad ciudadana número 16° que el 92% de robos de autos ocurre en la vía pública y el 8% restante en estacionamientos, locales comerciales y otros.

### 1.2. Justificación del Trabajo de Titulación.

El reconocimiento facial tiene un alto grado de aplicabilidad donde se requiere reconocer una o varias personas, clasificarlas de acuerdo a un patrón de rasgos físicos de la fisionomía de los rostros de las personas por lo cual el algoritmo usado en este proyecto puede ser involucrado en muchos casos donde se necesite hacer una autenticación de seguridad.

Existen tres técnicas o algoritmos de visión artificial para reconocimiento facial más usadas tales como EigenFace, FisherFace y LBPH, cada uno de ellos con sus características principales, ventajas y desventajas de uno sobre otro, Raspberry es una plataforma de sistemas embebidos de gran acogida y auge en los últimos tiempos y desde su creación ha llevado parte en el desarrollo de muchas nuevas tecnologías y aplicaciones, por el motivo de que es una plataforma de sistemas embebidos que se encuentra entre las más usadas y accesibles por cuestiones económicas y acceso a soporte y continuo desarrollo hacen que cada día incremente el número de usuarios en el mundo pensando nuevas y útiles aplicaciones, así mismo se debe mencionar que la arquitectura ARM es por no decir la más usada, si es una de las más requeridas a nivel mundial en el campo de la industria tecnológica y en la investigación por su estructura y demás características que han desplazado de a poco a las anteriores arquitecturas que fueron importantes para el desarrollo de ésta, ARM cuenta con presencia en la mayoría de los productos tecnológicos que adquirimos como smartphones, tablets, televisores, cámaras, gadgets, etc, lo cual es un visible indicador que debemos seguir y estar a la vanguardia del avance tecnológico en cuanto al uso y aplicabilidad de los microcontroladores de última generación ya que tiene grandes tendencias a globalizar su uso en su totalidad cabe destacar que la plataforma Raspberry usa microprocesadores con arquitectura ARM, OpenCV es un software de distribución de licencia libre con aplicabilidad en varios lenguajes de programación de ato nivel como C, Java, Python, etc, que fue diseñado para el análisis y tratamiento de imágenes, donde mediante algoritmos y técnicas de visión artificial es capaz de programar un algoritmo con sus herramientas disponibles para hacer una autenticación difícil de vulnerar teniendo así un respaldo ante posibles eventos.

### 1.3. Objetivos.

#### 1.3.1. Objetivo general.

* Diseñar e implementar un sistema de seguridad para un automóvil con autenticación por reconocimiento facial usando técnicas de visión artificial.

#### 1.3.2. Objetivos específicos.

* Analizar las cámaras que existen en el mercado para seleccionar la más idónea para una mejor obtención de las imágenes para el proyecto.
* Seleccionar la mejor técnica o método para el reconocimiento facial inteligente.
* Investigar las técnicas de filtro de imágenes para seleccionar el apropiado.
* Estudiar las posibles soluciones para la autenticación en la noche.
* Estudiar la mejor forma de comunicar el sistema embebido con el micro controlador.
* Seleccionar el más adecuado sistema embebido para el proyecto.
* Implementar un método de programar micro controladores de 32 bits para el control del sistema.
* Hacer el prototipo de control automático de algunos subsistemas del vehículo.

### 1.4. Hipótesis.

“La autenticación por reconocimiento facial ejecutada con recursos de licencia libre la hacen una de las opciones más seguras y fiables que están a la vanguardia del avance tecnológico en cuanto a sistemas de seguridad”.

# 

## 2. MARCO TEÓRICO.

### 2.1. Sistemas de seguridad de automóviles

Un sistema de seguridad de un automóvil puede estar enfocado a salvaguardar la integridad del usuario, del vehículo o ambos.

Son medidas que se ejecutan con el fin de que ocurra un suceso cuando este programado y de hacerlo antes tomar una decisión adecuada.

Los sistemas en la actualidad están muy avanzados en los servicios que ofrecen a los usuarios, pero a cambio de una gran inversión. Creando una brecha entre los usuarios que tienen acceso a esa tecnología que puede ser borrada en gran parte con el avance tecnológico y la licencia libre para ciertos casos haciendo accesible en la mayoría de los casos a todos los recursos necesarios.

### 2.2. Sistemas inteligentes

La implementación de sistemas inteligentes en vehículos es realmente nueva, con el lanzamiento en el año 2015 por la compañía Google, las características de estas técnicas son que se encargan de la correcta funcionalidad de todo el automóvil y el de la seguridad de los usuarios y del vehículo.

Son estructuras que tienen conexión y comunicación en tiempo real gran parte del tiempo lo que es posible la monitorización a cualquier momento.

#### 2.2.1. Control e interacción remota

El acceso o control remoto a ciertas funcionalidades que puede ofrecer un automóvil es muy importante ya que si usamos las herramientas necesarias para este fin de manera correcta será una gran ventaja brindando comodidad y ahorro de tiempo, caso contrario si no logramos dominarlo presentarnos problemas inesperados.

De acuerdo a la tendencia que hoy en día tiene la tecnología a mantenernos siempre conectados a la nube, en el automóvil es donde muchas personas pasan gran parte del tiempo generando la necesidad de brindar cobertura de conexión, lo que nos lleva indudablemente a estar conectado remotamente a la nube.

### 2.3. Seguridad y comodidad

La implementación de sistemas en los vehículos que puedan tomar decisiones ya sea por comodidad o por seguridad debería llevarnos a implementar ambos conceptos, ya que en ambos casos el proyecto podría ser capaz de cumplir funciones necesarias, implicando que si fue diseñado para brindar comodidad puede añadírsele funcionalidades que brinden también seguridad o viceversa.

### 2.4. Comunicación inalámbrica

#### 2.4.1. Tecnología GSM

Sus siglas provienen de Sistema Global para las Comunicaciones Móviles, según Martínez (2011, p.31) GSM es el sistema de telefonía móvil de segunda generación más conocido en el mundo y el emblemático de Europa.

Para la codificación para voz y datos GSM utiliza diferentes algoritmos.

GSM puede ser usada en la electrónica ya que hay muchos dispositivos que soportan las comunicaciones seriales y los comando AT lo que les permite trabajar sobre esta red.

#### 2.4.2. Tecnología Bluetooth

Bluetooth una de las opciones de conexión inalámbrica más usadas, que permite la transferencia de archivos y la comunicación entre dispositivos, a pesar de no tener gran alcance es muy adaptable en la mayoría de las aplicaciones por lo que incluso casi todos los teléfonos celulares de gama media en adelante cuentan con un periférico de este tipo.

Vignoni (2007, p.4) dice que Bluetooth se transformó en un estándar de facto, posteriormente considerado como estándar internacional IEEE 802.15.1, opera en la banda ISM, 2.4 a 2.485 GHz, utilizando spread septum, adaptativa frecuencia shopping, full- dúplex a una velocidad nominal de 1600 hops/sec.

### 2.5. Sistemas embebidos

Un sistema embebido (SE) o sistema empotrado lo vamos a definir como un sistema electrónico diseñado específicamente para realizar unas determinadas funciones, habitualmente formando parte de un sistema de mayor entidad. La característica principal es que emplea para ello uno o varios procesadores digitales (CPUs) en formato microprocesador, micro controlador o DSP lo que le permite aportar ‘inteligencia’ al sistema anfitrión al que ayuda a gobernar y del que forma parte. En el diseño de un sistema embebido se suelen implicar ingenieros y técnicos especializados tanto en el diseño electrónico hardware como el diseño del software. A su vez también se requerirá la colaboración de los especialistas en el segmento de usuarios de tales dispositivos, si hubiese lugar a ello (Úbeda, 2009, p.2).

#### 2.5.1. Raspberry PI 2

El Raspberry Pi es un ordenador, muy parecido a los ordenadores con los que estamos familiarizados. Utiliza un tipo diferente de procesador, por lo que no se puede instalar Microsoft Windows en la misma. Pero se puede instalar varias versiones del sistema operativo Linux que se parecen mucho a Windows. Si se lo desea, se puede utilizar el Raspberry Pi para navegar por Internet, enviar un correo electrónico o escribir una carta. Pero también se puede hacer mucho más.

Raspberry es fácil de usar, muy potente y asequible, esta mini computadora es la herramienta perfecta para los aspirantes a científicos de la computación. Esto se refiriere a aprender cómo funcionan los ordenadores para que pueda hacer que hagan lo que usted quiere que hagan, no lo que alguien que piensa debe hacer con ellos (Computing at School, 2012, p.5).

#### 2.5.3. Wandboard

El Wandboard es un tablero de Cortex-A9 de bajo costo creada por un grupo de ingenieros en su tiempo libre que trabajan con soluciones basadas en la arquitectura ARM en una base diaria. El espíritu que impulsó el proceso de implementación fue el hecho de que muchos de los consejos de desarrollo que ven hoy en día son divertidos para jugar, pero no hay flexibilidad en lugar de diseñar un producto más allá de la etapa inicial. Con la Wandboard tratan de superar esta por elegir un System-on-Module que funciona con un fácil diseñar algunas capas inferiores que todos puedan diseñar y hacer con que comienza conocimientos de ingeniería. El Wandboard se construyó como una tarjeta sin fines de lucro.

El Wandboard está disponible en tres configuraciones diferentes:

1. Wandboard solitario
   * Procesador i.MX6 solo núcleo
   * 512 MB de RAM
2. Wandboard Dual
   * Procesador de doble núcleo i.MX6
   * 1 GB de RAM
   * 802.11n inalámbrica y bluetooth
3. WandboardQuad
   * Quad Core i.MX6
   * 2 GB de RAM
   * 802.11n inalámbrica y bluetooth
   * SATA

### 2.6. Visión artificial

#### 2.6.1. Definición de visión artificial

Guillen (2011, p.1) dice que “El uso de Visión Artificial no se restringe al procesamiento de imágenes obtenidas solamente por sensores ópticos sino también las provenientes de sensores acústicos”. Para labores de detección de largo alcance resulta más conveniente el empleo de sensores acústicos, pero en entornos que requieren de gran precisión en corta distancia, se torna imprescindible el uso de sensores ópticos. En fin, otras aplicaciones de la visión artificial como en AUVs son evitar obstáculos, reconocimiento de objetos, mantenimiento e inspección de estructuras, o seguimiento de objetos. En sistemas submarinos tales como el océano, las condiciones dinámicas de la luz, y las turbulencias pueden afectar enormemente a las imágenes capturadas (Guillen, 2011, pp.1-2).

### 2.7 OpenCV

#### 2.7.1 Introducción a OpenCV

OpenCV es una biblioteca de licencia libre y que lo tenemos disponible en muchos sitios de la red que sirve principalmente para el procesamiento de imágenes y que fue creado por la empresa Intel, existe OpenCV para usarlos en Linux, Windows o Macintosh en varias versiones de actualización, aunque se aconseja usar la más reciente, los lenguajes de programación que puede soportar OpencCV son C, C++, Java, Python y Android.

OpenCV, aunque es una librería muy joven, hoy en día ha jugado un papel muy importante en cuanto a avance tecnológico y en el crecimiento de la visión por computador, con ello permitiendo a miles de personas ejecutar un trabajo más productivo en el campo de la visión artificial.

#### 2.7.2. Instalación de OpenCV en Raspbian

Actualmente se tiene disponible una versión de OpenCv un tanto genérica para instalarlo sobre Raspbian lo cual implica que no se va a poder tener muchas opciones que normalmente ofrecería una versión completa por lo que Raspbian al estar basado sobre debían y estar con el soporte de este, es posible instalar una versión completa disponible en la nube.

Los pasos que se deberían hacer son:

-Actualizar las cabeceras del sistema operativo.

-Instalar las dependencias necesarias que usara OpenCV.

-Descargar y descomprimir OpenCV.

-Crear una carpeta built dentro de la descompresión y ejecutar cmake y make.

Para estos pasos más detallados en internet hay muchos tutoriales sobre la instalación.

#### 2.7.3. Manipulación de imágenes

OpenCV cuenta con muchas funciones predefinidas las cuales podemos hacer referencia, estas funciones pueden ocuparse para manipular y editar imágenes, ya sea como reducir, filtrar, incluso algoritmos complicados como:

GaussianBlur(imagen, nueva,Size(13,7), 8);

Dónde:

imagen: Es la matriz donde esta guardada la imagen original

nueva: Es la nueva matriz donde se guarda el resultado de pasar la matriz anterior por el algoritmo de la función.

Size(width, height): Es el tamaño del Kernel, hay que tomar en cuenta que son números positivos e impares

8: Es una desviación de tipo estándar sobre el eje de las x.

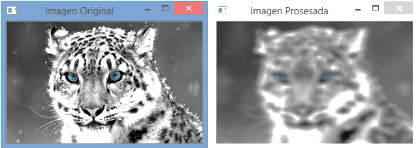
**

Figura 2‑1 Función GasussianBlur

**Fuente:** http://4.bp.blogspot.com/-zFk5fzn7qZc/UZ95-Ueo1jI/AAAAAAAAAP8/9b4O4XC2JFU/s1600/gaussianblur.png

erode(imagen, nueva, krl);

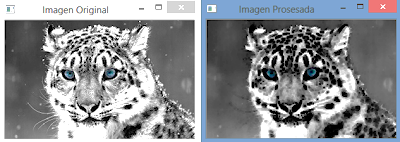
****

Figura 2‑2 Función Erode

**Fuente**: <http://1.bp.blogspot.com/-K0tBysflT4M/UZ959mPxlxI/AAAAAAAAAPw/1WtMcA5IOdM/s400/erode.png>

dilate(imagen, nueva, krl);

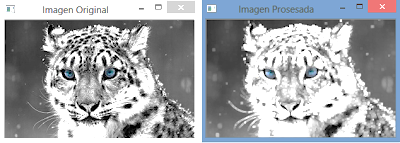
****

Figura 2‑3Función Dilate

**Fuente**: <http://3.bp.blogspot.com/-yJU7vxbyE7E/UZ95-Kr1FtI/AAAAAAAAAP4/OnZQrfhnVsM/s1600/dilate.png>

En muchas ocasiones es necesario guardar las imágenes que se han procesado, para lo cual podemos usar las líneas que aclaran el tipo de compresión que se usara para guardar la imagen, formato, matriz que la representa, nombre de la imagen a guardar, etc

vector<int>compression\_params;

compression\_params.push\_back(CV\_IMWRITE\_PNG\_COMPRESSION);

compression\_params.push\_back(9);

imwrite("editada.png", nueva, compression\_params);

#### 2.7.4. Acceso a la cámara y obtención de imágenes

El acceso a la cámara es muy importante en el procesamiento de imágenes ya que muchas de las aplicaciones así lo requieren.

Habría tres tipos de cámaras normalmente disponibles de acuerdo a su conexión y forma de introducir la información dentro del programa y que de estas puede haber variantes.

##### 2.7.4.1. Cámaras Web USB

Este tipo de cámara se conectan físicamente al puerto o conector USB, transmitiendo la información de forma paralela, dependiendo de la resolución de la cámara dependerá la cantidad de información a transmitir y por defecto de la latencia de la comunicación, para lo cual se puede contar con puertos USB mucho más rápidos como los d ella versión 3.0

Existen cámaras estéreo que se conectan por USB la desventaja podrían ser que se divide el ancho del puerto reduciendo a la mitad la velocidad de transmisión por cada cámara.

##### 2.7.4.2. Cámaras IP

Las cámaras IP también son una alternativa posible para introducir la información a manipular y procesar dentro de OpenCV, su interfaz de conexión es de forma remota lo cual dependiendo desde el punto de vista que se lo vea puede ser una ventaja o debilidad.

Al estar conectado a una red que está destinada no solo al servicio de la cámara IP a menos que así sea, se verá afectada la transmisión e incrementada mucho más la latencia de llegada d ella información.

##### 2.7.4.3. Cámaras CSI

Las cámaras con conector CSI son muy rápidas a pesar que se transmite la información de forma serial a costa de procesamiento para atender al periférico encargado, existen cámaras de muy buena resolución desde 5 MP en adelante, este tipo de cámaras se pueden conectar en sistemas embebidos ya que ahí podemos encontrar este tipo de conexión, a menos que se conecte con una interfaz extra para adaptar a una PC de escritorio normal lo que nos llevaría muy probablemente a los casos anteriores.

#### 2.7.5. Cámaras disponibles

Entre las cámaras disponibles tenemos:

##### 2.7.5.1. Cámaras Web USB normales

Se trata de las tradicionales webcams de conexión USB.

##### 2.7.5.2. Cámaras Web USB estéreo:

Con el mismo caso anterior en cuanto a la conexión, pero tratándose de dos cámaras separadas mínimamente una de la otra. Muy usadas para percepciones de profundidades.

##### 2.7.5.3. Cámaras de un Smartphone como cámara USB

Se trata de una aplicación que puede hacer uso de la cámara del Smartphone como una webcam USB y transmitiendo la información por medio de la conexión física con el cable de datos del dispositivo.

##### 2.7.5.4. Cámaras IP

Son cámaras que se usan en una red, puede ser privada o pública, sus principales funciones son la de seguridad, normalmente son de muy alta resolución.

##### 2.7.5.5. Cámaras de un Smartphone como cámara IP

Al igual que en el caso del Smartphone como cámara USB aquí también se trata de una aplicación que permite dar esta funcionalidad al dispositivo, debiendo estar conectado a la red e iniciada las funciones de transmisión y configuraciones previas.

##### 2.7.5.6. Cámaras CSI

Son cámaras para uso específico y en algunos casos industriales.

#### 2.7.6. Detectar ojos y rostro

La librería de OpenCV en sus distribuciones actuales cuenta con diversos clasificadores que nos ayudan a detectar ojos, rostros y demás.

Un algoritmo para la detección rápida en una sub - ventana, donde un resultado positivo de un clasificador desencadena la próxima clasificadora en la fila. Un resultado negativo desde el clasificador en cualquier punto durante la cascada será rechazo inmediato de la sub - ventana actual (Rondahl, 2011, p.5).

Estos clasificadores se los puede encontrar en un directorio llamado data en la carpeta que contiene la librería descomprimida que su vez tiene un directorio llamado haarcascades, mismo nombre que es prefijo de algunos de los distintos clasificadores seguido del tipo de clasificador.

Para nuestro caso podemos hacer referencia y llamado al clasificador con extensión XML para usarlo como una función más en nuestro proyecto escrito en C con la siguiente línea de código que debe contener la dirección donde está localizada:

*faceDetector.load("/home/cristian/opencv-2.4.9/data/haarcascades/haarcascade\_frontalface\_alt\_tree.xml")*

*lEyeDetector.load("/home/cristian/opencv-2.4.9/data/haarcascades/haarcascade\_eye\_tree\_eyeglasses.xml")*

*rEyeDetector.load("/home/cristian/opencv-2.4.9/data/haarcascades/haarcascade\_eye\_tree\_eyeglasses.xml")*

Teniendo como resultado un uno si se logró el objetivo o un caso contrario un cero.

La primera línea como podemos observar hace referencia al clasificador de rostros

El funcionamiento de los clasificadores en cascada es básicamente en dividir la imagen en rectángulos normalmente de 24 por 24 pixeles los cuales se les aplica el algoritmo haarlike donde los pixeles son convertidos en valores y sumados en cada ventana para determinar si es o no un candidato.

Después de esto se procede a aplicar un pruding que es un tipo de filtro basado en canny para descartar áreas que no contengan patrones y características de una posible cara y detectar bordes para así disminuir en gran escala la carga computacional ya que esto da como resultado una imagen solo en blanco y negro.



Figura 2‑4Filtro Canny

**Fuente:** http://3.bp.blogspot.com/-ybDW9YBG05s/UsvD3HRj7JI/AAAAAAAABWI/5bUbVWpXx-U/s1600/canny.JPG

Las líneas restantes hacen referencia a los clasificadores de ojos, uno por cada ojo (izquierdo y derecho), caso similar es este algoritmo que de igual manera usa un filtro para detectar los patrones característicos de un ojo, como lo es su iris, forma, y ubicación especialmente y para ello debemos especificar al clasificador las coordenadas donde buscar los ojos y después de reducir y recortar la imagen con el clasificador anterior se ha determinado que los puntos son:

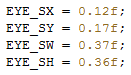


Figura 2‑5Coordenadas para buscar ojos

**Fuente:** Autor

Para una reducción de una imagen de



Figura 2‑6Parámetro para reducción de imagen

**Fuente:** Autor

Es decir 480 pixeles

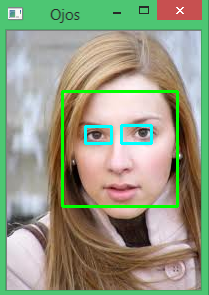


Figura 2‑7 Detección de ojos

**Fuente:** http://1.bp.blogspot.com/-Gb1tpH\_I5ZY/Uh0UoIGKyaI/AAAAAAAAAjo/5yBFviF4M3Y/s1600/detector+de+ojos.png

#### 2.7.7. Tipos de filtros de imágenes

Para el análisis de la imagen cualquiera que sea se puede proceder a filtrar la información ya que esta acción nos puede ayudar para su análisis o para reducir los costos computacionales.

##### 2.7.7.1. Reducción de la imagen

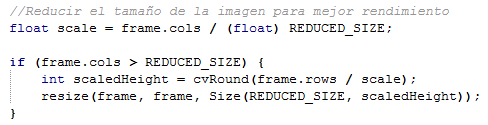


Figura 2‑8 Código para reducción de imágenes

**Fuente**: [autor](http://3.bp.blogspot.com/-yJU7vxbyE7E/UZ95-Kr1FtI/AAAAAAAAAP4/OnZQrfhnVsM/s1600/dilate.png)

La función es el resize o redimensionamiento la cual necesita la matriz a reducir, matriz de destino del proceso y el nuevo tamaño o Size con los valores que se pueden sacar con una breve relación con un tamaño predefinido o simplemente con una escala de la cual dependerá el resultado como del tamaño de la matriz de entrada.

##### 2.7.7.2. Corte

Es una función muy útil para subdividir la matriz general para poder aplicar algoritmos por separado para al final poder comparar y tener un resultado.

Se tiene recursos como la copia de regiones en sub matrices u otros

##### 2.7.7.3. Canny

Este filtro tiene la característica de reducir la información de la imagen y entregarla en binario muy diferente a la escala de grises que más adelante se va a detallar y que a más de ello es necesaria previa al proceso de canny, este resultado es la búsqueda de los bordes característicos del contenido de la imagen, en sus pixeles tiene información de en binario de tal forma que un uno lógico sería 255 y un cero lógico una saturación de 0 en el pixel.



Figura 2‑9Código para filtro de Canny

**Fuente**: [autor](http://3.bp.blogspot.com/-yJU7vxbyE7E/UZ95-Kr1FtI/AAAAAAAAAP4/OnZQrfhnVsM/s1600/dilate.png)

Dónde:

*detected\_edges : Fuente de la imagen, en escala de grises.*

*detected\_edges : salida del detector (puede ser la misma que la de entrada).*

*lowThreshold : El valor introducido por el usuario.*

*highThreshold : Situado en el programa como tres veces el umbral inferior (siguiendo la recomendación de Canny).*

*kernel\_size: Es el tamaño del kernelSobel a ser utilizado internamente.*

##### 2.7.7.4. Escala de grises

Transformar la imagen a escala de grises es otro filtro que reduce en la información. Todo filtro puede usarse principalmente para reducir información o para separar el área de interés.

Esta función de escala de grises se encuentra disponible en la librería de OpenCV



Figura 2‑10 Código para filtro a escala de grises

**Fuente**: [autor](http://3.bp.blogspot.com/-yJU7vxbyE7E/UZ95-Kr1FtI/AAAAAAAAAP4/OnZQrfhnVsM/s1600/dilate.png)

Previamente se mencionó dos funciones que también pueden ser usadas como filtros y que con las funciones Dilate y Erode, para más información podemos ver en la sección 2.7.3.

### 2.8. Reconocimiento facial

Para reconocimiento facial pueden existir varios métodos entre los cuales hay tres muy usados y disponibles en la librería instalada.

#### 2.8.1. Algoritmo Eigenface

Según Ottado (2010, p.2) este método realiza una proyección lineal del espacio de imágenes a un espacio de características de menor dimensión. Esta reducción se realiza utilizando la técnica PCA la cual toma aquella proyección lineal que maximiza la dispersión de todas las imágenes proyectadas.

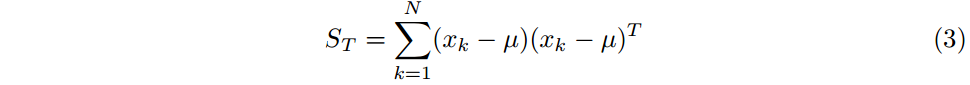
Primero se considera un conjunto de N imágenes de valores en el espacio de imágenes n-dimensional.



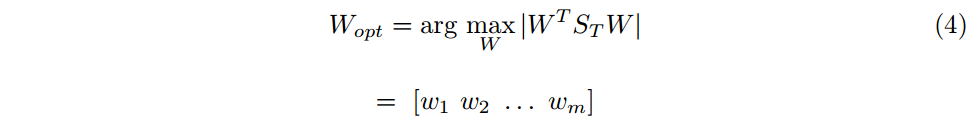
Asumimos también que cada imagen pertenece a una de las c clases {X1, X2, . . . , Xc}. Consideramos también una transformación lineal llevando el espacio original de las imágenes de n dimensiones al espacio de dimensión m, donde pues m < n. Los vectores resultantes de características yk∈ℜm son definidos por la siguiente transformación lineal (Ottado, 2010, p.3)



Donde W ∈ℜnxm es una matriz con columnas orto normales. También se define la matriz de distribución ST total



Donde µ ∈ℜn es la media de todas las imágenes de (1). Luego de aplicar la transformación lineal WT , la distribución de los vectores de características {y1, y2, ..., yN } es WT ST W. Se toma aquella proyección Wopt que maximiza el determinante de la distribución total de la matriz de las imágenes proyectadas, esto es



Donde {wi | i = 1, 2, . . . , m} es el conjunto de vectores propios n-dimensionales de ST correspondiente a los mayores m vectores propios. Dichos vectores propios tienen la misma dimensión que las imágenes originales y se les denomina eigenfaces.

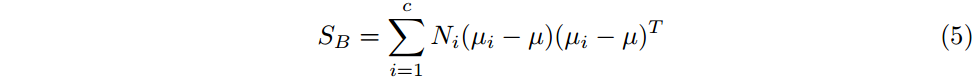


Figura 2‑11Eigenfaces de rostros

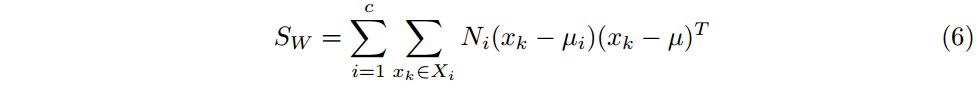
**Fuente**: https://eva.fing.edu.uy/file.php/514/ARCHIVO/2010/TrabajosFinales2010/informe\_final\_ottado.pdf

#### 2.8.2. Algoritmo Fisherface

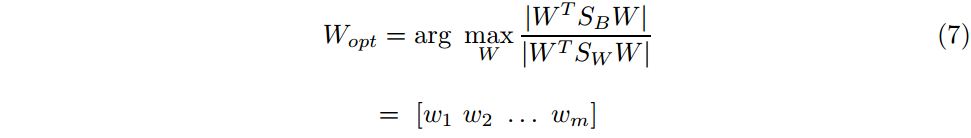
Así mismo explica Ottado (2010, p.3) que el método Fisherfaces utiliza el Discriminante Lineal de Fisher (FLD) para la reducción de dimensión. Este método selecciona el W de la ecuación (1) de manera que el cociente entre la distribución entre clases y la distribución intra-clases sea máxima. Para esto se define la matriz SB de distribución entre clases como



Y la matriz SW de distribución intra-clases



Donde µi es la imagen media de la clase Xi , y Ni es el número de imágenes en la clase Xi . Si la matriz SW es no singular, la proyección Wopt se elige como la matriz con columnas orto normal que maximiza el cociente del determinante de la matriz de distribución entre clases de las imágenes proyectadas y el determinante de la matriz de la distribución intra-clases de las imágenes proyectadas, esto es



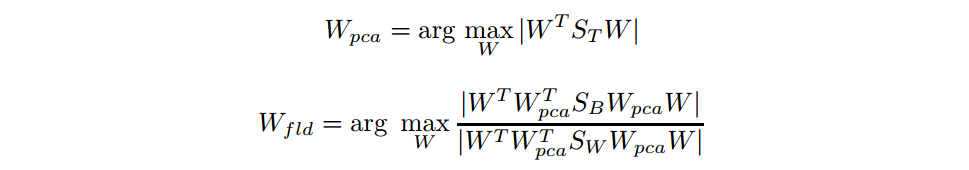
Donde {wi | i = 1, 2, . . . , m} es el conjunto de valores propios de SB y SW correspondiente a los m mayores valores propios {λi | i = 1, 2, . . . , m}, esto es



Se observa entonces, que a lo sumo se tienen c − 1 valores propios distintos de cero, y por lo tanto el límite superior de m es c − 1, donde c es el número de clases. Para el problema de reconocimiento de caras, se tiene que la matriz SW ∈ℜnxn es siempre singular, dado que el rango de SW es a lo sumo N − c, y en general, el número de imágenes de entrenamiento: N, es mucho más chico que el número de pixeles de cada imagen: n. Por lo tanto, puede ser posible elegir una matriz W tal que la distribución intra-clases de las imágenes proyectadas pueda ser exactamente cero. Como alternativa entonces, al criterio establecido en la ecuación (7), se proyecta el conjunto de imágenes a un espacio de menor dimensión, de manera que la matriz resultante de la distribución intra-clases SW es no singular. Utilizando PCA se realiza la reducción de dimensiones del espacio de características a N − c y luego, aplicar FLD definido en (7) para reducir la dimensión a c−1. De esta manera Wopt es dado por la siguiente ecuación (Ottado, 2010, p.4).



Donde



Las columnas de esta matriz se les refieren como fisherfaces (Ottado, 2010, p.2-5).

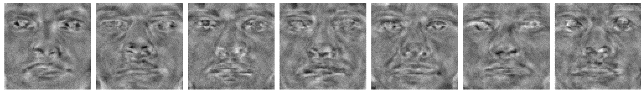
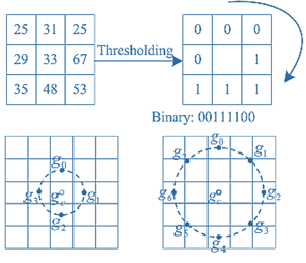


Figura 2‑12Fisherfaces de rostros

**Fuente**: https://eva.fing.edu.uy/file.php/514/ARCHIVO/2010/TrabajosFinales2010/informe\_final\_ottado.pdf

#### 2.8.3. Algoritmo LBPH

Para este algoritmo podemos destacar la explicación de Zhiguang (2007, p.2) que dice que el operador LBP Básico es un operador eficiente computacional. Tomando cada píxel como un umbral, el operador transfiere su vecindad 3 × 3 en un código binario de 8 bits, como se muestra en la Figura 13-2 (a). Más tarde, en, el operador LBP que se extiende un número arbitrario de píxeles interpolados bilineales en un círculo con tamaño arbitrario se usan como píxeles vecinos, en lugar de su vecindad 3 × 3, como se muestra en Figura 13-2 (b).



Operador LBP Básico b) Operador LBP extendido

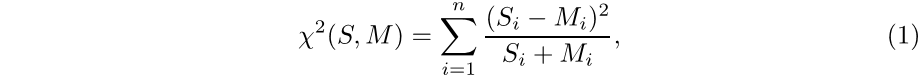
Figura 2‑13 Operadores LBP

**Fuente:** http://media.cs.tsinghua.edu.cn/~ahz/papers/ICB07\_demographic.pdf

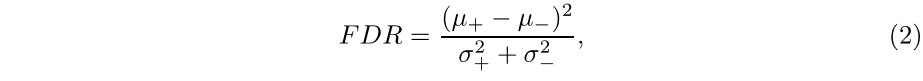
Para el operador básico LBP es la llamada LBP uniforme que se encontró que la propiedad fundamental de la textura de la imagen local. A LBP se llama uniforme si no hay más de dos 0/1 o 1/0 transiciones bit a bit en su código binario, siendo considerado como un código circular. El operador LBP extendido es donado como LBP, donde P es el número de píxeles interpolados bilineales, R es el radio del círculo vecino y u2 significa criterio uniforme. La contribución de patrón uniforme a LBP y la LBP es de aproximadamente 87,2% y 70,7%, respectivamente. Es decir, los patrones uniformes toman un porcentaje mayoritario de todos los patrones. Como resultado, cada patrón uniforme se da una etiqueta única y todas las demás minorías se les dan una etiqueta mutua en el cálculo del histograma. En este trabajo, el dolor lumbar se utiliza, y todo el valor de característica LBP es cuantificar en [1, 59] por criterio uniforme. Dado un parche textura local LBPH se utiliza para resumir todos los píxeles (excepto píxeles en el borde de la imagen) con cada píxel representado por la etiqueta cuantificada LBP (Zhiguang, 2007, pp.2-3).

2.8.3.1. Construcción del clasificador

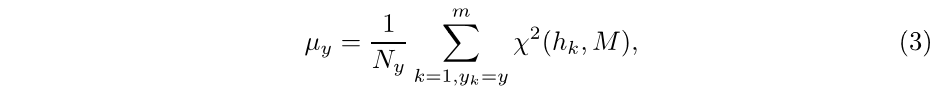
Si suponemos que S y M son dos diferentes histogramas, entonces la distancia de Chi cuadrado se la puede definir como:

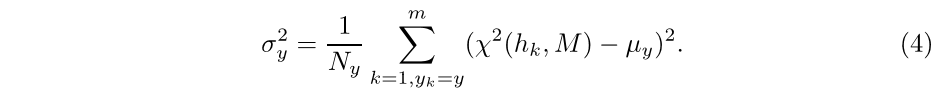


Donde n es el número de elementos en el histograma (n = 59 en este documento). Chi distancia cuadrado es una efectiva medida de similitud entre un par de histogramas, por lo tanto, es adecuado para el vecino más cercano. Sin embargo, encontramos un par de muestras similares no tiene sentido en la mayoría de problemas de clasificación binario. En esta sección, queremos hallar una óptima plantilla de histograma M como la plantilla de referencia para todas las muestras positivas o negativas para calcular una distancia de Chi cuadrado para el pedido. Con la hipótesis de distribución gaussiana de muestras positivas y negativas, respectivamente, se espera que la plantilla M estar cerca de un clúster y lejos de la otra, con el fin de que las muestras positivas y negativas se pueden discriminar con éxito por Chi cuadrado de la distancia. Vía Chi distancia plaza, la separabilidad de cada característica LBPH podría ser medido como proporción discriminante de Fisher (Zhiguang, 2007, pp.3).

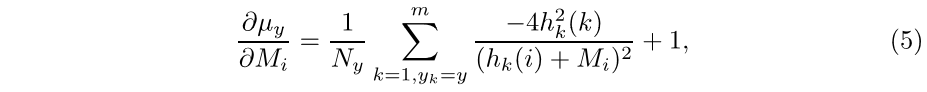


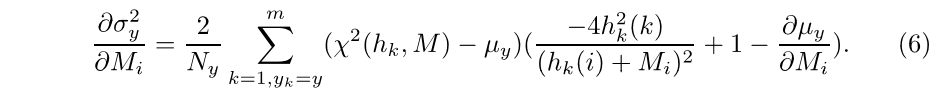
Donde μ y σ donar el valor de valor de la media y la varianza de Chi cuadrado distancia muestras 'a la plantilla M. Con el fin de hallar el histograma óptima plantilla, Primero se inicializamos M como el valor medio de histogramas positivos, y luego utilizamos el método decente empinada para hallar una solución óptima. Dada conjunto de muestras S = (h, y),. . . , (H, y), donde h es la característica LBPH, ey = ± 1 es la etiqueta de clase. μ y σ de muestras positivas y negativas ilustrados por y = 1 e y = -1, respectivamente, se pueden escribir como (Zhiguang, 2007, pp.3-4).



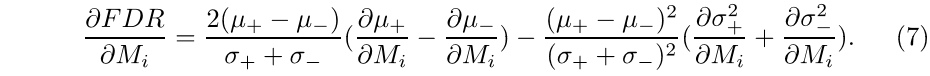


Para el histograma plantilla de n-elementos, cuyos artículos resumir en 1, el anterior n - 1 elementos se supone que son independientes, y su derivada parcial de μ y σ se puede escribir como:





Como resultado, el gradiente de Fisher discrimina relación se puede calcular como sigue



Por este medio, un M óptima podría ser encontrado por búsqueda iterativa, y se utiliza como la plantilla de referencia para el parche textura local dada. Dividimos el dominio principal característica de Chi cuadrado de la distancia a partir de muestras de formación a la plantilla de referencia en 32 contenedores, y la Figura 14-2 da una distribución ejemplo de Chi cuadrado distancia muestras al histograma de referencia, que se reúne la hipótesis previa de distribución de Gauss también (Zhiguang, 2007, p.4).

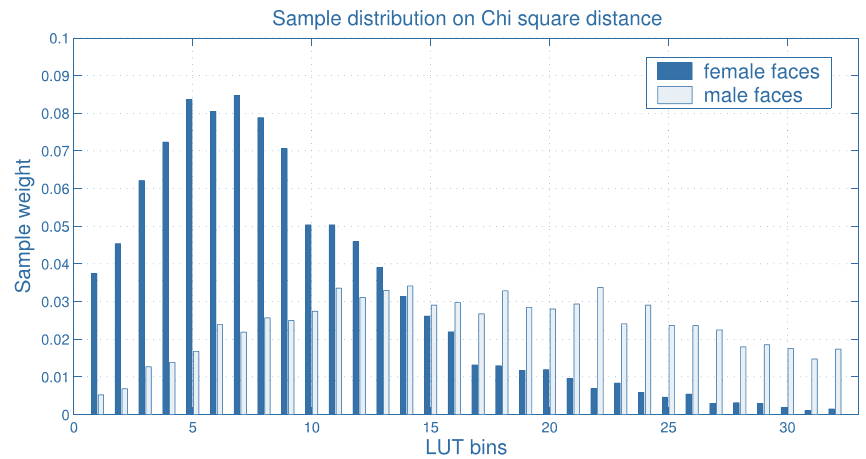
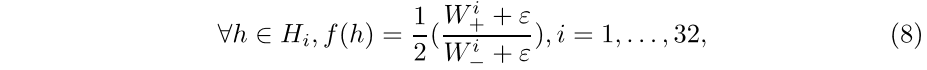


Figura 2‑14 Distribución de la muestra de Chi cuadrado distancia

**Fuente:** http://media.cs.tsinghua.edu.cn/~ahz/papers/ICB07\_demographic.pdf

Con respecto a la distribución de muestras de entrenamiento 'de la distancia Chi cuadrado, se usa un débil clasificador basado en LUT. Su salida en cada dominio puede ser definida como:



Donde W es el peso suma de muestras en el dominio H i-ésimo y ε es una pequeña constante positiva (Zhiguang, 2007, p.5).

### 2.9. Microcontroladores con arquitectura ARM

#### 2.9.1. Introducción a la arquitectura ARM

La arquitectura ARM ha evolucionado hasta un punto donde se apoya implementaciones en un amplio espectro de puntos de rendimiento. Más de dos mil millones de partes han enviado, estableciéndola como la arquitectura dominante en muchos segmentos de mercado. La simplicidad arquitectónica de procesadores ARM ha llevado tradicionalmente a muy pequeñas implementaciones y pequeñas implementaciones permiten los dispositivos con muy bajo consumo de energía.

Implementación tamaña, rendimiento y muy bajo consumo de energía siguen siendo atributos clave en el desarrollo de la arquitectura ARM.

La ARM es una Instrucción Reducida Set Computer (RISC), ya que incorpora esta típica arquitectura RISC

Características:

* Un gran archivo de registro uniforme
* Una arquitectura de carga / tienda, donde las operaciones de procesamiento de datos sólo funcionan en contenido del registro, no directamente sobre contenidos de la memoria
* Modos de direccionamiento simple, con todas las direcciones de carga / tienda que se determinan a partir de los contenidos de registro y Sólo los campos de instrucción
* Los campos uniformes e instrucción de longitud fija, para simplificar la decodificación de instrucciones.

Además, la arquitectura ARM proporciona:

* Control tanto sobre la unidad lógica aritmética (ALU) y la sección de cambios en la mayoría de las instrucciones de procesamiento de datos para maximizar el uso de una ALU y una palanca de cambios
* Incremento automático y auto-decremento modos de direccionamiento para optimizar bucles de pro-grama
* Cargar y almacenar instrucciones múltiples para maximizar el rendimiento de datos
* Ejecución condicional de casi todas las instrucciones para maximizar el rendimiento de ejecución.

Estas mejoras en la arquitectura básica RISC permiten procesadores ARM para lograr un buen equilibrio de alto rendimiento, tamaño pequeño código, bajo consumo de energía, y una pequeña zona de silicio (ARM Limited, 2005, p. A1-2).

#### 2.9.2. Características de la arquitectura ARM

RISC Arquitectura

* Reducción del conjunto de instrucciones - sólo 25 tipos de instrucciones básicas.
* La mayoría de las operaciones se ejecutan sobre registros.
* Todas las instrucciones pueden ser condicionales.
* Múltiples modos de direccionamiento se proporcionan (incluyendo los modos que permiten el desplazamiento de bits directa).
* Manipulación manual de pila.
* Pila abordar debe programarse de forma explícita.
* Subrutinas (incluido el retorno) debe ser explícitamente programado.

#### 2.9.3. Microcontroladores STM32

La familia STM32 de micro controladores flash de 32 bits basado en el procesador ARM ® Cortex® -M está diseñado para ofrecer nuevos grados de libertad para los usuarios de MCU. Ofrece una gama de productos de 32 bits que combina muy alto rendimiento, capacidades en tiempo real, procesamiento de señales digitales, y bajo consumo de energía, operación de baja tensión, mientras se mantiene la plena integración y facilidad de desarrollo.

La gama sin igual y gran cantidad de dispositivos STM32, basado en un núcleo estándar de la industria y acompañada de una amplia selección de herramientas y software, hace que esta familia de productos de la elección ideal, tanto para pequeños proyectos y para las decisiones de la totalidad de la plataforma.

#### 2.9.4. STM32CubeMX

STM32CubeMX es parte de STMicroelectronicsSTMCube ™ iniciativa original para facilitar la vida a los desarrolladores mediante la reducción de esfuerzos de desarrollo, tiempo y costo. STM32Cube cubre la cartera STM32.

STM32Cube incluye el STM32CubeMX que es una herramienta de configuración de software gráfico que permite generar código C de inicialización utilizando asistentes gráficos.

También incorpora una plataforma de software completa, entregado por serie (como STM32CubeF4 para la serie STM32F4). Esta plataforma incluye la STM32Cube HAL (un software integrado STM32 capa de abstracción, lo que garantiza la portabilidad maximizada a través del portafolio STM32), además de un conjunto coherente de componentes de middleware (RTOS, USB, TCP / IP y gráficos). Todos los servicios de software embebido vienen con un conjunto completo de ejemplos.

STM32CubeMX es una extensión de la herramienta MicroXplorer existente. Es una herramienta gráfica que permite configurar los microcontroladores STM32 muy fácilmente y generar el código de inicialización C correspondiente a través de un proceso paso a paso.

El primer paso consiste en seleccionar el micro controlador STMicroelectronics STM32 que coincide con el conjunto necesario de los periféricos.

El usuario debe configurar el controlador de cada integrado y gracias a un programa de solución Pinout se facilita el proceso, tiene un entorno ayudante reloj-árbol, una calculadora de consumo de energía, y una utilidad para realizar la configuración de la periférica (GPIO, USART,) y las pilas de middleware (USB, TCP / IP).

Finalmente, el usuario inicia la generación del código de inicialización C basado en la configuración seleccionada. Este código está listo para ser utilizado dentro de varios entornos de desarrollo. El código de usuario se mantiene en la próxima generación de código.

#### 2.9.5. MDK V5 (KEIL)

Keil MDK es la solución más completa de desarrollo de software para micro controladores ARM Cortex M. Incluye IDE, C / C ++, depurador, gestión Software Pack y CMSIS. MDK Versión 5 paquetes de software agregan componentes de soporte de dispositivo y software que se gestionan mediante el nuevo Run Time Environment (RTE). Depuración y traza capacidades en tiempo real mejoran con RTOS marcas de agua de pila y el Visor de sucesos.

El MDK - Professional Edition añade middleware y certificación TÜV para aplicaciones críticas de seguridad.

##### 2.9.5.1. MDK Core

μVision IDE y depurador.

ARM C / C ++ Compiler cadena de herramientas.

Empaca Installer para gestionar paquetes de software.

##### 2.9.5.2. Paquetes de Software

Dispositivos paquetes familiares agregan soporte y ejemplos de proyectos de dispositivos para placas de evaluación.

CMSIS es un marco de software estándar para microcontroladores Cortex- M.

MDK - Profesional Middleware añade componentes de software para la comunicación, almacenamiento de archivos y pantallas gráficas.

#### 2.9.6. STM Studio

STM Studio ayuda a depurar y diagnosticar aplicaciones STM8 y STM32 mientras se ejecutan mediante la lectura y la visualización de sus variables en tiempo real.

Correr en un PC, las interfaces STM Studio con STM8 y STM32 MCU a través de las herramientas de desarrollo estándar, como el bajo costo ST- LINK y RLink junto con el de gama alta sistema de emulación STM8 Stice.

STM Studio es una herramienta no intrusiva, preservando el comportamiento en tiempo real de aplicaciones.

STM estudio se complementa perfectamente con herramientas de depuración tradicionales para afinar aplicaciones. Es muy adecuado para la depuración de las aplicaciones que no se puede detener, como las aplicaciones de control de motores.

Diferentes vistas gráficas están disponibles para que coincida con las necesidades de depuración y el diagnóstico o para demostrar el comportamiento de la aplicación. STMicroelectronics (2015)

#### 2.9.7. Programación de comunicación universal serial asincrónica de un microcontrolador

##### 2.9.7.1. Comunicación UART

El corazón del sistema de comunicaciones serie es la UART, acrónimo de Universal Asyncronous Receiver-Transmiter.

Es un periférico cuya misión principal es convertir los datos recibidos del bus en formato paralelo, a un formato serie que será transmitido hacia el exterior.

También hace el proceso inverso (Benalcazar, 2012, p.2).

##### 2.9.7.2. Comunicación UART por sondeo

La comunicación serial asincrónica en los micro controladores con arquitectura ARM utiliza el mismo periférico para la comunicación sincrónica USART, entre las diferentes funciones que presenta es la recepción simple por sondeo, que quiere decir que nosotros podemos esperar el trabajo del micro controlador hasta que se complete una comunicación o en su defecto programar y conocer cuándo va a haber comunicación serial.

Esta puede ser programada y manejada en lazo infinito do while.

##### 2.9.7.3. Comunicación UART por interrupciones

La comunicación UART por interrupciones en un proyecto generado y ejecutado con STM32CubeMX y MDK v5 utiliza parámetros importantes que son la afinación de una prioridad para esta interrupción que se guardara en el vector de prioridades NVIC del micro controlador.

Existe un archivo de interrupciones ajeno al archivo main del proyecto el cual tiene las funciones y manejadores de las interrupciones configuradas en el micro controlador para un caso especial.

Dentro de estas funciones se puede agregar líneas de código del usuario a más de querer usar las mismas variables que en el archivo main se debe declararlas como externas para poder referirlas sin ningún problema.

#### 2.9.8. MIT APP Inventor y Bluetooth

APP Inventor es un entorno o IDE de programación en el cual podemos crear aplicaciones ejecutables en el Sistema operativo de Android creado por Google Labs, las características principales de este entorno son que es gratuito, online, sencillo, etc.

Es posible añadir funciones propias hechas en Java a APP Inventor para uso en nuestras aplicaciones, tiene la funcionalidad de poder probar las aplicaciones antes de instalarlas como si ya hubiesen estado instaladas con el uso de una red privada para el propósito.

El entorno de programación cuenta de dos partes principales que son: la parte de diseño donde podemos agregar y dar forma de la apariencia y funciones que va a tener la aplicación creando enlaces y las variables que podrán ser usados en la segunda parte que es la parte de la programación, donde los bucles principales de un lenguaje de programación y otros personalizados para las funcionalidades que puede ofrecer un Smartphone son representados en bloques o figuras.

Para poder tener acceso a este entorno de programación es necesario tener una cuenta con Google la cual puede servir para muchos otros más servicios con las que cuenta esta gran compañía.

#### 2.9.9. Comandos AT para para uso de red GSM y GPS

El módulo SIM908 es un avanzado terminal que puede ofrecer el servicio de comunicación en cuatro bandas.

Las funcionalidades del SIM908 pueden ser programadas mediante comunicación serial al módulo y con el ingreso de comandos AT los cuales son entendidos por el periférico.

Teniendo así a continuación una serie de comandos importantes para operar en la red GSM y GPS:

Tabla 2‑1 Lista de comandos AT usados para comunicación con SIM908

|  |  |  |
| --- | --- | --- |
| **Comando** | **Respuesta** | **Descripción** |
| AT | OK | Si usted recibe bien, la comunicación con el módulo está funcionando |
| AT + CPIN = "\*\*\*\*" | OK | Si la tarjeta SIM está bloqueada con PIN (\*\*\*\* es el número de PIN) |
| AT + COPS? |  | Información del operador |
| ATD \*\*\*\*\*\*\*\*\*; |  | \*\*\*\*\*\*\*\*\* Es el número al que llamar. |
| ATA | OK | Contestar una llamada entrante. |
| ATH | OK | Cancelar las llamadas de voz. |
| AT + CMGF = | OK | Especifica el formato de entrada y salida de los mensajes cortos. 0 para el modo PDU y 1 para el modo texto. |
| AT + CMGS |  | Envía un mensaje. |
| AT + CMGR = \* |  | Lee un mensaje. \* Es el número del mensaje. |
| AT + CGPSPWR | OK | Energía del GPS |
| AT + CGPSRST | OK | Establece el modo de restablecimiento |
| AT + CGPSSTATUS |  | Obtiene el estado: Desconocido, No Fix, Fix 2D y 3D Fix |
| AT + CGPSINF | Cadena NMEA | Obtiene cadenas NMEA |

**Realizado por:** Christian Salazar

**Fuente:** SIMCON, 2011, Manual de Referencia Septiembre 011

#### 2.10. Dispositivos y componentes del sistema.

#### 2.10.1. Tarjeta Discovery STM32F3.

STM32F3DISCOVERY es una tarjeta de desarrollo construida con el micro controlador stm32F303VCT6 de 32 bits ARM Cortex M4, cuenta con una gran cantidad de pines disponibles al usuario. Sobre la base de la tarjeta cuenta a más del micro controlador con la herramienta de depuración incrustado ST-LINK / V2, acelerómetro, giroscopio y brújula electrónica ST MEMS, conexión USB, LEDs y pulsadores.

Características principales:

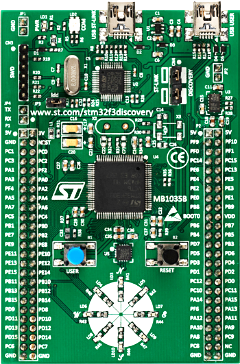
* Micro controlador STM32F303VCT6 con 256 KB Flash, 48 KB de RAM en un paquete LQFP100.
* A bordo ST-LINK / V2 con interruptor de modo de selección para utilizar el kit como autónomo ST-LINK / V2 (con conector SWD para la programación y depuración).
* Fuente de alimentación de tarjeta: a través del bus USB o desde una tensión externa 3 V o 5 V de alimentación.
* El poder de aplicación de alimentación externa: 3 V y 5 V.
* L3GD20 ST MEMS sensor de movimiento de 3 ejes de salida digital giroscopio
* LSM303DLHC, ST MEMS sistema-en paquete que ofrece un sensor lineal digital en 3D aceleración y un sensor magnético digital en 3D.
* Dos pulsadores (Usuario y Reajuste).
* USB de usuario con conector Mini-B.
* Cabecera de extensión para todos LQFP100 E/S para la conexión rápida a bordo de prototipos y fácil sondeo.
* Diez LEDs:
* LD1 (rojo) de 3,3 V de encendido.
* LD2 (rojo/verde) para la comunicación USB.
* Ocho LEDS de usuario de varios colores.

Figura 2‑15 STM32F3DISCOVERY

**Fuente:** http://www.st.com/st-web-ui/static/active/en/fragment/product\_related/rpn\_information/board\_photo/stm32f3discovery.jpg

#### 2.10.2. Módulo USART MAX232.

Es un módulo USB-TTL basado en CP2102 que crea un COM virtual en la PC como el mismo cable USB-Serial RS232 y crea una interfaz hacia UART TTL, normalmente usado para comunicar la PC con micro controladores y demás módulos que usan esta interfaz de comunicación, tiene opción de alimentar un circuito con 5 V, su fusible en la placa asegura que en caso de cortocircuito la corriente no exceda el limite soportado por el puerto USB (500 mA).

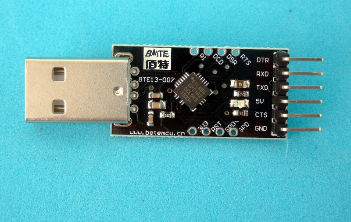


Figura 2‑16 USB 2.0 Serial RS232 UART TTL CP2102.

**Fuente:**http://www.tienda.siliceo.es/myfiles/cp2102\_USB\_uart\_ttl\_rs232\_cable\_serie\_1.jpg

#### 2.10.3. Módulo Bluetooth HC-05.

Es una pequeña tarjeta que sirve para comunicarse por medio de Bluetooth con otro dispositivo, el HC 05 tiene la particularidad de ser master/esclavo, el modulo viene configurado de fábrica para trabajar en modo maestro, algunos modelos cuentan con un botón, el cual al mantenerlo presionado por un pequeño lapso de tiempo nos permite comunicarnos con el modulo por comando AT para poder cambiar los parámetros que deseemos.

El modulo Bluetooth HC-05 utiliza el protocolo UART RS 232 serial.

Los pines de la board correspondientes son:

* EN
* VCC
* GND
* TX
* RX
* STATE

## Características

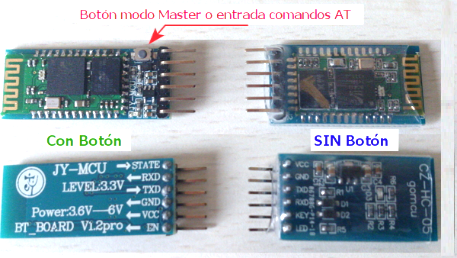
* Compatible con el protocolo Bluetooth V2.0.
* Voltaje de alimentación: 3.3VDC – 6VDC.
* Voltaje de operación: 3.3VDC.
* Baudrate ajustable: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.
* Baud rate pordefecto: 9600
* Tamaño: 1.73 in x 0.63 in x 0.28 in (4.4 cm x 1.6 cm x 0.7 cm)
* Corriente de operación: < 40 mA
* Corriente modo sleep: < 1mA

Figura 2‑17 Módulo Bluetooth HC-05.

**Fuente:** http://www.cecilioruiz.com/wp-content/uploads/2014/02/hc05.jpg

#### 2.10.4. Módulo GSM/GPRS/GPS SIM908.

La tarjeta o módulo de desarrollo GSM/GPRS/GPS SIM908 es QuadBand, tiene como interfaz de comunicación la UART 232 para el módulo GSM al igual que otro independiente para GPS, teniendo la opción de obtener resultados de GPS en el módulo GSM, puede operar en redes GSM de 850 MHz, así como en la banda de 900 MHz (EGSM), DCS 1.800 y 1.900 MHz, es compatible con GPRS multi-slot clase 10 y 8 y esquemas de codificación CS-1, CS-2, CS-3 y CS-4.

La energía de la célula se coloca en la clase 4 (2W) en redes EGSM 850 y GSM 900, y la Clase 1 (1W) a DCS 1800 y PCS 1900. En lo que respecta a la parte de GPS, es de 42 canales y ofrece un excelente rendimiento en términos de tiempo requeridos para el primer punto de referencia (TTFF, Time To FirstFix): 1s de acuerdo con el arranque en caliente y 30 s en el arranque en frío.  
En términos de conectividad, este módulo tiene la maravillosa interfaz serie de costumbre, que también integra la funcionalidad de TCP / IP y extendido los comandos AT para su gestión.

La sensibilidad es -143 dBm y -160 dBm en modo normal y el seguimiento de la actualización es de no menos de 1 Hz, lo que significa que disponen de datos sobre la posición (exacta a menos de 2,5 metros) se actualiza cada segundo.

Las dimensiones extremadamente compactas (30x30x3, 2 mm), le permiten desarrollar dispositivos muy compactos como localizadores, dialers, telecontrol GSM, alarmas remotas, etc. El módulo es de montaje superficial (SMT), y no requiere enchufe.

****

Figura 2‑18 Módulo SIM908 GSM GPRS GPS

**Fuente:** http://g04.a.alicdn.com/kf/HTB1Ndj5HVXXXXXgXXXXq6xXFXXXE/-font-b-SIM908-b-font-Module-GSM-GPRS-GPS-Development-Board-IPX-SMA-with-GPS.jpg

#### 2.10.5. Raspberry PI 2.

La Raspberry pi 2 Modelo B es un mini PC y es la segunda generación de Raspberry Pi, sustituyó a Raspberry Pi 1 Modelo B+ en febrero del 2015.

Por tener un procesador ARMv7, puede funcionar con la gama de distribuciones ARM GNU / Linux, así como Microsoft Windows 10.

Raspberry Pi 2 tiene:

* Un CPU de 900 MHz de cuatro núcleos ARM Cortex-A7.
* 1 GB de RAM.
* 4 puertos USB.
* 40 pines GPIO.
* Puerto HDMI Full.
* Puerto Ethernet.
* Conector de audio de 3,5 mm combinado y video compuesto.
* Interfaz de la cámara (CSI).
* Interfaz de pantalla (DSI).
* Ranura para tarjeta micro SD.
* Núcleo de gráficos VideoCore IV 3D.

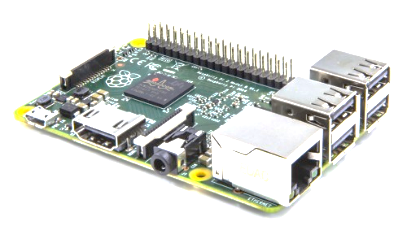


Figura 2‑19Raspberry PI 2 modelo B

**Fuente:** https://www.raspberrypi.org/wp-content/uploads/2015/01/Pi2ModB1GB\_-comp-500x283.jpeg

#### 2.10.6. Cámara Webcam

La cámara webcam de tipo VGA cuenta con la opción de Plug and Play en los sistemas operativos de LINUX, cuenta con seis diodos led que ayudan a la captura de imágenes en la obscuridad, tiene un soporte modificable y de fácil uso para ubicarla a conveniencia del usuario.



Figura 2‑20 Cámara Webcam

**Fuente:** http://ecx.images-amazon.com/images/I/6106Vce214L.\_SL1000\_.jpg

#### 2.10.7. Atmega 328.

El Atmega328 AVR 8-bits es un Circuito Integrado de alto rendimiento basado en micro controlador RISC, 32KB ISP flash una memoria capaz de leer mientras escribe, 1 KB de memoria EEPROM, 2KB de SRAM, 23 líneas de E/S, 32 registros de proceso general, tres temporizadores flexibles/contadores, interrupciones internas y externas, programador modo USART, SPI puerto serial, 6 canales de 10 bits conversores A/D y una frecuencia de operación máxima de 16 MHz.

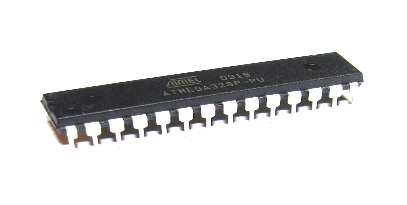


Figura 2‑21 IC Atmega328P

**Fuente:** http://www.nkcelectronics.com/assets/images/atmega328p.jpg

#### 2.10.8. Convertidor STEP UP STEP DOWN XI 6009

Este convertidor DC-DC tiene un integrado XL6009 construido por transistores NMOS, con una eficiencia mayor que el tríodo LM2577 y menor al 94%, es automáticamente ajustable por un potenciómetro de precisión para cambiar el voltaje de salida, pudiendo brindar hasta 35 V a 3A de corriente.

Características:

* Entrada de voltaje de 3,2 V a 32 V.
* Salida de 1,25 V a 35 V

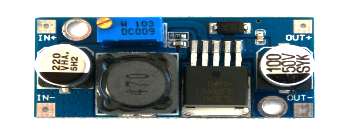


Figura 2‑22 Convertidor step up step down XI 6009

**Fuente:** http://www.flytron.com/978-thickbox\_default/dc-dc-adjustable-step-up-converter-module-xl6009-replace-lm2577.jpg

#### 2.10.9. Sensor de temperatura LM35.

El LM35 es un sensor de temperatura de Circuito Integrado que entrega una salida eléctrica proporcional a la temperatura en °C, puede medir la temperatura con mayor precisión que un termistor, por su empaquetado es poco propenso a la oxidación, genera una tensión mayor a la generada por los termopares y puede no requerir amplificación de la salida.

Mantiene una precisión de ± 0,4°C a temperatura ambiente y +/-0.8°C en un rango de 0°C a 100°C.

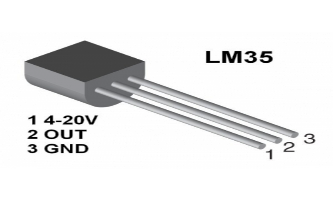


Figura 2‑23 Sensor de temperatura LM35

**Fuente:** http://blog.rastating.com/content/images/2014/Feb/LM35\_500x500\_1\_.jpg

#### 2.10.10. Sensor de distancia SHARP GP2Y0D340K.

El sensor SHARP GP2Y0D340K es de tipo reflectivo ya que tiene un emisor de luz infrarroja y un receptor fototransistor los cuales se obtiene a la salida del sensor una tensión o señal digital si se ha activado o no el sensor detectando un cuerpo en un rango de 40 cm, siendo un 0 lógico si se activa y un 1 lógico si se desactiva, se alimenta con una tensión de funcionamiento de 4,5 V a 5,5 V y una corriente de consumo de 40 mA, tiene un rango de medición de 0 cm a 40 cm, con una tasa de muestreo de 60 Hz, en el datasheet del sensor podemos encontrar el filtro adecuado para su correcto funcionamiento.



Figura 2‑24 Sensor Sharp GP2Y0D340K 0cm a 40cm

**Fuente:**https://i2.wp.com/www.robotshop.ca/Images/big/en/sharp-gp2y0d340k-ir-sensor.jpg

#### 2.10.11. Sensores switch y/o pulsadores.

Este tipo de elemento electrónico son sensores de salida digital y que entre otras aplicaciones con estos se puede obtener el estado de las puertas de los automóviles, existen de dos tipos: normalmente abiertos que al pulsarlos o accionarlos cierran el circuito y los normalmente cerrados que hacen lo contrario, la configuración más común para trabajar con micro controladores es usando una resistencia pull-up.

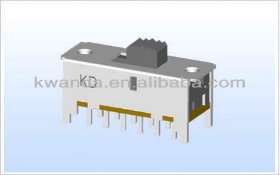


Figura 2‑25 Sensor switch

**Fuente:** http://spanish.alibaba.com/product-gs-img/tipo-vertical-4p2t-pines-electr-nicos-interruptor-deslizante-410784163.html

#### 2.10.12. Salidas a módulos relés mecánicos de potencia

Esta es una placa diseñada con cuatro relés mecánicos a 5V, los cuales se activan con la energía entregada por los transistores individuales para cada relé, estos transistores están opto acoplados para poder independizar la señal de entrada de la energía de polarización de los relés y de ser el caso se puede compartir la misma energía por el jumper que cuenta el diseño, además de los respectivos conectores para la potencia y para la señal de activación.

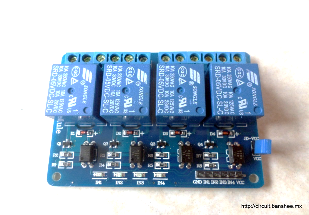


Figura 2‑26 Módulo de relés a 5V

**Fuente:** <http://mco-s2-p.mlstatic.com/modulo-rele-de-4-canales-para-arduino-20386-MCO20189281690_102014-F.jpg>

#### 2.10.13. Sabertooth 2x5.

Este es un módulo Sabertooth 2x5 que tiene la funcionalidad de un puente H con regulación de giro y rangos modificables mediante un dipswitch, tiene conectores para suministrar 5V a 100mA máximo, entrada de dos señales analógicas a 5V máximo pudiendo estas estar en PWM directo sin filtro aunque en el datasheet se puede encontrar el diseño adecuado del filtro, a sus salidas tiene dos canales los cuales suministran 6V a máximo 5A los cuales son obtenidos de la entrada de fuente la cual debe ser superior a un valor fijo y siendo regulado posteriormente, este módulo tiene la característica de ser un puente H regenerativo con la inercia para no desperdiciar energía.

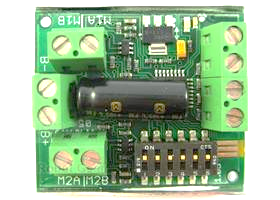


Figura 2‑27Sabertooth 2x5

**Fuente:** <http://www.robotgear.com.au/Cache/Files/ProductImageOriginals/695_Sabertooth%20dual%205A%20motor%20driver%20-%20Top.jpg>

#### 2.10.14. Sensor de alcohol MQ-3

El sensor de alcohol MQ-3 cuenta con tres terminales, se alimenta a una tensión de 5 V en dos de ellos, en la tercera línea es la salida analógica relacionada al nivel de alcohol que se encuentra en ambiente alrededor del sensor, tiene un potenciómetro para calibrar la sensibilidad del sensor.



Figura 2‑28 Sensor de alcohol MQ-3.

**Fuente:** https://abra-electronics.com/images/thumbnails/280/280/product/100/SEN0128\_\_79499.jpg?t=1415451398

#### 2.10.15. Sensor de luz LDR

El sensor de luz LDR es una fotocelda resistiva, la cual varía su resistencia conductiva conforme se varía el flujo de luz sobre el área relevante del sensor, es un sensor sensible, hay sensores que dan resistencia en el orden de los kilo ohmios.

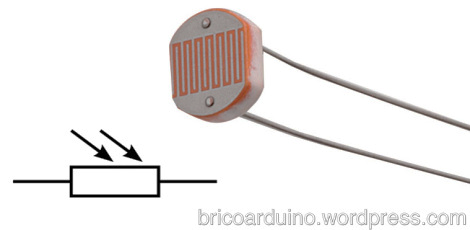


Figura 2‑29 Sensor de luz LDR

**Fuente:** https://bricoarduino.files.wordpress.com/2013/03/ldr.jpg?w=470&h=229

#### 2.10.16. Joystick

El modulo joystick para Arduino es un arreglo mecánico que componen dos potenciómetros, uno para cada eje (x,y) y un botón con la configuración electrónica con resistencia pull-up, se alimenta normalmente con 5 V y a su regreso entrega una señal analógica por la variación en cualquiera de sus ejes, tiene unos muelles que se encargan de mantenerlo centrado en sus ejes a medio valor idealmente, al hacer presión en el eje z se accionara el botón.

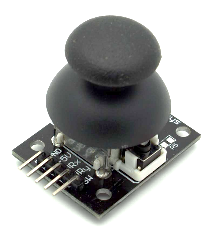


Figura 2‑30 Joystick

**Fuente:** <http://www.electronica2001es.com/images/arduino%20joystick%20.jpg?osCsid=m6fesnnbuu3nqhg0hr7ptj75q2>

#### 2.10.17. Potenciómetros

Un potenciómetro es una resistencia variable que en este caso y diseño convierte la variación de giro del sujetador en una variación eléctrica, sea en voltaje o en corriente y dando de acuerdo a la alimentación de sus terminales extremos un equivalente al giro en voltaje o corriente en la patilla sobrante, siendo por ende el terminal de salida.



Figura 2‑31 Potenciómetros

**Fuente**: http://www.cetronic.es/sqlcommerce/ficheros/dk\_93/productos/451220018-1.jpg

#### 2.10.18. LED SMD 5050

Led SMD 5050 de medidas estándar 1210 th (milésimas de pulgada) de alto brillo con 120 grados de apertura, tiene las mismas configuraciones que un diodo led normal, cuenta con un corte en una de sus esquinas identificando así el cátodo del diodo.

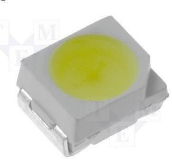


Figura 2‑32 Led SMD 5050

**Fuente:** http://image.ec21.com/image/wanglu615010/oimg\_GC05294067\_CA05595235/SMD\_5050\_%26\_1210\_LED\_Chips.jpg

#### 2.10.19. Resistencias SMD

Son elementos electrónicos que se oponen al paso de la corriente, del tipo de montaje superficial, usados en la sináptica de la electrónica, vienen en medidas estándar y de acuerdo a ellas su capacidad de potencia y codificado su valor en tres números, representando los dos primeros a cifras de su valor y el ultimo es el número de ceros que hay que unir a los dos primeros para leer el valor.

****

Figura 2‑33 Resistencias SMD

**Fuente:** http://img-europe.electrocomponents.com/largeimages/R0566636-01.jpg

#### 2.10.20. Zumbador

Es un actuador de tipo sonoro, que se puede usar como alerta para algunas aplicaciones, su funcionamiento se basa en la resonancia magnética para producir el sonido por golpes a cierta frecuencia, usa 6 V para inducir el campo magnético necesario.

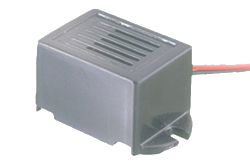


Figura 2‑34 Zumbador

**Fuente:** http://www.diotronic.com/imgs/35\_000.jpg

#### 2.10.21. Micro motor POLOLU

Es un micro motor a 6 V, existe modelos HP (High Power), que traduce a 1,6 A, cuenta con una caja de engranajes metálicos para reducir la velocidad de rpm del eje del motor al eje de la caja reductora.

****

Figura 2‑35 Micro motor Pololu

**Fuente:** http://www.tme.eu/html/gfx/ramka\_15354.jpg

#### 2.10.22. Motor POLOLU A 12V

Al igual que el modelo anterior cuenta con una caja, la diferencia está en su diseño eléctrico que es a 12 V y 5 A con una reducción en su caja de 26:1 que traduce que por cada 26 vueltas que da el eje del motor, entonces da una vuelta el eje de la caja reductora.

****

Figura 2‑36 Motor Pololu a 12V

**Fuente:** https://a.pololu-files.com/picture/0J901.200.jpg?7715ab68301097f0568e1d0c2904985b

#### 2.10.23. Tarjeta Discovery STM32F4

La tarjeta DISCOVERY STM32F4 al igual que la F3 es una tarjeta de desarrollo con arquitectura ARM Cortex M4, capacidades de soportar alimentaciones de 3,3 V y 5 V, cuenta con cuatro Leds para uso general, un pulsador para resetear y uno de usuario, además de alrededor de 60 pines disponibles y conectados a los periféricos que se encuentran en el micro controlador.

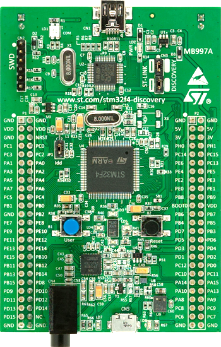
****

Figura 2‑37 Tarjeta DISCOVERY STM32F4

**Fuente:** http://www.st.com/st-web-ui/static/active/en/fragment/product\_related/rpn\_information/board\_photo/stm32f4\_discovery.jpg

#### 2.10.24. Wandboard Dual FREESCALE i.MX6 CORTEX A9

Es una tarjeta de desarrollo mini ordenador de dos núcleos de Freescale que usa un procesador ARM, cuenta con un puerto USB, un micro USB, dos ranuras para memorias micro SD, salida HDMI, audio 3.5 mm, VGA, botón de reset, puerto Ethernet, procesador gráfico, conector CSI para cámara, módulos Bluetooth y Wifi, además de una amplia gama de GPIOS.

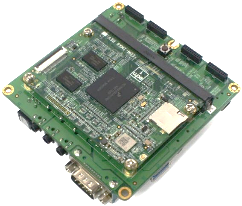
****

Figura 2‑38 Wandboard dual Freescale i.MX6 Cortex A9.

**Fuente**: http://www.mouser.com/images/microsites/Wandboard\_DUAL.jpg

# 

## 3. MARCO APLICATIVO

### 3.1. Diseño del sistema

Se procede a enumerar las distintas partes hardware y software del sistema en etapas separas, se tomó la decisión de realizar por fases para poder independizar en partes fuentes y partes que están diseñadas para cumplir funciones diferentes unas que otras lo que al diferenciarlo en secciones pude ayudar a la comprensión del funcionamiento del sistema.

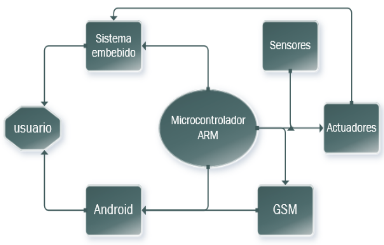


Figura 3‑1 Diagrama de bloques del abstracto del sistema completo

**Fuente**: Autor

#### 3.1.1. Estructura Base

Para métodos didácticos y demostraciones se va a conseguir una estructura o base rígida y abierta de tal manera que muestre o tenga a la vista la electrónica y demás.

#### 3.1.2. Etapa de control

En esta etapa debe contar con los microcontroladores o sub etapas, las cuales se encarguen de tomar decisiones.

Un micro controlador ARM STM32F3 cuenta con las características más que suficientes para poder llevar a cabo la función de controlar el sistema, ya que las lecturas de las conversiones de analógico al digital lo hace en 12 bits lo cual mucho es mejor a lo ofrecido por otros micro controladores como AVR o PIC, hay que mencionar que existen micro controladores STM32 con conversores denominados analógico a digital SIGMA-DELTA que tiene las caracterizas de ofrecer ganancias en sus entradas y una resolución de hasta 16 bits lo que aumentaría la precisión en las lecturas, un recurso muy útil para hacer lecturas de temperaturas con sensores industriales con grandes rangos de variación y la medición de peso las cuales usan sensores con variaciones muy pequeñas que un micro controlador normal puede no percibir, por esta razón se procedió a optar el uso de un sensor de temperatura LM35 que tiene un rango y ganancia adecuados y más que suficiente para le resolución de 12 bits de los STM32.

Para interactuar con el módulo SIM908 se inclinó por la opción de usar un micro controlador Atmega328 que se encargará de comunicarse por UART ya que es el protocolo el cual entiende el módulo y que se lo ha determinado como persistente ya que está a la espera o se detiene hasta que el módulo le responda ya que es una situación que se ve un poco obligada puesto que el sensor GPS tiene una latencia bastante considerable para recolectar la ubicación de los satélites disponibles la cual no es manejable por el micro controlador STM32.  
A más de las partes antes mencionadas tiene conectores para enlazar con las demás etapas respectivamente.

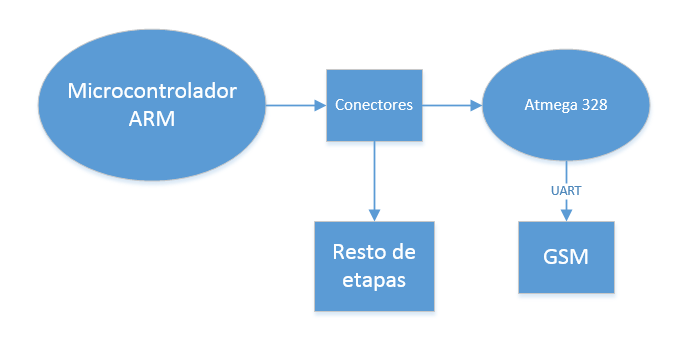


Figura 3‑2 Diagrama de bloques de etapa de control

**Fuente**: Autor

#### 3.1.3. Etapa de adquisición de datos

Aquí se conectará una alimentación de 3,3 voltios los cuales con más que suficientes para la medición de los sensores lo cual podría representar una pequeña ventaja ya que no se necesitaría un proceso de acondicionamiento de señal para introducir esta información al micro controlador ya que funciona a 3,3 voltios.

Por la necesidad de que los sensores deben estar en el campo de medida y no cerca del campo de control se pondrá conectores para sacarlos a los sensores y conectores para el micro controlador.

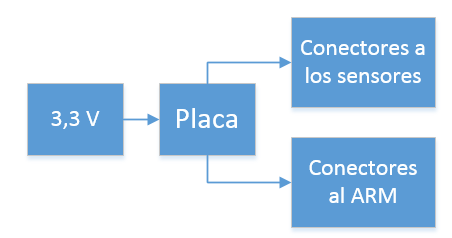


Figura 3‑3 Diagrama de bloques de etapa de adquisición de datos

**Fuente**: Autor

#### 3.1.4. Etapa de potencia

Por la necesidad del manejo de cantidades considerables de corriente debe estar aislado el circuito por acopladores y por ello se procederá a usar módulos de relés los culés tienen la opción de hacer un doble acoplamiento ya que tiene opto acopladores y acopladores mecánicos, al usar estos módulos no hay necesidad de usar acopladores de señal de salida para amplificarlo ya que cuenta con transistores encargados de esta función para los relés mecánicos.

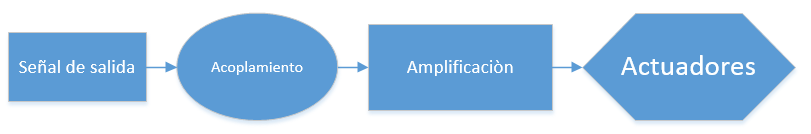


Figura 3‑4 Diagrama de bloques de etapa de potencia

**Fuente**: Autor

### 3.2. Selección del Sistema Embebido

En este punto se puso en comparación dos tarjetas como son las Raspberry Pi 2 modelo B y la Wandboard Dual iMX6.

Tabla 3‑1 Tabla comparativa entre Raspberry Pi 2 y Wandboard iMX6 Dual

|  |  |  |
| --- | --- | --- |
|  | Raspberry Pi 2 | Wandboard iMX6 Dual |
| Procesador | ARM CorteX-A7 QUAD de 0,7 GHz | ARM CorteX-A9 DUAL de 1.2 GHz |
| Memoria | 1 GB | 1 GB |
| Gráficos | 250 MHz | 320MHz |
| HDMI | 1 | 1 |
| C. Cámara | 1 | 1 |
| Micro SD | 1 | 2 |
| USB | 4 | 2 |
| Ethernet | 1 | 1 |
| Wifi | 0 | 1 |
| Bluetooth | 0 | 1 |
| P. Serial DB9 | 0 | 1 |
| Audio | 1 | 1 |
| Costo | 85 | 200 |

**Realizado por:** Christian Salazar

**Fuentes:** WANDBOARD, Año Desconocido, Datasheet y Adafruit, 2015, DatasheetFebreo 015.

En cuanto a características generales de las dos tarjetas podemos notar que Wandboard tiene el doble de características mejores que la Raspberry Pi 2 lo que podría decir a breves rasgos que sería mejor, pero viendo a más profundidad cada una de ellas tenemos:

El procesador de Raspberry Pi 2 es 0.4 Ghz mayor que el anterior lo cual lo hace muy bueno ya que procesaremos grandes cantidades de información.

En cuanto a gráficos la Wandboard parece ser la más adecuada, pero es irrelevante para la selección en esta aplicación ya que no es necesario mostrar gráficos que requieran mucho procesamiento de este tipo porque además se pude dejar corriendo la aplicación solo en modo consola.

En las tarjetas de memoria micro SD y los Puertos USB podría juntarse ambos eliminando una ranura micro SD en cada tarjeta ya que es necesaria para la instalación del Sistema Operativo dejando al resto para periféricos y almacenamiento compensando así la Raspberry al tener más puertos USB.

Wifi es una ventaja muy grande en la Wandboard lo cual lo hace sin duda un dato relevante por sus diversas ventajas así mismo como las de Bluetooth.

Por otro lado, el puerto serial no es muy relevante en esta aplicación ya que con el USART que tiene las Raspberry es suficiente a menos que se necesite más control sobre el protocolo.

En cuanto a costo es evidente que Raspberry lleva ventaja por su bajo precio en comparación con la Wandboard.

Para el Raspberry se probó con el sistema Operativo Raspbian aconsejado por Raspberry Pi Foundation que permite instalar y probar sin ninguna complicación el software necesario como lo es OpenCV completo y corriendo ejemplos sin ningún problema.

Lo que por el contrario Wandboard se probó con Ubuntu 15 para Wandboard que a pesar de estar sustentado por la organización Canonical presento problemas para la instalación del software antes mencionado.

Entonces por la relevancia del procesamiento, numero de puertos USB, precio y las ventajas y respaldo que tiene Raspbian se optó por usar Raspberry tomando la opción de adquirir los módulos individuales de wifi y Bluetooth que encarece la tarjeta dejando a un costo aproximado de la Raspberry de la mitad que la Wandboard.

### 3.3. Comunicaciones del sistema

#### 3.3.1. Comunicación STM32F3-RASPBERRY

Los métodos disponibles para comunicar las tarjetas ST32 y la Raspberry son USART, SPI e I2C.

El micro controlador no tiene mayores problemas al usar estos protocolos, ya sea por interrupciones, DMA o por sondeo simple, dejando la decisión en cuanto a las posibilidades y opciones mucho más cómodas y viables a implementar en la Raspberry.

En cuanto a UART tiene mayor velocidad de transmisión disponible, mayor distancia, menor número de cables.

SPI e I2C al ser protocolos destinados a la comunicación específicamente con periféricos extras y no entre sistemas como es este caso podrían ser un dato relevante para descartarlos.

La facilidad del uso de los puertos seriales en los sistemas operativos con la librería apropiada da otro punto a favor a UART.

En las pruebas y desarrollo de la interacción de OpenCV con un micro controlador se lo hizo con lenguaje C++, teniendo en cuenta que SPI e I2C cuenta con librerías escritas en lenguaje Python al igual que el periférico UART se puede optar por usar una librería general para manejar puertos seriales desde OpenCV.

Dejando como resultado que la opción más viable es usar el UART

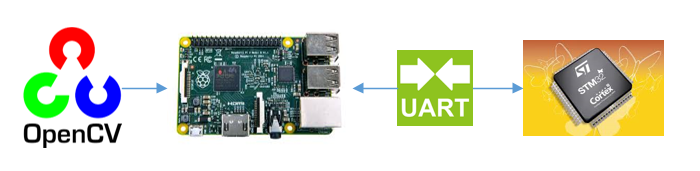


Figura 3‑5 Comunicación Sistema Embebido - Micro controlador

**Fuente**: Autor

#### 3.3.2. Comunicación STM32F3-Módulo Bluetooth

Por defecto del diseño del módulo HC 05 que se pretende usar es posible usar el protocolo de comunicación serial que es el único disponible.

#### 3.3.3. Comunicación ATMEGA328p-Modulo GSM

Al igual que el caso anterior el módulo SIM908 tiene disponible el uso del protocolo UART para interactuar con el atmega328.

### 3.4. Etapa de potencia y alimentación del sistema

Para un caso ya aplicado el uso de la batería disponible en el automóvil es suficiente para el sistema con las protecciones suficientes y mientras que para demostración se puede usar fuentes diferentes e independientes unos de otros en algunos casos.

### 3.5. Interfaz de control con Android

Para interactuar con el sistema podríamos usar un Smartphone con su respectiva aplicación lo cual nos puede obviar del uso de controles remotos extras, teclados extras para ingresar claves, receptor de coordenadas de ubicación de GPS, aplicación disponible para la localización.

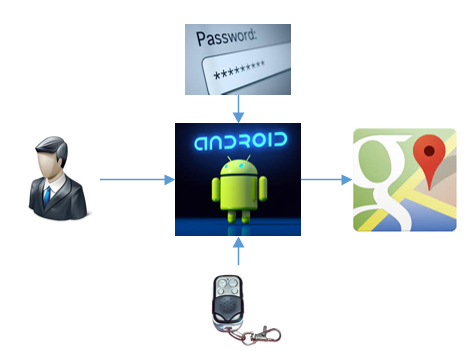


Figura 3‑6 Interfaz de control con Android

**Fuente**: Autor

La aplicación fue desarrollada en APP Inventor, es una aplicación personalizada para el sistema ya que se usa la MAC ADDRESS del módulo Bluetooth en el código fuente para conectarse al mismo desde el Smartphone, teniendo así botón en la parte superior para conectar al sistema, un botón en la parte inferior para salir de la aplicación, un botón de apagado de las luces o faros opcional si no se desea entrar en el modo seguro en el cual apaga las estas, tiene la parte de geo localización un text box que se llena al recibir las coordenadas por SMS desde el sistema y que al presionar el botos de buscar el mapa nos dará la ubicación aproximada del módulo GPS, así también un botón extra para ver mi localización y finalmente la parte de la clave de súper usuario que da arranque previo a estar en modo conducción que usa una clave secreta de cuatro dígitos, a continuación se muestra una figura de la apariencia de la aplicación.

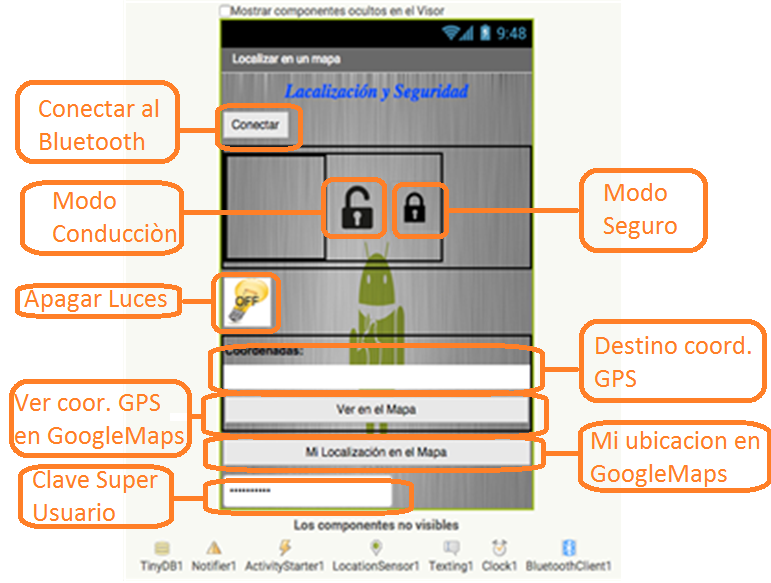


Figura 3‑7 Funciones de la aplicación de Android

**Fuente**: Autor

### 3.6. Reconocimiento facial

#### 3.6.1. Selección de cámara

Para la aplicación que se pretende llevar a cabo la cámara es un dispositivo muy importante e imprescindible, por lo que se ha seleccionado tres tipos de fuentes de imágenes para ser analizadas en el Raspberry, estos tres tipos pueden ser conectados a la Raspberry y se pone a consideración las siguientes características para poder seleccionar la más se adapte al sistema.

Tabla 3‑2 Tabla comparativa entre los tres tipos de cámara

|  |  |  |  |
| --- | --- | --- | --- |
|  | Cámara Ip con Smartphone | Cámara web USB | Cámara Pi |
|
| Resolución | 3 Mp | 320x240(0,76Mp) | 5 Mp |
| Conexión | Wifi | USB | SCCB |
| Streaming | Lento | Rápido | Rápido |
| Flash | No | Si | No |
| Auto Focus | Si | No | Si |
| Costo | Incluye | 20 | 65 |
|  |  |  |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

En las pruebas que se pudo realizar ayuda para descartar la posibilidad del uso de la cámara Ip simulada con el Smartphone es que tiene una latencia muy grande en llegar a parte de procesamiento lo que podría dificultar en gran dimensión el sistema llevando esta latencia a otros lados y en segunda instancia la conexión seria suplida con el uso de un módulo wifi en el Raspberry para poder transmitir haciendo al Smartphone o al Raspberry que trabaje como router de datos.

En segundo plano tenemos las dos últimas posibilidades que en cuanto a resolución es suficiente con la más baja ofrecida ya que la aplicación no requiere e incluso se puede requerir en la disminución del tamaño por lo que estaría sobredimensionado los 5 Mp de la cámara Pi, la conexión de USB no se ve afectada en su velocidad por la webcam puesto que contamos con los puertos suficientes y uno de ellos puede ser destinado para este propósito, mientras que la conexión SCCB puede ser un poco dificultosa al llevarlo a la práctica ya que para transmitir la información usa cableado súper sensible, de difícil manipulación y de dificultoso acceso al cable flexible como a la misma cámara Pi.

Al tener un juego de diodos led que pueden ayudar a l autenticación en la noche lo hace aún más viable su uso.

Y por último la relación beneficio costo determinan finalmente que el uso de una webcam USB es suficiente para la aplicación y hay que tomar en cuenta que la tecnología está muy avanzada y que por unos dólares más se puede tener una de aun mayor resolución, se tomó la menor resolución ya que es más que suficiente en este caso.

#### 3.6.2. Autenticación en la noche

Para la obtención de imágenes por la noche se tienen varias opciones como son el uso de cámaras estrictamente nocturnas, pero de grandes costos que por el análisis anterior queda descartado.



Figura 3‑8 Cámara de visión nocturna

**Fuente**: http://www.electronica-basica.com/images/gafas-vision-nocturna.jpg

Una segunda opción podría ser el uso de diodos led infrarrojos con un Angulo de apertura de luz muy amplio que hacen difícil conseguirlos, puesto que normalmente se encuentran son los diodos led infrarrojos con un Angulo muy cerrado ya que para las aplicaciones que se han llevado a cabo han sido necesarios como hacer un emisor receptor.

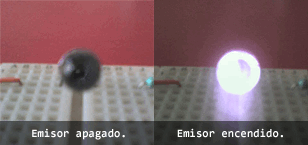


Figura 3‑9 Led infrarrojo

**Fuente**: http://1.bp.blogspot.com/\_jDvul0Mj1So/S\_7vf-ylWkI/AAAAAAAABfs/IPsmgafWhZ0/s1600/Comprobando+LED+emisor+de+infrarrojos.png

Finalmente tenemos el uso de los diodos Led blancos que a más de contar con un juego incluido en la misma cámara sería necesario hacer un propio juego con la cantidad de luz suficiente ya que el juego de la cámara podría no ser suficiente y debido a que se está limitando a la energía que es capaz de suministrar el Raspberry Pi por el puerto USB.

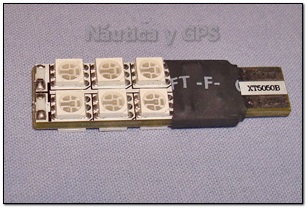


Figura 3‑10 Flash de led

**Fuente**: http://www.nauticaygps.com.ar/Tecnologia/estrobo/images/modulo\_led2.jpg

#### 3.6.3. Filtrado de la imagen

##### 3.6.3.1. Reducción de imagen

Al aplicar este tipo de filtro lo que se pretende es reducir el tamaño de la imagen en cuanto al número de pixeles que la conforman teniendo así una matriz de trabajo normalizada a un tamaño adecuado.

Por ello se ha determinado REDUCED\_SIZE a 480 para que sin importar el tamaño de la entrada se tenga un mismo tamaño para que no cambien los valores de las coordenadas de los ojos.

##### 3.6.3.2. Corte

Para poder aplicar en el reconocimiento facial es necesario hacer un recorte de la imagen ya reducida, para ello lo que se hace es identificar los ojos para hacer el recorte de los mismos junto con partes relevantes del rostro, hay que destacar que no toda la imagen se usa para el reconocimiento. A paso seguido del recorte se la alinea de forma horizontal.

Para esto se puede hacer de varias formas y una de ellas es la siguiente:

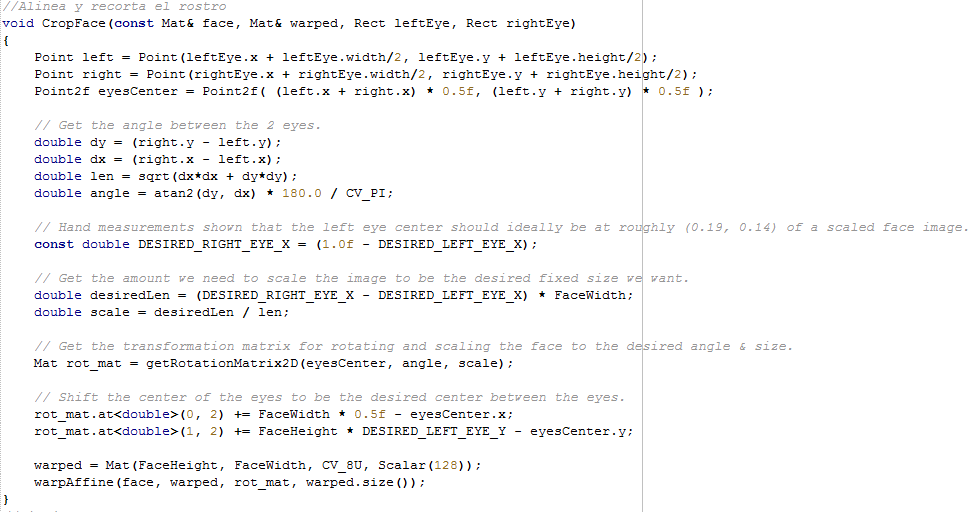


Figura 3‑11 Funciones en C++ para filtro de imágenes

**Fuente**: Autor

Es una función que por entrada tiene coordenadas previamente definidas y devuelve sub matriz recortada

##### 3.6.3.3. Escala de Grises

Es necesario poner en binario la imagen (tener 1 y 0 en la matriz de la imagen), para ello se aplica un filtro para convertir la imagen en RGB a escala de grises reduciendo así en este punto gran cantidad de información que no es relevante.

#### 3.6.4. Selección de algoritmo para reconocimiento facial

El algoritmo que está corriendo tiene un gran respaldo para su utilización en la parte de análisis estadístico de este trabajo, entrando en detalles del programa principal tiene una función que obtiene las coordenadas de los ojos y el rostro, devuelve true si las encuentra, función para alinear y recortar el rostro por la zona de interés, otra para resaltar y dibujar marcas del rostro encontrados, una función de inicialización que se encarga de abrir los algoritmos disponibles para el reconocimiento de ojos, rostros y reconocimiento facial, para en el bucle principal main hacer la respectiva apertura del puerto serial, filtros necesarios y llamadas a las funciones antes mencionadas junto con otra embarcándolas en un menú que cuenta con las opciones de entrenar el programa para luego dejarlo en modo reconocimiento y a la espera de una solicitud de respuesta por UART si hay compatibilidad de reconocimiento facial con el uso del algoritmo LBPHFaceRecognizer, librería disponible en el directorio de OpenCV.

#### 3.6.5. Breve descripción del algoritmo para reconocimiento facial

El algoritmo tiene una función que obtiene las coordenadas de los ojos y el rostro, devuelve true si las encuentra, función para alinear y recortar el rostro por la zona de interés, otra para resaltar y dibujar marcas del rostro encontrados, una función de inicialización que se encarga de abrir los algoritmos disponibles para el reconocimiento de ojos, rostros y reconocimiento facial, para en el bucle principal main hacer la respectiva apertura del puerto serial, filtros necesarios y llamadas a las funciones antes mencionadas junto con otra embarcándolas en un menú que cuenta con las opciones de entrenar el programa para luego dejarlo en modo reconocimiento y a al espera de una solicitud de respuesta por UART si hay compatibilidad de reconocimiento facial con el uso del algoritmo LBPHFaceRecognizer, librería disponible en el directorio de OpenCV.

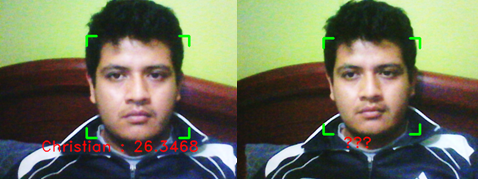


Figura 3‑12 Reconocimiento facial

**Fuente**: Autor

### 3.7. Método de programación del microcontrolador ARM

Existen tres métodos o tipos de programación más conocidos como lo son por sondeo, por interrupciones y por DMA.

#### 3.7.1. Programación por sondeo

Esta consiste en la programación de instrucciones de tal forma que la secuencia se la haga en un orden. Desde la primera hasta la última, método que complica en ciertas aplicaciones.

#### 3.7.2. Programación por interrupciones

Es una herramienta muy potente a la hora de programar ya que el micro controlador permite interrumpir la programación anterior para atender una sub rutina generada por una interrupción interna o externa. Teniendo la característica de que después de atender la interrupción puede regresar a donde se dejó el programa principal para continuar con su ejecución.

En ARM tiene una característica muy particular de un sistema Operativo que es el manejo de prioridades de las interrupciones el cual está disponible en un vector llamado NVIC.

#### 3.7.3. Programación por DMA

Otro recurso muy interesante es el manejo de los periféricos de DMA que quiere decir acceso directo a memoria, la característica principal de este periférico disponible es que puede hacer transferencias directamente entre un periférico como USART, ADC, etc a memoria o viceversa, a más de poder hacerlo de memoria a memoria sin que el programa principal sea interrumpido, se la puede combinar con interrupciones.

Para esta aplicación se ha incurrido en los tres métodos para optimizar los recursos que ofrece el micro controlador aplicando así:

***ADC con DMA***

*HAL\_ADC\_Start\_DMA(&hadc1,(uint32\_t\*)&valores,10);*

***Dos UART por interrupciones***

*if(HAL\_UART\_GetState(&huart4) == HAL\_UART\_STATE\_READY)*

*{*

*HAL\_UART\_Receive\_IT(&huart4, &RxByte, 1);*

*}*

*if(HAL\_UART\_GetState(&huart1) == HAL\_UART\_STATE\_READY)*

*{*

*HAL\_UART\_Receive\_IT(&huart1, &RxByte, 1);*

*}*

*while (1)*

*{}*

### 3.8. Control del sistema

El sistema tiene varias características similares a las convencionales que encontramos en los sistemas de seguridad comunes y corrientes, cuenta para ponerlo en dos modos: modo seguro y modo conducción, en cada uno de estos modos hay procesos o algoritmos que se pueden ejecutar solo en ese modo y así mismo otros que son de forma general en cualquier instancia como la geo localización, al arrancar el sistema se deberá seleccionar uno de los modos ya que de no ser así solo estará ejecutando los algoritmos generales, lo cual se lo hará con la aplicación.

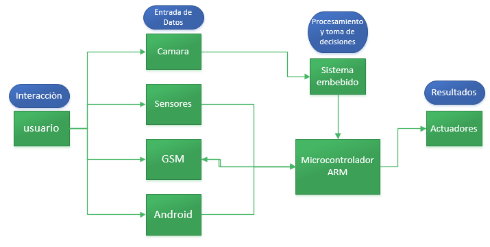


Figura 3‑13 Diagrama de bloques secuencial de procesos

**Fuente**: Autor

#### 3.8.1. Modo seguro

Las características principales de este modo son las de estar continuamente a la espera de alguna violación del sistema para proceder a alarmar con una sirena y con una llamada por la red GSM, además de hacer un aviso con la alarma si se acercan mucho al auto por la parte en la que se encuentre localizado el sensor.

#### 3.8.2. Modo conducción

En este modo tenemos las opciones de protocolo a seguir para poder usar el automóvil que son dos: la de autenticación por reconocimiento facial y autenticación por clave de súper usuario, estos serán los que permitirán dar arranque para encenderlo, cuenta con sensor de luz para poder determinar si es necesario adecuar el enfoque de la cámara y para el control de las luces o faros para conducir en la noche, además de incluir un sensor de alcohol que podría quitar de igual manera el arranque de acuerdo al nivel de alcohol en el ambiente pero es poco aplicable, cuenta con las opciones hacer mando sobre los motores de seguros eléctricos, hace un control automático de la temperatura ambiente.

### 3.9. Construcción del sistema

Una vez que se ha determinado y tenido una idea más clara se procedió a diseñar el sistema completo a partir de un diagrama general representado en la siguiente figura.

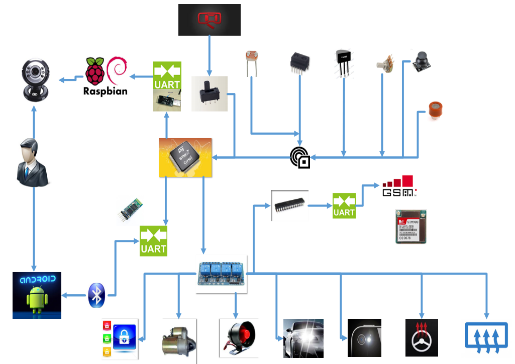


Figura 3‑14 Diagrama de bloques detallado del sistema

**Fuente**: Autor

#### 3.9.1. Estructura base

Se tomó la decisión de usar unas planchas de aluminio acopladas con estructuras del mismo material para separarlo una cierta distancia por la parte inferior a manera de patas.

#### 3.9.2. Etapa de control

Esta etapa está conformada principalmente por el micro controlador STM32F303VCT de arquitectura ARM incrustado en la tarjeta de desarrollo Discovery STM32F3 que es el encargado de la toma de decisiones programadas en los algoritmos insertados en su memoria de acuerdo a las condiciones y lecturas de los sensores que están conectados a sus respectivos pines, por el diseño de la tarjeta se procedió a realizar una placa adaptativa en el software Proteus que en la Figura 15-3 se puede apreciar, teniendo una alimentación principal de 7.5 Voltios a 2.5 Amperios y sobre la cual va montado la tarjeta Discovery, los módulos de comunicación Bluetooth HC-05 y USB a Serial CP2102, el controlador Atmega328p encargado de gobernar el módulo SIM908 e interactuar con la tarjeta Discovery, reguladores de voltaje a 5V y 3.3V y los conectores adecuados para conectar con las demás etapas.

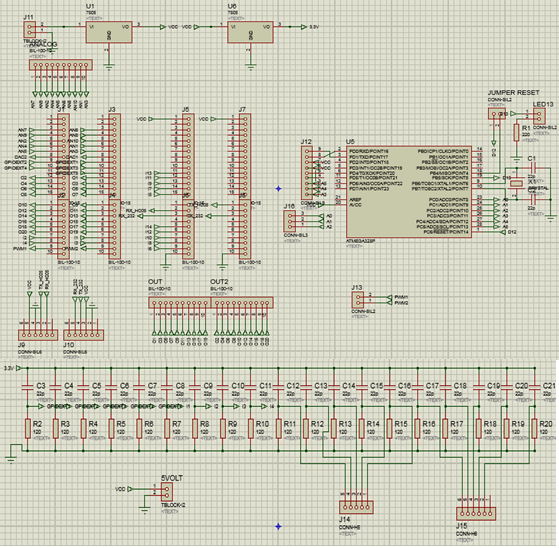


Figura 3‑15 Diagrama electrónico de la etapa de control

**Fuente**: Autor

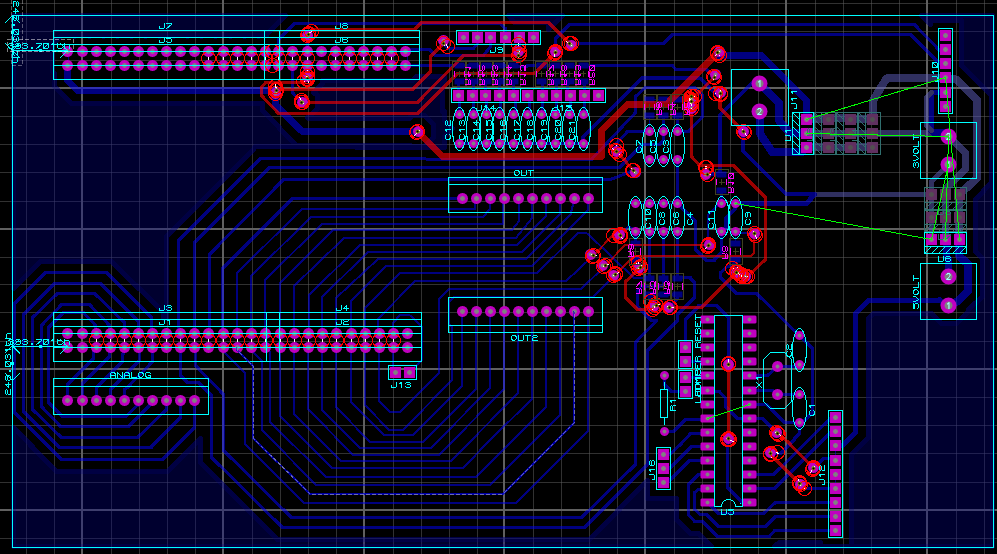


Figura 3‑16 PCB del diagrama electrónico de la etapa de control

**Fuente**: Autor

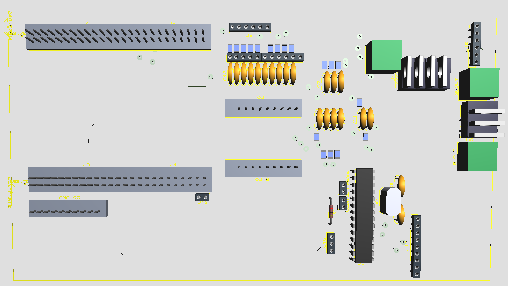


Figura 3‑17 Imagen 3D del PCB del diagrama de electrónico de la etapa de control

**Fuente**: Autor

#### 3.9.3. Etapa de adquisición de datos

Esta etapa cuenta con dos partes, la primera que es una placa que suministra la energía a los sensores que se puede observar en la Figura 18-3 y recoge las señales de los mismos para enviarlos al micro controlador por conexión directa por medio de cables y la segunda que son los sensores en sí, montados en placas individuales con sus conectores a manera de módulos de sensores, de igual manera el diseño de las placas fue en Proteus, además que en esta etapa entra el sistema embebido que está actuando como sensor que da la información al micro controlador.

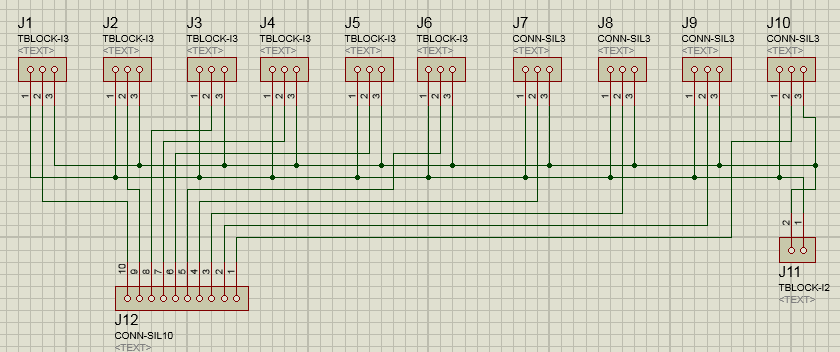


Figura 3‑18 Diagrama electrónico de la etapa de adquisición de datos

**Fuente**: Autor

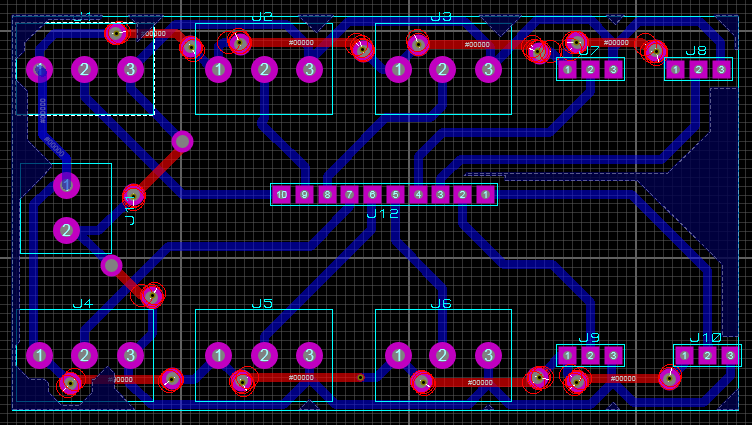


Figura 3‑19 PCB del diagrama electrónico de la etapa de adquisición de datos

**Fuente**: Autor

#### 3.9.4. Etapa de potencia

La etapa de potencia está conformada por los dos módulos de relés que se encargaran de suministrar energía a los actuadores cuando el micro controlador así se lo ordene, además que también contamos con un módulo Sabertooth antes mencionado para manejar la energía de dos actuadores, otra parte del sistema que podría estar relacionado con la potencia seria las respectivas conexiones a la red de energía eléctrica.

### 3.10. OpenCV sobre Raspberry PI 2 con Raspbian

Se optó por usar la versión de OpenCV 4.9 completa para instalarlo en el Raspberry para lo cual fue necesario usar una memoria de 16 GB.

Antes de instalar cualquier cosa se debe hacer un update y un upgrade del sistema operativo para que actualice las cabeceras de donde procederán las dependencias necesarias.

Las dependencias instaladas para la aplicación fueron:

* *libopencv-dev*
* *cmake*
* *pkg-config*
* *python-numpy*
* *libdc1394-22-dev*
* *libpng12-dev*
* *libjasper-dev*
* *libavformat-dev*
* *libxine-dev*
* *libgstreamer-plugins-base0.10-dev*
* *libtbb-dev*
* *libfaac-dev*
* *libopencore-amrnb-dev*
* *libtheora-dev*
* *libxvidcore-dev*
* *build-essential*
* *libgtk2.0-dev*
* *python-dev*
* *libdc1394-22*
* *libjpeg-dev*
* *libtiff4-dev*
* *libavcodec-dev*
* *libswscale-dev*
* *libgstreamer0.10-dev*
* *libv4l-dev*
* *libqt4-dev*
* *libmp3lame-dev*
* *libopencore-amrwb-dev*
* *libvorbis-dev*
* *v4l-utils*

Una vez instalado y descargado OpenCV4.9 se lo descomprime.

Dentro de la carpeta creada con la descompresión se crea una carpeta necesaria llamada built para compilar.

Luego se compila con cmake y make.

Dejando con este breve resumen listo a OpenCV para programar o correr ejemplos.

### 3.11. Funcionamiento y algoritmos del sistema

La idea principal del Proyecto es reducir la incidencia de atracos exitosos por parte de anti sociales reduciendo así los índices e indicadores delincuenciales de este tipo en nuestro país.

La Tarjeta Discovery STM32F3 se encarga de tomar decisiones, ver el estado de las puertas, capturar valores analógicos de la distancia, temperatura y luminosidad, avisar al Atmega 328p para que comunique las anomalías por GSM con el SIM908, sonar sirenas, encender flash de led, activar enfriadores y calentadores, recibir órdenes por Bluetooth y pulsadores. Además, el Atmega 328p tiene la capacidad de responder por SMS a la aplicación Android con la localización por GPS obtenida con el SIM908, esto se logra programando los algoritmos necesarios usando el lenguaje de programación C de forma estructurada.

La Raspberry Pi 2 se comparó con un mini computador Wandboard IMX6 Dual y por los requerimientos del software OpenCV, es la encargada del reconocimiento facial sobre el Sistema Operativo Raspbian, en OpenCV encontramos las librerías EigenFaces, FisherFaces y LBPH para el reconocimiento facial, que sirven para calcular índices de coincidencia entre un rostro y otro, de tal forma que pueda autorizar el arranque del vehículo.

El proceso para la toma de decisiones del sistema en el vehículo está dado por:

1. La aplicación Android cuenta con dos botones virtuales debidamente identificados para la interacción del usuario y del sistema, permitiendo con ellos enviar caracteres a la tarjeta STM32F3 la cual decidirá abrir o cerrar seguros, poner o quitar alarma. Otro botón permite apagar luces, tiene disponible un text box para Password propio de aplicaciones Android hace autenticación con clave de súper usuario que también informa si hay coincidencia de clave para casos de emergencia y permitir el arranque del vehículo por un cierto tiempo. La aplicación también receptará la localización por SMS que llegue al dispositivo mientras está ejecutando la aplicación en primer plano, que con acceso a internet se podrá vincular y ubicarla en GoogleMaps.
2. Al quitar la alarma, abrir los seguros y una vez dentro del vehículo se puede solicitar autenticación por reconocimiento facial mediante un pulsador en la tarjeta STM32F3 la cual por UART ordena a la Raspberry que capture rostros y analice si hay o no coincidencia para responder con un carácter si hay autenticación superada o no, a más de ello la tarjeta STM32F3 enciende un flash led si hace falta luz si se lo indica el sensor LDR.
3. Cuando esta sin alarma y listo el vehículo para conducir se puede hacer el control de temperatura, haciendo un control ON/OFF. En el caso que el vehículo ya se ha puesto la alarma la tarjeta STM32F3 acciona la alarma si se fuerza a abrir los pulsadores en las puertas, también ordena al Atmega 328p que haga una llamada perdida con el módulo SIM908, de esta manera podremos enviar un SMS al número del módulo SIM908 con una solicitud de localización para poder ubicarlo. Otra operación en este estado es que, si se reduce la distancia frontal del vehículo, este acciona una advertencia con la sirena.

Los algoritmos del sistema fueron representados en diagramas de flujo que a continuación se muestra.

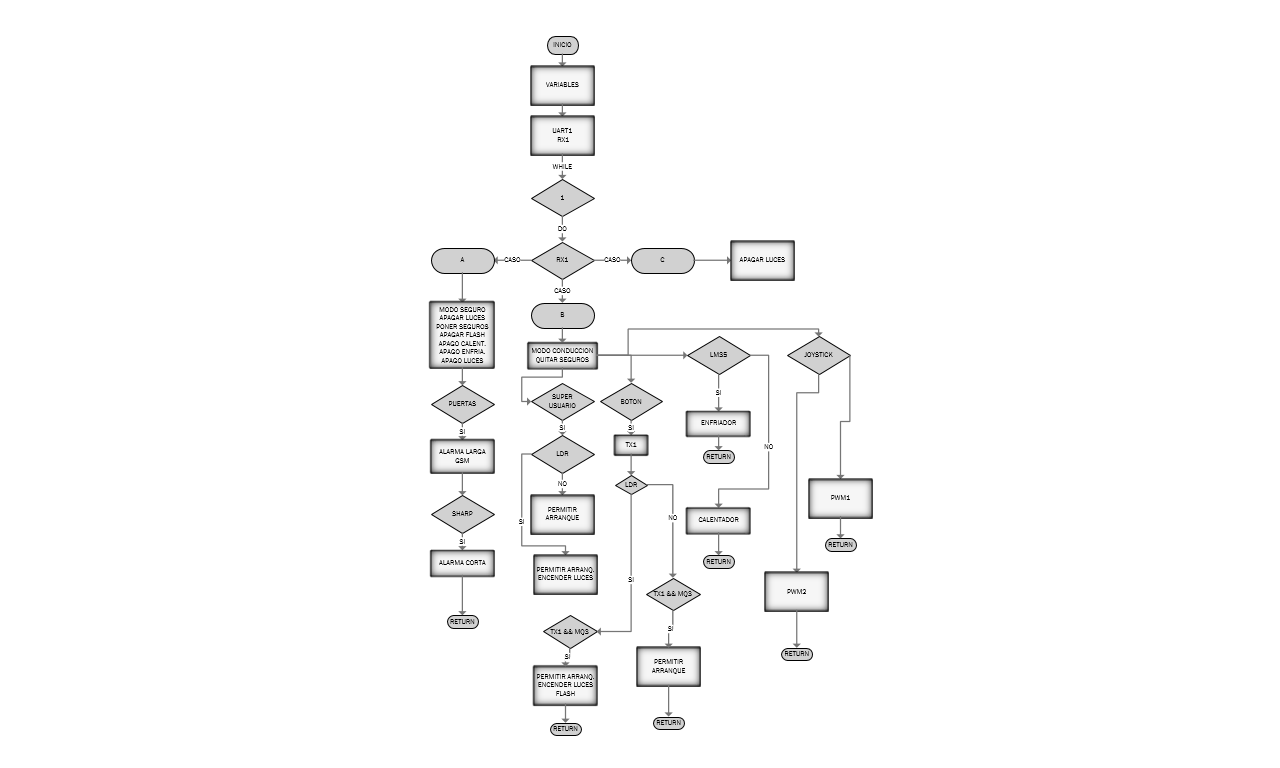


Figura 3‑20 Algoritmo de la tarjeta STM32F303

**Fuente**: Autor

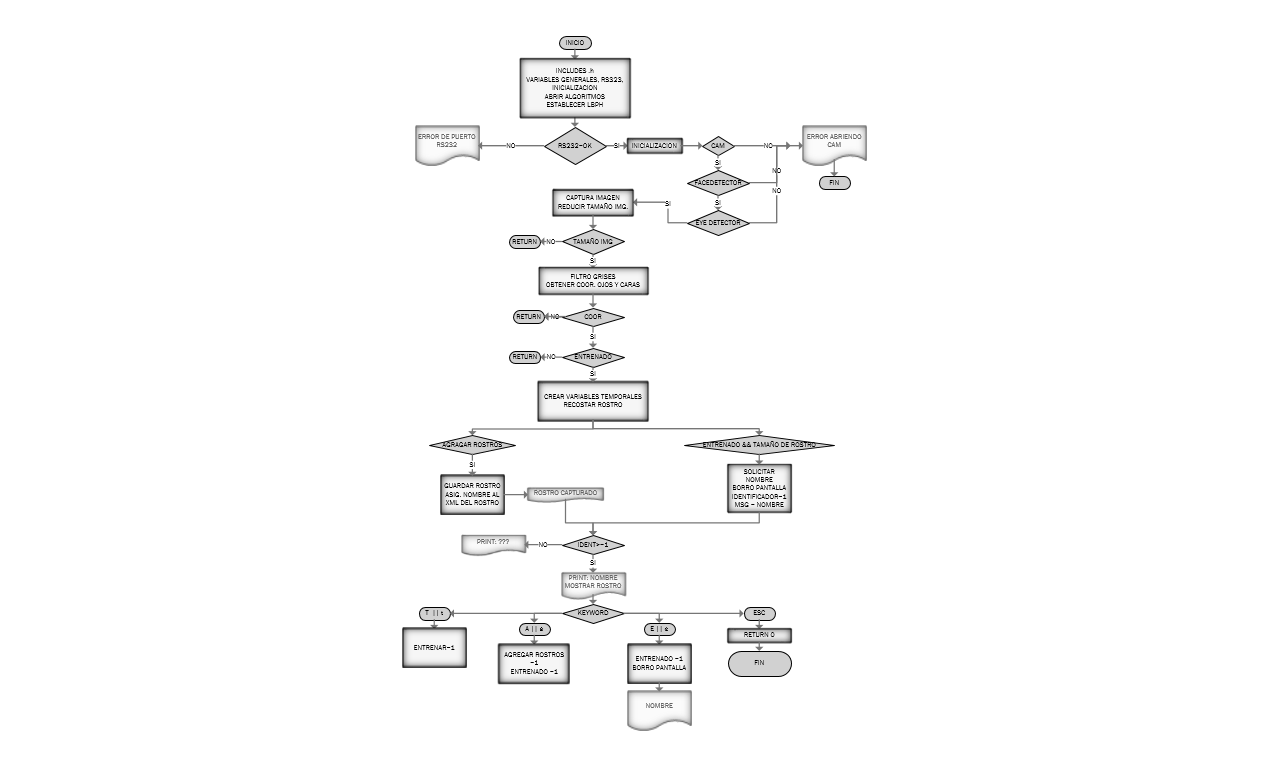


Figura 3‑21 Algoritmo del programa de reconocimiento facial

**Fuente**: Autor

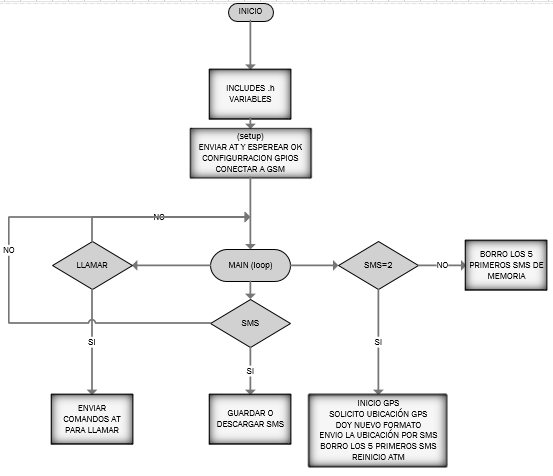
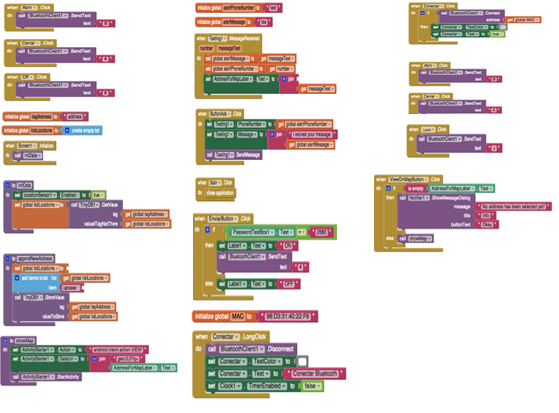


Figura 3‑22 Algoritmo del Atmega328 con SIM908

**Fuente**: Autor



**Figura 3‑23 Algoritmo de la aplicación de Android**

**Fuente**: Autor

# 

## 4. COSTOS, PRUEBAS Y RESULTADOS

Para la elaboración del prototipo demostrativo del sistema de seguridad se incurrió en gastos para la adquisición de módulos y componentes electrónicos, herramientas, servicios, etc.

### 4.1. Costo de los materiales

Tabla 4‑1 Lista de precios

|  |  |  |  |
| --- | --- | --- | --- |
| LISTA DE PRECIOS DEL PROTOTIPO DEL SISTEMA DE SEGURIDAD | | | |
| MATERIALES | CANT. | P.U. | TOTAL |
| Tarjeta STM32F303VCT Discovery F3 | 2 | 45,00 | 90,00 |
| Tarjeta STM32F407VCT Discovery F4 (Pruebas) | 2 | 50,00 | 100,00 |
| Raspberry Pi 2 tipo B (1 extra para imprevistos) | 3 | 95,00 | 285,00 |
| Kit de discipadores para Raspberry (1 extra para imprevistos) | 3 | 8,00 | 24,00 |
| Case para Raspberry Pi 2 (1 extra para imprevistos) | 3 | 15,00 | 45,00 |
| Tarjeta de Memoria micro SD 16 GB Clase 10 (Pruebas) | 4 | 20,00 | 80,00 |
| Cargador de 5 V a 2 A (1 extra para imprevistos) | 3 | 15,00 | 45,00 |
| Cargador de 7,5 V a 2,5 A | 1 | 22,00 | 22,00 |
| Hub USB | 2 | 10,00 | 20,00 |
| Cámara USB tipo webcam | 1 | 22,00 | 22,00 |
| Cable HDMI | 1 | 12,00 | 12,00 |
| Wandboard IMX6 (Pruebas) | 1 | 270,00 | 270,00 |
| Módulo de Relés a 5 V | 2 | 15,50 | 31,00 |
| Kit módulo SIM908 con antenas | 1 | 152,00 | 152,00 |
| Atmega328p | 1 | 8,00 | 8,00 |
| Reguladores a 3,3 V | 2 | 2,50 | 5,00 |
| Reguladores a 5 V | 4 | 0,80 | 3,20 |
| Modulo Bluetooth HC-05 | 1 | 17,00 | 17,00 |
| Modulo CP2102 USB to RS232 | 1 | 16,50 | 16,50 |
| Sensor LM35 | 1 | 3,50 | 3,50 |
| Sensor LDR | 2 | 0,50 | 1,00 |
| Sensor SHARP | 1 | 45,00 | 45,00 |
| Sensor de Alcohol Mq-3 | 1 | 20,00 | 20,00 |
| Joystick | 1 | 8,00 | 8,00 |
| Potenciómetros lineales | 3 | 1,50 | 4,50 |
| Leds Blancos SMD | 10 | 0,50 | 5,00 |
| Resistencias SMD | 30 | 0,05 | 1,50 |
| Módulo Sabertooth 5x2 | 1 | 70,00 | 70,00 |
| Zumbador a 6 V | 1 | 1,00 | 1,00 |
| Micro motor Pololu | 1 | 23,00 | 23,00 |
| Motor a 12 V 5 A | 2 | 45,00 | 90,00 |
| Estructura Base de aluminio | 1 | 35,00 | 35,00 |
| Switchs electrónicos | 10 | 0,25 | 2,50 |
| Gastos varios | 1 | 230,00 | 230,00 |
| TOTAL | | | 1787,70 |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

### 4.2. Pruebas realizadas

El algoritmo de reconocimiento facial se programó en C++ usando los tres métodos de reconocimiento facial individualmente para tomar datos del comportamiento del Raspberry mediante el uso de comandos bash para sistemas basados en Linux. Los comandos bash usados en el terminal son: df, top, free, vmstat, ps, cat /proc/meminfo. Los comandos anteriores arrojaron resultados de rendimiento. Las tablas de Baremo son el producto del promedio de 4 experimentos obtenidos al ejecutar el mismo algoritmo y técnica de reconocimiento facial, pero con diferente tipo de cámara.

### 4.3. Análisis estadístico

#### 4.3.1. Tipos de cámara

Se procedió a comparar los datos medidos para cada tipo de cámara en el sistema embebido:

Tabla 4‑2 : Datos recolectados de los tres tipos de cámara funcionando en el Sistema Embebido

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| WebCam | | | | | | |
|  | KiB Mem | Swap | Procesos | | Memoria | |
| Total | 948292 | 102396 | bash | 2638 | KernelStack | 1576 |
| Usado | 295936 | 0 | ps | 2653 | PageTables | 2248 |
| Libre | 652356 | 102396 |  |  |  |  |
|  |  |  |  | | | |
| Buffers | 20072 |  |
| Cache | 143672 |  |
| Cámara IP | | | | | | |
|  | KiB Mem | Swap | Procesos | | Memoria | |
| Total | 948292 | 102396 | bash | 2638 | KernelStack | 1576 |
| Usado | 310372 | 0 | ps | 2792 | PageTables | 2104 |
| Libre | 637920 | 102396 |  |  |  |  |
|  |  |  |  |  |  |  |
| Buffers | 21432 |  |  |  |  |  |
| Cache | 170448 |  |  |  |  |  |
| Cámara CSI | | | | | | |
|  | KiB Mem | Swap | Procesos | | Memoria | |
| Total | 884228 | 102396 | bash | 2721 | KernelStack | 1561 |
| Usado | 381164 | 0 | ps | 2722 | PageTables | 2257 |
| Libre | 503064 | 102396 |  |  |  |  |
|  |  |  |  |  |  |  |
| Buffers | 18444 |  |  |  |  |  |
| Cache | 204280 |  |  |  |  |  |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

Organizando estos datos en una tabla de Baremo se obtuvo:

Tabla 4‑3Tabla de Baremo de los datos de la Tabla 2-4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | CAM-IP | CAM-WEB | CAM-CSI |
| cpu | | 3 | 1 | 2 |
| memoria | | 310372 | 295936 | 381164 |
| cpu\_bash | | 2638 | 2638 | 2721 |
| cpu\_ps | | 2792 | 2722 | 2653 |
| Mem\_kernel | | 1576 | 1576 | 1561 |
| Mem\_paginación | | 2104 | 2248 | 2253 |
| Buffer | | 21432 | 20072 | 18444 |
| Caché | | 170448 | 143672 | 204280 |
| gastos extras | | 3 | 1 | 1 |
| latencias | | 3 | 2 | 1 |
| latencia focus | | 3 | 1 | 3 |
| ventaja\_conector | | 3 | 1 | 2 |
| Suma | | 511407 | 468887 | 613112 |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

Donde:

Para cpu,gastos extras, latencias, latencias focus y ventaja\_conector toma valores de significancia entre 1 y 3 que tratan de reflejar después de un análisis de las características de cada tipo de cámara las diferencias entre ellas.

Y Para los demás ítems como memoria, cpu\_bash, cpu\_ps, Mem\_kernel, Mem\_paginación, Buffer y caché son los valores tomados durante las pruebas.

Con los datos obtenidos al aplicar tres análisis de la t student de los tres tipos de cámaras en el software SPSS se obtuvo los datos:

Tabla 4‑4Resultados del análisis estadístico de Tabla 3-4 en software SPSS

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Diferencias relacionadas | | | | | t | gl | Sig. (bilateral) |
| Media | Desviación típ. | Error típ. de la media | 95% Intervalo de confianza para la diferencia | |
| Inferior | Superior |
| Par 1 | c1 - c2 | 6540,53846 | 13475,45782 | 3737,41955 | -1602,59920 | 14683,67612 | 1,750 | 12 | 0,106 |
| Par 2 | c2 - c3 | -22187,69231 | 45953,63297 | 12745,24461 | -49957,19479 | 5581,81017 | -1,741 | 12 | 0,107 |
| Par 3 | c3 - c1 | 15647,15385 | 33324,36646 | 9242,51631 | -4490,55926 | 35784,86695 | 1,693 | 12 | 0,116 |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

Dónde:

C1 es Cámara IP

C2 es Cámara Web USB

C3 es Cámara CSI

Se puede observar que los datos tienen una distribución normal a un nivel de confianza del 95% para la diferencia.

Al tener el valor de significancia podemos determinar los valores críticos α1, α2 y α3 para los 12 grados de libertad teniendo:

Valor critico α1=±1.782

t=1.750

Teniendo las siguientes hipótesis:

Ho= Hipótesis Nula

H1= Hipótesis Alternativa.

Ho(1): Las características significativas de la cámara c1 no es igual a las características significativas de la cámara c2.

En caso de que la hipótesis nula no se cumpla tenemos la hipótesis alternativa:

H1(1): Las características significativas de la cámara c1 es igual a las características significativas de la cámara c2.

Valor critico α2=±1.772

t=-1741

Lo que nos dice que se acepta la hipótesis nula de que las características significativas de la cámara c1 es igual a las características significativas de la cámara c2.

Ho(2): Las características significativas de la cámara c2 no es igual a las características significativas de la cámara c3.

En caso de que la hipótesis nula no se cumpla tenemos la hipótesis alternativa:

H1(2): Las características significativas de la cámara c2 es igual a las características significativas de la cámara c3.

Valor critico α3=±1.762

t=1.693

Lo que nos dice que se acepta la hipótesis nula de que las características significativas de la cámara c2 no son igual a las características significativas de la cámara c3.

Ho(3): Las características significativas de la cámara c1 no es igual a las características significativas de la cámara c3.

En caso de que la hipótesis nula no se cumpla tenemos la hipótesis alternativa:

H1(3): Las características significativas de la cámara c1 es igual a las características significativas de la cámara c3.

Lo que nos dice que se acepta la hipótesis nula de que las características significativas de la cámara c1 no son igual a las características significativas de la cámara c3.

Lo que además nos conlleva a deducir que hay diferencias significativas entre cada una de ellas y pudiendo escoger entre estas de acuerdo a la tabla de Baremo la de menor suma, debido a que menores requerimientos necesita, aunque vale recalcar que por análisis de percentiles solo se descarta la CAM IP dejando a elección las dos anteriores ya que sus diferencias son mínimas, por estas razones se procede a usar la cámara Web USB que en la gráfica se puede ver que tiene un 29%.

Figura 4‑1 Gráfico porcentual de las Sumas de la tabla de Baremo de tipo de cámaras

**Fuente**: Autor

#### 4.3.2. Métodos de reconocimiento facial

Así mismo los datos para los métodos de reconocimiento facial fueron:

Tabla 4‑5Datos recolectados de los tres métodos de reconocimiento facial funcionando en el Sistema Embebido

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| LBPH | | | | | | |
|  | KiB Mem | Swap | Procesos | | Memoria | |
| Total | 948292 | 102396 | bash | 2638 | KernelStack | 1576 |
| Usado | 295936 | 0 | ps | 2653 | PageTables | 2248 |
| Libre | 652356 | 102396 |  |  |  |  |
|  |  |  |  |  |  |  |
| Buffers | 20072 |  |  |  |  |  |
| Cache | 143672 |  |  |  |  |  |
| EIGENFACES | | | | | | |
|  | KiB Mem | Swap | Procesos | | Memoria | |
| Total | 948292 | 102396 | bash | 2638 | KernelStack | 1584 |
| Usado | 332368 | 0 | ps | 2979 | PageTables | 2376 |
| Libre | 615924 | 102396 |  |  |  |  |
|  |  |  |  |  |  |  |
| Buffers | 21760 |  |  |  |  |  |
| Cache | 170228 |  |  |  |  |  |
| FISHERFACES | | | | | | |
|  | KiB Mem | Swap | Procesos | | Memoria | |
| Total | 948292 | 102396 | bash | 2638 | KernelStack | 1560 |
| Usado | 326316 | 0 | ps | 2911 | PageTables | 2268 |
| Libre | 621976 | 102396 |  |  |  |  |
|  |  |  |  |  |  |  |
| Buffers | 21656 |  |  |  |  |  |
| Cache | 170084 |  |  |  |  |  |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

Organizando estos datos en una tabla de Baremo se obtuvo:

Tabla 4‑6Tabla de Baremo de los datos de la Tabla 5-4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | LBPH | EIGEN | FISHER |
| Cpu | | 1 | 3 | 2 |
| Memoria | | 295936 | 332368 | 326316 |
| cpu\_bash | | 2638 | 2638 | 2721 |
| cpu\_ps | | 2653 | 2979 | 2911 |
| Mem\_kernel | | 1576 | 1584 | 1560 |
| Mem\_paginación | | 2248 | 2376 | 2268 |
| Buffer | | 20072 | 21760 | 21656 |
| Caché | | 143672 | 170228 | 170084 |
| Latencias | | 2 | 1 | 3 |
| Suma | | 468798 | 533937 | 527521 |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

Dónde:

Para cpu y latencias toma valores de significancia entre 1 y 3 que tratan de reflejar después de un análisis de las características de cada método de reconocimiento facial.

Y Para los demás ítems como memoria, cpu\_bash, cpu\_ps, Mem\_kernel, Mem\_paginación, Buffer y caché son los valores tomados durante las pruebas.

Con los datos obtenidos al aplicar tres análisis de la t student de los métodos de reconocimiento facial en el software SPSS se obtuvo los datos:

Tabla 4‑7Resultados del análisis estadístico de Tabla 6-4 en software SPSS

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Diferencias relacionadas | | | | | T | gl | Sig. (bilateral) |
| Media | Desviación típ. | Error típ. de la media | 95% Intervalo de confianza para la diferencia | |
| Inferior | Superior |
| Par 1 | LBPH - EIGEN | -13027,8000 | 22561,85923 | 7134,68634 | -29167,58181 | 3111,98181 | -1,826 | 9 | 0,101 |
| Par 2 | LBPH - FISHER | -11744,6000 | 20254,37058 | 6404,99436 | -26233,70387 | 2744,50387 | -1,834 | 9 | 0,100 |
| Par 3 | EIGEN - FISHER | 1283,2000 | 2611,53526 | 825,83996 | -584,97978 | 3151,37978 | 1,554 | 9 | 0,155 |

**Realizado por:** Christian Salazar

**Fuentes:** Autor

Dónde:

LBPH es Método LBPH

EIGEN es Método por Eigenfaces

FISHER es Método por Fisherfaces

Se puede observar que los datos tienen una distribución normal a un nivel de confianza del 95% para la diferencia.

Al tener el valor de significancia podemos determinar los valores críticos α1, α2 y α3 para los 9 grados de libertad teniendo:

Valor critico α1=±2,262

t=-1,826

Teniendo las siguientes hipótesis:

Ho= Hipótesis Nula

H1= Hipótesis Alternativa.

Ho(1): Las características significativas del método por LBPH no es igual a las características significativas del método por Eigenfaces.

En caso de que la hipótesis nula no se cumpla tenemos la hipótesis alternativa:

H1(1): Las características significativas del método por LBPH es igual a las características significativas del método por Eigenfaces.

Valor critico α2=±2,262

t=-1,834

Lo que nos dice que se acepta la hipótesis nula de que las características significativas del método de LBPH no son igual a las características significativas del método por Eigenfaces.

Ho(2): Las características significativas del método por LBPH no es igual a las características significativas del método por Fisherfaces.

En caso de que la hipótesis nula no se cumpla tenemos la hipótesis alternativa:

H1(2): Las características significativas del método por LBPH es igual a las características significativas del método por Fisherfaces.

Valor critico α3=±2,033

t=1.554

Lo que se deduce que se acepta la hipótesis nula de que las características significativas del método por LBPH no son igual a las características significativas del método por Fisherfaces.

Ho(3): Las características significativas del método por Eigenfaces no es igual a las características significativas del método por Fisherfaces.

En caso de que la hipótesis nula no se cumpla tenemos la hipótesis alternativa:

H1(3): Las características significativas del método por Eigenfaces es igual a las características significativas del método por Fisherfaces.

Lo que se concluye que se acepta la hipótesis nula de que las características significativas del método por Eigenfaces no son igual a las características significativas del método por Fisherfaces.

Además, esto nos conlleva a deducir que hay diferencias significativas entre cada una de ellas y pudiendo escoger entre ellas de acuerdo a la tabla de Baremo el método de menor suma ya que es lo que menos requerimientos necesita, por estas razones se procede a usar el método de LBPH que como muestra en la gráfica de porcentajes tiene un 31%.

Figura 4‑2 Gráfico porcentual de las Sumas de la tabla de Baremo de tipo de métodos de reconocimiento facial

**Fuente**: Autor

### 4.4. Resultados

Analizando la Tabla I y usando el método de análisis estadístico de la t de student se determina la selección de la webcam ya que tiene una necesidad de un 76,47% de recursos en relación a la cámara con CSI que es la de mayor requerimiento. Reforzando esta decisión se toma en cuenta que para el empleo de la cámara IP se necesitaría montar una pequeña red LAN lo que implicaría la adquisición de hardware extra, mientras que la cámara de conector CSI tiene un bus de comunicación demasiado frágil para el medio de instalación y de difícil adquisición en el mercado.

El método LBPH con un 87,8% frente a EigenFaces de índice más alto, es el que menos recursos necesita dejando de lado a los dos métodos que usan procesos estadísticos y por esta razón arrojan mucha información. Adicionalmente podemos destacar que LBPH a más de no proporcionar mucha información, es muy tolerante a la variación de luz y frontalidad de las capturas, además el entrenamiento es menos riguroso ya que en las pruebas, en el mejor de los casos se necesitó solo una imagen como entrenamiento para que el algoritmo pueda hacer reconocimiento facial.

Al momento de instalar el primer prototipo del sistema en el vehículo, se tuvo grandes problemas de tamaño, por lo que se rediseño la estructura al 50% para montarlo de tal manera que no sea difícil el proceso.

La potencia de alimentación se dimensionó un 20% más de lo que el sistema requiere previniendo sobrecalentamiento de los dispositivos, además es necesario el uso de fuentes de alimentación aisladas para evitar que a la parte de control del circuito entre ruido por el accionamiento de actuadores de carga inductiva de grandes potencias como los motores de los seguros de las puertas y sirenas de alerta.

La interacción de Android con el sistema por medio de Bluetooth trabaja normalmente a una distancia menor o igual a 4 m debido a que el HC 05 que es el receptor del sistema tiene un rango de potencia de -4 a 6 dB.

## CONCLUSIONES.

* La aplicación del sistema no requiere una cámara de gran resolución como las que ofrecen las cámaras IP y las con conector CSI, debido a que esto afectaría la velocidad de llegada de la información al programa principal de reconocimiento facial y aumentar el consumo de recursos computacionales por lo que el uso de una cámara webcam USB brinda un aceptable desempeño en cuanto a este aspecto.
* La posible exposición del medio físico que conecte la cámara al sistema embebido pide como requisito una cierta robustez en ello por lo cual una cámara webcam USB tiene la más adecuada por sus características físicas y desempeño ya que una cámara con conector CSI tiene un conector muy conjunto de conductores muy frágil que podría verse afectado en algún momento, mientras que por otro lado una cámara IP su medio de conexión fue inalámbrico lo cual hubiese sido una buena opción pero debido a que no justificaría levantar una red para una sola función
* La técnica de LBPH tiene mayor tolerancia en cuanto al grado de frontalidad para la captura de la imagen a procesar, a más de tener un gran desempeño con variaciones significantes de iluminación.
* Un filtro muy necesario en cualquier método de reconocimiento facial es la transformación de la imagen RGB en una matriz binaria adecuada con el objetivo de reducir la información y por ende el costo computacional.
* La relación beneficio costo de una cámara con visión nocturna, la calidad presentación de la imagen de estas cámaras, la eficiencia de la iluminación led y la disponibilidad de un flash con esta tecnología hacen una excelente opción cuando exista ausencia de luz natural que dificulte el adecuado funcionamiento del sistema.
* El protocolo UART es el que menos líneas de comunicación requiere, a más de tolerar mayor distancia en relación con los SPI e I2C que necesitan una línea extra cada uno y además vale recalcar que en SPI por cada esclavo extra necesita una línea más.
* Raspberry al soportar la instalación de Raspbian que podría decirse que es un S.O. Debian optimizado que tiene mucho soporte y desarrollo continuo en todo el mundo que permite su uso con mucha estabilidad y este a su vez no presenta problemas para instalar OpenCV en su versión completa y sin mayores contratiempos, mientras que por el contrario al momento de hacer las pruebas con Wandboard con Ubuntu oficial de Canonical se presentan varios obstáculos con las dependencias disponibles.
* La alta prioridad que necesita la comunicación del Sistema embebido y el Smartphone con el micro controlador de 32 bits hacen necesaria la implementación del método de interrupciones sobre el de sondeo ya que también se cuenta con sub rutinas repetitivas.
* La implementación de las entradas analógicas con el periférico DMA permiten reducir el código del algoritmo permitiendo añadir varias funciones de control automático ya que habilitar la funcionalidad una sola vez al inicio en el micro controlador hacen que sea casi instantánea la disponibilidad de los datos en cualquier punto del programa principal.
* Con los recursos económicos necesarios, se puede añadir más y mejores prestaciones del prototipo diseñado en el proyecto en cuanto a seguridad y confort.

## 

## RECOMENDACIONES.

* Si se opta por el uso de una cámara IP se recomienda la creación de una VLAN para garantizar un ancho de banda constante para el tráfico de la información de interés para la aplicación.
* A pesar de que el algoritmo LBPH no presenta muchos problemas con la variación de iluminación en el ambiente es aconsejable hacer el entrenamiento con las variaciones de luz que podrían presentarse.
* Si se pretende usar una cámara con una resolución muy baja se aconseja usar las funciones de erosión y/o dilatación para poder mejorar los detalles de la imagen obtenida.
* El flash de led da muy buenos resultados, pero sería de mucha ayuda si se puede conseguir un flash de xenón que pude brindar una mejor iluminación en la ausencia de luz.
* Instalar el flash junto a la cámara para evitar que se produzcan sombras en el rostro que con los filtros aplicados no se detectaría las áreas de interés del rostro en la imagen.
* Para no tener complicaciones por la necesidad de interrupciones en el algoritmo de OpenCV se recomienda usar un tiempo fuera establecido por el usuario para el algoritmo de UART en el Raspberry para recuperar la ejecución del programa y este no se cuelgue esperando respuesta del periférico.
* Si se pretende hacer comunicación serial a grandes distancias se aconseja aislar estas comunicaciones y hacerlo a bajas velocidades para evitar problemas comunes como lo es la pérdida de información y de ser posible usar algún circuito integrado de apoyo como el necesario para comunicación por RS485.
* Instalar la librería completa de OpenCV para no tener problemas al momento de usar alguna de las funciones especiales disponibles, también es aconsejable en este punto hacerlo en una memoria micro SD de por lo menos 16 GB para no tener problemas de insuficiencia de memoria ya que en la ejecución de este proyecto fue insuficiente una de 8 GB.
* Debido a que los dos UART disponibles en el micro controlador STM32F303 comparten el mismo buffer de recepciones aconseja respaldar la información en la interrupción de los UART y no en las funciones generales del periférico USART.
* Se aconseja hacer retardos con timers con interrupción con ajuste de la prioridad para evitar usar los retardos del sistema ya que la aplicación requiere del uso de interrupciones y estas se podrían perder si la prioridad del retardo es mayor por defecto a menos que se lo aclare.
* Si se desea tener una lectura más aproximada a la real de la temperatura en el ambiente, se recomienda usar varios sensores y promediarlos para evitar tener magnitudes erróneas y poco acertadas.

## BIBLIOGRAFÍA.

**ALEMANIA,** ARM LIMITED**.** *ARM Architecture Reference Manual* (MANUAL DE REFENCIA), 2005.

https://www.scss.tcd.ie/~waldroj/3d1/arm\_arm.pdf

2015-09-22

**ARGENTINA,** VICNONI, José Roberto**,** *Bluetooth*, La Plata, 2007, pp. 4.

http://www.ing.unlp.edu.ar/electrotecnia/procesos/transparencia/Bluetooth.pdf

2015-09-18

**BENALCAZAR, F.** Comunicación UART de la PC con la Plataforma Interactiva Creada, Basada en la Tarjeta de Desarrollo AVR BUTTERFLY a Través del Puerto Serie RS232, Guayaquil – Ecuador: Frank Benalcázar, 2012, pp. 1-9.

https://www.dspace.espol.edu.ec/bitstream/123456789/20833/1/Frank%20Benalcazar\_Jorge%20Arevalo\_Paper.pdf

2015-09-22

**CANADÁ,** DENAULT, A.*Other Architectures (ARM, IA-64)*.2015, pp. 1-19.

<http://www.cs.mcgill.ca/~adenau/teaching/cs573/Other_Arch.pdf>

2015-08-13

**COLOMBIA,** ELECTRONICLAB, *Módulo Bluetooth HC-05 Serial Rs232*. Bogotá: Electronilab.

http://electronilab.co/tienda/modulo-bluetooth-hc-05-serial-rs232/

2015-06-15

**CONVERSOR DE USB A UART SILÍCEO USB 2.0 SERIAL RS232 UART TTL CP2102 COMPATIBLE PARA ARDUINO PRO MINI.**

<http://www.tienda.siliceo.es/usb-serie-modulo-rs232-uart-ttl-cable-puerto-com-chip-cp2102-p-34.html>

2015-06-15

**CHINA,** ALIEXPRESS**,** Fuente de alimentación ajustable XL6009 DC-DC BoostBuck convertidor Step Up / Down fuente de alimentación módulo regulador #090040.[http://www.aliexpress.com/item/Adjustable-Power-Supply-XL6009-DC-DC-Boost-Buck-Converter-Step-Up-Down-Power-Supply-Regulator-Module/1607822336.html](http://www.aliexpress.com/item/Adjustable-Power-Supply-XL6009-DC-DC-Boost-Buck-Converter-Step-Up-Down-Power-Supply-Regulator-Mo)

2015-06-16

**CHINA,** SIMCOM, *SIM908-C\_Hardware Design\_V1.01* (*MANUAL DE REFERENCIA*), Shanghai -: Shanghai SIMCom Wireless Solutions Ltd, Septiembre 2011.

http://www.sos.sk/a\_info/resource/c/SIM908-C.pdf

2015-09-22

**CHINA** ZHIGUANG, Y.*Demographic Classiﬁcation with Local Binary Patterns*, Beijing: Yang Zhiguang, 2007.

http://media.cs.tsinghua.edu.cn/~imagevision/papers/ICB07\_demographic.pdf

2015-08-14

**ESPAÑA,** ÙBEDA, B.*Sistemas embebidos*. Murcia, 2009.

<http://ocw.um.es/ingenierias/sistemas-embebidos/material-de-clase-1/ssee-t01.pdf>

2015-09-18

**GUILLÈN, J.** *Visión Artificial aplicada en Vehículos Autónomos Submarinos*. Madrid: Guillen Julio, 2011, pp. 2.

<http://jderobot.org/store/julio.guillen/visartap_guillen.pdf>

2015-08-18

**IC ATMEGA328**.

http://saber.patagoniatecnology.com/atmega-328-micro-microcontrolador-chip-uno-nano-pro-mini-tutorial-arduino-arduino-argentina-ptec/

2015-06-15

**MARTINEZ MARTINEZ, Rafael Eduardo de Jesús.** Dimensionamiento de las interfaces ABIS y ATER de la red GSM, (**TRABAJO TÉCNICO DE GRADO**) Universidad Católica Andrés Bello, Especialista en Telecomunicaciones, Caracas, Venezuela. 2011. pp. 31.

http://biblioteca2.ucab.edu.ve/anexos/biblioteca/marc/texto/AAS2653.pdf

2015-09-18

**MDK MICROCONTROLLER DEVELOPMENT KIT**.

<http://www2.keil.com/mdk5/>

2015-08-13

**MÓDULO SIM CON GSM/GPRS/GPS SIM908**.

http://www.todoelectronica.com/modulo-sim-con-gsmgprsgps-sim908-p-14950.html

2015-06-15

**OTTADO, G.** *Reconocimiento de caras: Eigenfaces y Fisherfaces*, Uruguay: Guillermo Ottado, 2010, pp. 1-15.

https://eva.fing.edu.uy/file.php/514/ARCHIVO/2010/TrabajosFinales2010/informe\_final\_ottado.pdf

2015-06-15

**REINO UNIDO,** CAS**.** *The Raspberry Pi Education Manual*.Computing at school, 2012. <http://pi.cs.man.ac.uk/download/Raspberry_Pi_Education_Manual.pdf>

2015-09-18

**RONDAHL, T.** *Face Detection in Digital Imagery Using Computer Vision and Image Processing* (**TÉSIS**), (Bachelor’s Thesis in Computing Science) UMEA University, Suecia. 2011. pp. 5.

http://www.diva-portal.org/smash/get/diva2:480900/FULLTEXT02

2015-09-20

**STM32 32-BIT ARM CORTEX MCUS.**<http://www.st.com/web/en/catalog/mmc/FM141/SC1169?sc=stm32>

2015-08-13

**STM32F3DISCOVERY.**

<http://www.st.com/web/catalog/tools/PF254044>

2015-06-15

**STM32CubeMX.**

http://www.st.com/web/catalog/tools/FM147/CL1794/SC961/SS1743/PF259242?sc=microxplorer

2015-08-13

**STM-STUDIO.**

http://www.st.com/web/en/catalog/tools/PF251373>

2015-08-13

**TEMPERATURE SENSOR** <http://www.facstaff.bucknell.edu/mastascu/eLessonsHTML/Sensors/TempLM35.html>

2015-06-15

**TUTOR DE PROGRAMACIÓN**.

http://acodigo.blogspot.com/p/tutorial-opencv.html

2015-06-08

**WANDBOARD**.

<http://archlinuxarm.org/platforms/armv7/freescale/wandboard>

2015-09-18