



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA DE INGENIERÍA DE SISTEMAS

**DISEÑO DE UN SISTEMA DE DETECCIÓN FACIAL UTILIZANDO
CÁMARAS IP PARA EL
RECONOCIMIENTO DE INDIVIDUOS EN LA CERCANÍA DE
RESIDENCIAS FAMILIARES**

Trabajo de titulación previo a la obtención del

Título de Ingenieros de Sistemas

AUTORES: CRISTHIAN VICENTE HEREDIA SALAZAR
DANIEL ALEXANDER REA RODRIGUEZ

TUTOR: MANUEL RAFAEL JAYA DUCHE

Quito – Ecuador

Julio del 2022

**CERTIFICADO DE RESPONSABILIDAD Y AUTORIA DEL TRABAJO
DE TITULACIÓN**

Nosotros: Cristhian Vicente Heredia Salazar, con documento de identificación N.º1724255417 y Daniel Alexander Rea Rodríguez, con documento de identificación N.º1751800606, manifestamos que:

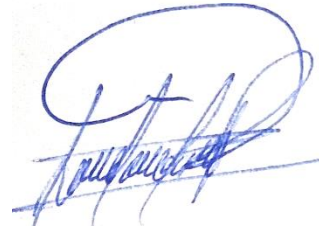
Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 20 de Julio del 2022

Atentamente



Cristhian Vicente Heredia Salazar
1724255417



Daniel Alexander Rea Rodriguez
1751800606

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL
TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA
SALESIANA**

Nosotros: Cristhian Vicente Heredia Salazar, con documento de identificación N.º1724255417 y Daniel Alexander Rea Rodríguez, con documento de identificación N.º1751800606, manifestamos que voluntariamente otorgamos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: “Diseño de un sistema de detección facial utilizando cámaras IP para el reconocimiento de individuos en la cercanía de residencias familiares”, el cual ha sido desarrollado para optar por el título de Ingenieros de Sistemas en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Quito, 20 de Julio del 2022

Atentamente,



Cristhian Vicente Heredia Salazar
1724255417



Daniel Alexander Rea Rodriguez
1751800606

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Manuel Rafael Jaya Duche con documento de identificación N.º 1710631035, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO DE UN SISTEMA DE DETECCIÓN FACIAL UTILIZANDO CÁMARAS IP PARA EL RECONOCIMIENTO DE INDIVIDUOS EN LA CERCANÍA DE RESIDENCIAS FAMILIARES, realizado por: Cristhian Vicente Heredia Salazar, con documento de identificación N.º1724255417 y Daniel Alexander Rea Rodríguez, con documento de identificación N.º1751800606, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 20 de Julio del 2022

Atentamente,



Ing. Manuel Rafael Jaya Duche, MSc
1710631035

INDICE GENERAL

Contenido	
INTRODUCCIÓN	1
Antecedentes.....	2
Problema.....	3
Justificación	4
Objetivos.....	5
Objetivo General:	5
Objetivos Específicos:	5
Metodología.....	5
Metodología Scrum	5
CAPITULO I	6
MARCO DE REFERENCIAS	6
1.1 Requerimientos	6
1.1.1 Funcionales	6
1.1.2 No funcionales	6
1.2. Diseño del prototipo	6
1.2.1 Diseño Lógico.....	6
1.3 Diseño de la solución.....	7
1.3.1 Definición y análisis requerimientos.	7
1.3.2 Análisis preliminar.	7
1.4 Arquitectura del módulo.	8
1.4.1 Diagrama caso de uso.	9

1.4.2 Arquitectura de base de datos.....	9
1.5 Diseño físico.....	9
1.5.1 Diseño Lógico.....	11
1.6 Procedimiento y fases de desarrollo.....	12
1.6.1 Captura de rostros.....	13
1.6.2 Entrenamiento de datos.....	15
1.6.3 Reconocimiento Facial.....	16
1.7 Precedentes.....	17
1.8 Implementación física del sistema.....	17
CAPITULO II.....	19
MARCO TEÓRICO.....	19
2. Fundamentación Teórica.....	19
2.1 Reconocimiento Facial.....	19
2.1.1 Detección de rostros.....	19
2.1.2 Etapas del reconocimiento facial.....	20
2.1.3 Alineamiento de la cara.....	20
2.1.4 Extracción de características.....	20
2.1.5 Reconocimiento de cara.....	20
2.1.6 Métodos para la detección facial.....	21
2.1.7 Método Eigenfaces.....	21
2.1.8 Método Fisherfaces.....	21
2.1.9 Librería OpenCV.....	22
2.1.10 Sistemas Embebidos.....	22
2.2 Software.....	23
2.2.1 Sistemas Operativo para Raspberry pi.....	23

2.3 Hardware.....	26
2.3.1 Entorno de la cámara	26
2.3.2 Luz.....	26
2.3.3 Ubicación.....	27
2.3.4 Modelos de Raspberry	27
CAPITULO III	32
METODOLOGIA.....	32
3.1 Análisis De Software Y Hardware.	32
3.1.1 Hardware	32
3.1.2 Software.....	33
3.2 Análisis económico del módulo.	35
3.2.1 Requerimientos para la funcionalidad del sistema	35
3.2.2 Costos de hardware y software del módulo.....	36
CAPITULO IV.....	38
PRUEBAS Y RESULTADOS.....	38
4.1 Análisis de pruebas	38
4.1.1 Pruebas de funcionalidad del reconocimiento.	38
4.1.2. Análisis de detección facial y reconocimiento en miembros del grupo hogar.	39
4.1.3 Pruebas de reconocimiento según el número de fotos.....	41
4.1.4 Pruebas de luminosidad.	43
4.1.5 Pruebas de variación de la distancia.	45
4.2. Pruebas de sensibilidad.....	48
4.3. Pruebas de confiabilidad.....	51
4.3.1 Pruebas de confiabilidad de monitoreo.	51

4.3.2 Pruebas de confiabilidad de notificación.....	53
4.4 Análisis de resultados	54
CONCLUSIONES	56
RECOMENDACIONES	57
LISTA DE REFERENCIAS	58

ÍNDICE DE FIGURAS

Figura 1: Diseño Lógico	7
Figura 2: Arquitectura del módulo	8
Figura 3: Diagrama de casos de uso	9
Figura 4: Diseño físico de la base de datos	11
Figura 5: Diseño lógico de la base de datos	12
Figura 6: Capturas de los rostros utilizando el método EigenFaces.....	14
Figura 7: Librería OPENCV	15
Figura 8: Proceso para captura y entrenamiento de rostros.....	16
Figura 9: Etapa de reconocimiento.....	17
Figura 10: Implementación física del sistema	18
Figura 11: Etapas del reconocimiento facial	20
Figura 12: Modelo Cliente – Servidor.....	26
Figura 13: Raspberry pi 1	27
Figura 14: Raspberry pi 2.	28
Figura 15: Raspberry pi 3.	29
Figura 16: Raspberry pi 4.	30
Figura17: Partes de la Raspberry pi 4.....	30
Figura 18: Reconocimiento facial de Ricardo Rea.....	39
Figura 19: Reconocimiento facial de Cristhian Heredia.	40
Figura 20: Reconocimiento facial de Silvia.	40

Figura 21: Detección del rostro más cercano.	41
Figura 22: Reconocimiento facial luz nocturna.....	43
Figura 23: Reconocimiento facial luz natural	44

ÍNDICE DE TABLAS

Tabla 1: Partes de la Raspberry Pi.....	31
Tabla 2: Tipos de cámaras.....	32
Tabla 3: Tipos de Raspberry pi.....	33
Tabla 4: Tipos de sistemas operativos en Raspberry pi 4.....	34
Tabla 5: Tipos de lenguaje de programación.....	35
Tabla 6: Costos de Hardware.....	36
Tabla 7: Costos de Software e instalación del módulo.....	36
Tabla 8: Costo total del proyecto.....	37
Tabla 9: Resultados de reconocimiento por el número de fotos.....	42
Tabla 10: Pruebas de funcionalidad de luminosidad.....	44
Tabla 11: Promedios de porcentajes de las pruebas de funcionalidad de los usuarios de reconocimiento por luminosida.....	45
Tabla 12: Pruebas de reconocimiento facial con diferentes distancias.....	46
Tabla 13: Promedios de porcentajes de las pruebas de funcionalidad de los usuarios con variación de distancias.....	47
Tabla 14: Interpretación de resultados de pruebas de sensibilidad.....	48
Tabla 15: Porcentajes totales de luminosidad del grupo familiar.....	50
Tabla 16: Porcentajes totales de distancias del grupo familiar.....	50
Tabla 17: Resultados finales de sensibilidad.....	51
Tabla 18: Pruebas de confiabilidad de monitoreo.....	52

Tabla 19: Pruebas de confiabilidad de notificación.....	54
---	----

Resumen

El presente proyecto fue diseñado para desarrollar un prototipo de un sistema de detección facial para el reconocimiento de los individuos en las cercanías de residencias familiares, debido a que la seguridad en los hogares se torna como un tema importante a tener en cuenta.

Este diseño fue desarrollado tomando en cuenta la seguridad en los hogares utilizando metodologías ágiles de desarrollo de software, además de dispositivos hardware. El módulo requiere principalmente una cámara que se encuentre ubicada en la parte exterior del domicilio, para la visualización de personas que estén rondando en las cercanías de la residencia familiar, para lo cual, se conectará a una Raspberry Pi 4, que alojará el sistema de detección facial.

En las pruebas de extracción de características faciales se comprobó que el sistema necesita mínimo 600 fotos para mejorar el entrenamiento del sistema, obteniendo un 91.1% de registros reconocidos, comprobando la mejora en la precisión del reconocimiento en los miembros del grupo hogar, con estas pruebas se evaluó al sistema en diferentes escenarios tales como: luminosidad, distancias y sensibilidad, así se comprobó el comportamiento del sistema al momento del reconocimiento facial en el exterior del domicilio.

Abstract

In this project, a prototype of a facial detection system was developed for the recognition of individuals in the vicinity of family residences, because home security becomes an important issue to consider.

This design was developed with home security in mind using agile software development methodologies, in addition to hardware devices. The module mainly requires a camera that is located outside the home, to view people who are hanging around the family residence, for which it will be connected to a Raspberry Pi 4, which will host the face detection.

In the facial feature extraction tests, it was found that the system needs a minimum of 600 photos to improve the training of the system, obtaining 91.1% of recognized records, verifying the improvement in recognition accuracy in the members of the home group with these tests the system was evaluated in different scenarios such as: luminosity, distances and sensitivity, thus the behavior of the system at the time of facial recognition outside the home was verified.

INTRODUCCIÓN

En la actualidad para mejorar la seguridad se toma en cuenta el uso de herramientas tecnológicas, puesto que se relaciona con factores hardware y software que proveen aportes significativos al usuario, como la facilidad de reconocer a un individuo a través de un dispositivo electrónico. La disponibilidad de sistemas de vigilancia permite usar de manera fácil los sistemas de reconocimiento facial, a través de dispositivos como cámaras que permiten la detección de individuos sospechosos. Es por esta razón que la seguridad y la tecnología crean una alianza directa (Howse.J, 2013).

En los sistemas de reconocimiento facial se han desarrollado componentes software a través de lenguajes de programación utilizando inteligencia artificial, debido a que existen librerías y algoritmos pre-entrenados y de código abierto. Años atrás, en la seguridad, no era tomado en cuenta el software, esto por la inexistencia y la falta de investigación de algoritmos de inteligencia artificial relacionados con la detección y reconocimiento facial. Actualmente debido a la tecnología, los sistemas de seguridad se han visto mejorados debido al desarrollo de nuevas tecnologías que ayudan a incrementar la seguridad de las personas (Jiménez. J.G, 1999).

El reconocimiento facial se puede aplicar en diferentes ámbitos como el monitoreo de hogares, controles de acceso a oficinas, claves de acceso telefónico, banca móvil, desbloqueo de dispositivos electrónicos, entre otros.

De todo esto, la importancia de implementar un sistema de video vigilancia que realice el reconocimiento facial de personas en los exteriores del hogar permita diferenciar que persona pertenece o no al grupo familiar, brindando una mayor sensación de seguridad a los usuarios que habitan en el hogar.

Antecedentes

En la actualidad hay un desarrollo en el campo tecnológico, donde se puede incorporar sistemas más sofisticados. Con la aparición de nuevas tecnologías han permitido adaptar la seguridad en los hogares añadiendo dispositivos hardware y software para poder garantizar seguridad y comodidad a los usuarios que se encuentran habitando dentro de sus viviendas.

Se diseña un sistema basado en el lenguaje de Python que contiene librerías para detección y reconocimiento facial, mismas que pueden ser utilizadas para dar solución a una serie de problemáticas.

El sistema necesita datos de entrada (videos de cada miembro del grupo familiar) para realizar el entrenamiento respectivo donde posteriormente se generará un archivo XML, mismo que contiene las características faciales de cada miembro del grupo familiar y servirá como base para realizar el reconocimiento.

Problema

La problemática que se ha encontrado dentro de los hogares de muchas personas se basa en la inseguridad y aumento de robos a casas, lo que se convierte en una preocupación entre los ecuatorianos.

En Ecuador, en el periodo 2020-2021, la fiscalía general del Estado publicó un informe en el que muestra un incremento de 2730 a 3187 robos a domicilios, con una variación del 16,74%.

Las provincias más afectadas en los meses de enero-octubre del 2021 han sido Guayas con 23 %, seguido de Pichincha con el 20,5% y Manabí con el 6,5% siendo estas las provincias más críticas como víctimas de robos a domicilios.

De acuerdo a la Fiscalía General del Estado, el número de denuncias en el período de enero—octubre del 2021 en la ciudad de Quito, aumentó un 22% al contrario del año 2020 que fueron de 3711 denuncias ya que en este año estuvo marcado por las restricciones debido a la pandemia del COVID 19, por ello hubo un descenso de robos en ese año, por ello en el año 2021 las cifras aumentaron a 4550 en el distrito metropolitano de Quito.

Este panorama muestra una visión a la ciudadanía a demandar posibles soluciones para que la situación mejore, por lo que es importante considerar aplicativos basados en inteligencia artificial que ayuden a solventar de cierta manera la seguridad en los hogares.

Justificación

El presente proyecto se basa en diseñar un sistema de video vigilancia utilizando cámaras IP para el reconocimiento de individuos en las cercanías de residencias familiares, para el monitoreo en los exteriores de los hogares, ya que actualmente se han incrementado los robos domiciliarios, por lo tanto, el propósito principal de este diseño permitirá al propietario del domicilio incrementar su seguridad y tener la facilidad de monitorear qué personas se encuentran en los exteriores de su hogar, aportando de cierta forma la seguridad de los miembros del grupo familiar.

Para este proyecto se utilizarán librerías open source de inteligencia artificial para la detección y reconocimiento de los miembros que pertenecen al domicilio, con el fin de incrementar la seguridad en los hogares, utilizando un dispositivo Raspberry PI y una cámara IP. Para el envío de notificaciones se utilizará correos electrónicos enviados a una persona del grupo hogar cuando exista personas desconocidas (sospechosas) en la cercanía de su residencia.

Objetivos

Objetivo General:

- Diseñar un prototipo de un sistema de reconocimiento facial para control de ingreso a domicilios.

Objetivos Específicos:

- Investigar los estados del arte para comprender la operacionalidad de las cámaras de seguridad y detección facial.
- Realizar el análisis de Hardware y Software para conocer el funcionamiento de los módulos.
- Utilizar la metodología Scrum en el módulo de inteligencia artificial necesario para el manejo de la aplicación.
- Evaluar el módulo de reconocimiento facial en un ambiente de pruebas.

Metodología

Metodología Scrum

La metodología Scrum es la escogida para este proyecto, debido a la sencillez de uso y la preparación que tiene para alcanzar los tiempos establecidos. Es una metodología veloz y adaptable para proyectos que poseen un alto grado de incertidumbre.

Scrum permitirá utilizar equipos de trabajo donde los desarrolladores desempeñan roles específicos para poder completar las tareas asignadas de manera correcta, para lo cual establecemos al Ingeniero Rafael Jaya como nuestro Scrum Master quien orientará y ayudará en cada etapa del proyecto. Los desarrolladores serán Daniel Alexander Rea Rodríguez Y Cristian Vicente Heredia Salazar, encargados del diseño de la aplicación.

Debido a la metodología que se llevará a cabo se realizarán reuniones constantes tanto con el Scrum Máster, los desarrolladores y el product owner (Dueño de la residencia), evitando retrasos en los avances y coordinando las diferentes etapas con los requerimientos incluidos por cada uno de los involucrados.

CAPITULO I

MARCO DE REFERENCIAS

1.1 Requerimientos

1.1.1 Funcionales

- El sistema debe detectar y reconocer al grupo de personas que conforman el hogar.
- El sistema debe notificar mediante correo al instante que reconoce un rostro desconocido.
- El sistema debe tener mínimo 600 fotos como muestra para tener una mejor precisión al momento del reconocimiento.

1.1.2 No funcionales

- El sistema deberá tener la disponibilidad de los 365 días en el año.
- El sistema no enviara falsos positivos ni falsos negativos al momento de reconocer.
- El sistema tendrá un porcentaje de exactitud al reconocer en las noches.

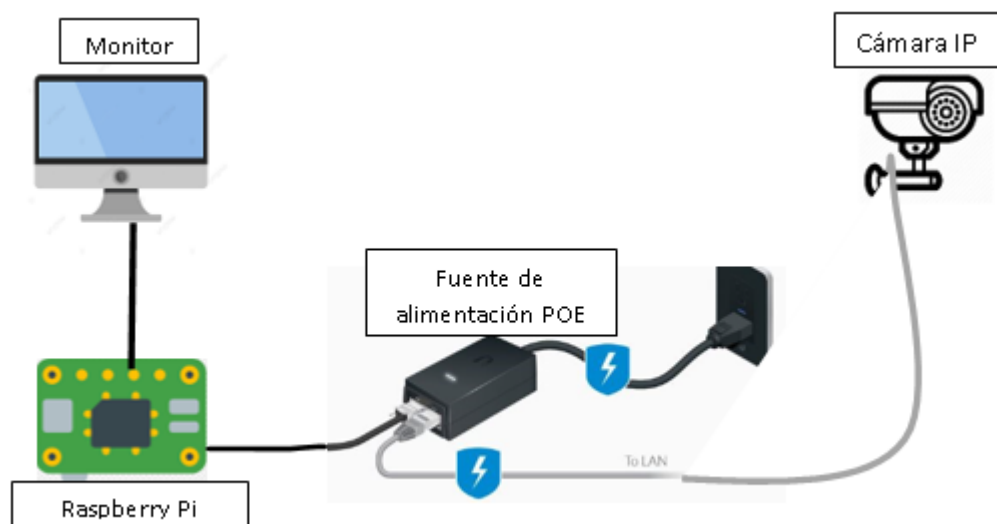
1.2. Diseño del prototipo

Este capítulo engloba el diseño lógico y físico de las herramientas de hardware y Software que se van a ocupar, mostrando los diferentes diagramas, bases de datos, librerías y códigos que van en el plan del proyecto.

1.2.1 Diseño Lógico.

Para el diseño de conexiones del prototipo se procede a conectar un monitor con el fin de monitorear todas las capturas de imágenes que la cámara muestre, así demostrando todos los dispositivos que se enlazo a la Raspberry-PI para su funcionamiento, tal como se indica en la figura 1 del diseño lógico, que muestra la forma de implementación del sistema.

Figura 1
Diseño Lógico



*Nota. Diseño lógico del módulo.
Elaborado por: Daniel Rea y Cristhian Heredia.*

1.3 Diseño de la solución.

1.3.1 Definición y análisis requerimientos.

Al hablar de módulo se conoce que es una parte electrónica que puede operar independientemente. Según Gardey y Perez (2009) cada módulo es parte de un sistema que se conecta con sus componentes. De esta manera, en el hogar objetivo se implementará un microordenador conocido como Raspberry Pi 4, acompañado de una cámara donde se capturan las imágenes para enviar la información al minicomputador y procesar los datos recibidos.

1.3.2 Análisis preliminar.

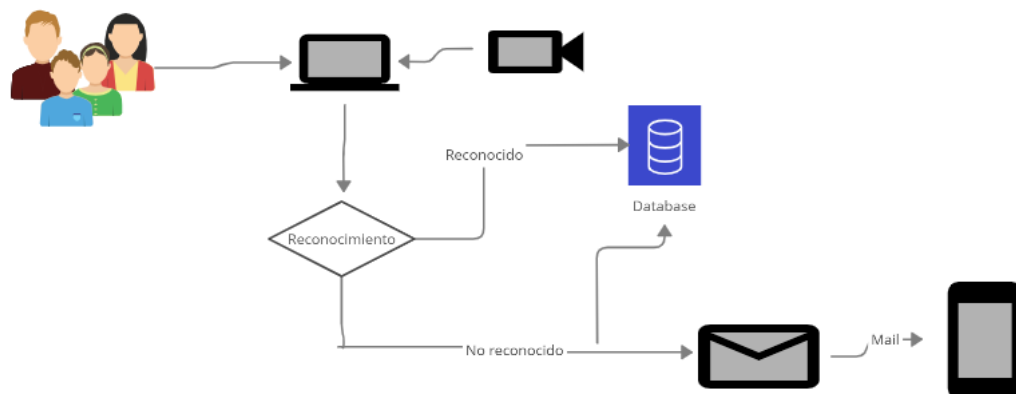
Para el análisis preliminar, se comenzó con la verificación de la estructura física del hogar, debido a que se debe examinar el lugar en el que se va a colocar el miniordenador, la cámara y su posición, puesto que debe tener una buena iluminación y enfoque en el exterior del domicilio para tener mayor facilidad al momento de identificar y detectar a la persona que se encuentra fuera de la vivienda. Se ha decidido utilizar el lenguaje de PHP porque hay librerías

asociadas con Python. La que se usará para enviar los mensajes a través de correo será smtplib, ya que Python fija un objeto de sesión de cliente SMTP que es utilizado para remitir mensajes de correo a cualquier máquina de internet.

1.4 Arquitectura del módulo.

La arquitectura del módulo se compone de dos elementos principales para que pueda funcionar de manera correcta, estos deben ser colocados en sus respectivos lugares, el minicomputador Raspberry Pi 4 gestor que realiza la detección a través del código Python con sus librerías, su base de datos correspondiente, junto a este va conectado la cámara IP que se encarga de capturar las imágenes de las personas que se ubiquen en el exterior del domicilio.

Figura 2
Arquitectura del módulo.



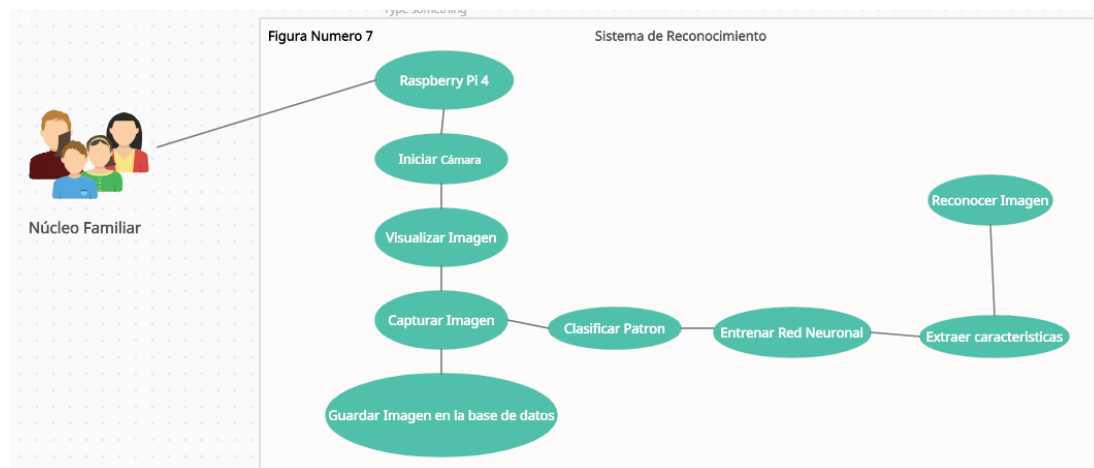
Nota: Arquitectura lógica del módulo.
Elaborado por: Daniel Rea y Cristian Heredia.

1.4.1 Diagrama caso de uso.

Este diagrama muestra el patrón por el que está conformado, y consecuentemente realizar el análisis y la detección del rostro, sea este conocido o desconocido.

Figura 3

Diagrama de casos de uso.



Nota. Diagrama de casos de uso para núcleo familiar.

Elaborado por: Daniel Rea y Cristian Heredia.

1.4.2 Arquitectura de base de datos.

Se ha tomado las siguientes especificaciones para que el usuario del grupo familiar pueda interactuar con los datos de manera intuitiva.

Ingreso en la tabla de Cliente (C.I, Nombres, Apellidos y Correo).

1. Verificar los datos almacenados en las tablas del rostro de personas conocidas y desconocidas.
2. Visualización de tablas Cliente, Reconocimiento, Desconocimiento.
3. Alerta sobre la detección y verificación de la persona en la base de datos.

1.5 Diseño físico

En el diseño físico de la base de datos se utilizó MySQL, donde se crearon tres tablas, la principal llamada Cliente que consta de los campos:

- **Id_Cliente:** “Primary Key” campo donde se almacenará el número de Cedula.

- **Nombre:** En este campo se guardará el nombre del cliente o grupo familiar.
- **Apellido:** Como se mencionó en el ítem de “Nombre” se almacenará el apellido.
- **Correo:** Este campo es considerado como uno de los principales porque registrará los correos de las personas del núcleo familiar para que al momento en que el sistema capture el rostro de la persona, sea conocido o desconocido, se puedan visualizar en la bandeja de entrada estos datos.

La segunda tabla se llama “Reconocimiento” que tiene como campos:

- **Id_Autoincrementable:** Este es un identificador que automáticamente incrementará el Id de tipo numérico, para que este se guarde dentro de la base al momento de reconocer a la persona.
- **Status:** En este campo se mostrará un estado de “OK” al momento de capturar la imagen siendo conocido.
- **N_Veces_Reconocido:** Aquí se va a guardar como dato numérico las veces en que se reconoció a la persona.
- **Fecha:** En este campo se almacenará la fecha y hora en tiempo real al momento de tomar la imagen.
- **Id_Cliente:** Este campo va a ser “Foreign key”, conocido como llave secundaria ya que está relacionada con la tabla de Cliente.

La tercera tabla llamada “Desconocimiento” contiene los siguientes campos:

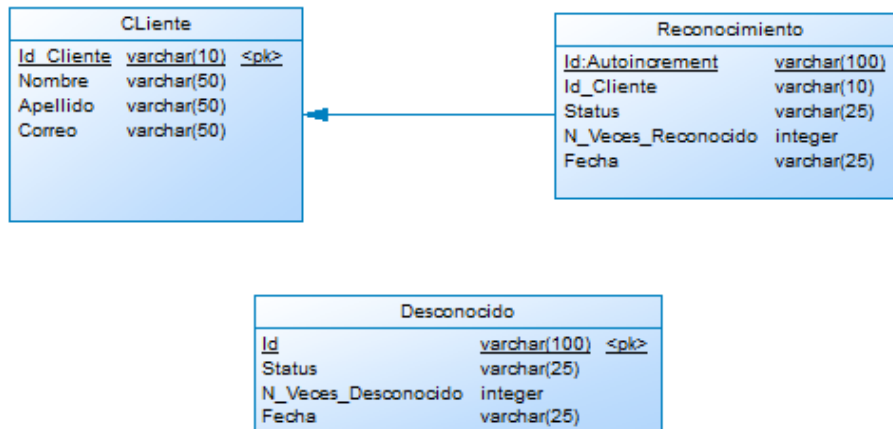
- **ID:** Campo auto incrementable de tipo numérico que se guardara en la base en el instante que la persona sea desconocida.
- **Status:** Este campo se mostrará un estado de “OK” en el instante de capturar la imagen y este sea de no conocido.

- **N_Veces_Desconocido:** Campo de tipo entero que almacenara las veces en la que no se reconozca a la persona.

- **Fecha:** Campo donde se guardará fecha y hora en tiempo real.

Figura 4

Diseño físico de la base de datos.



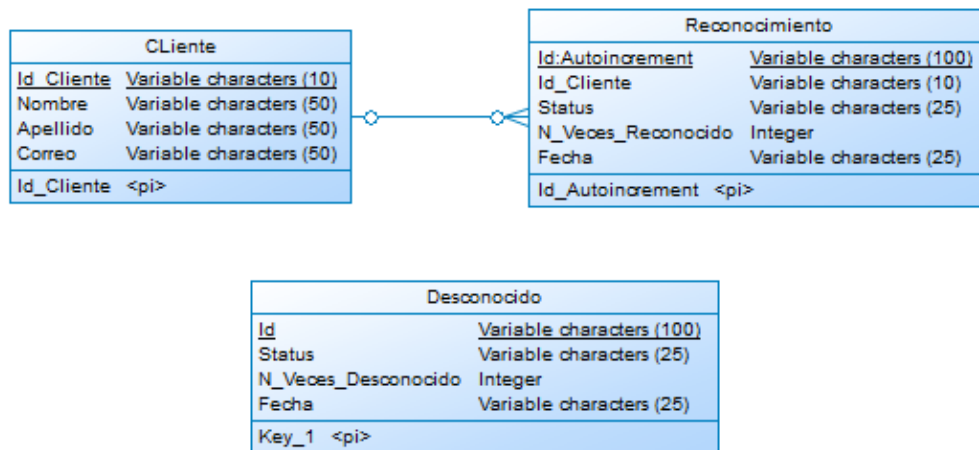
Nota. Diseño físico implementado en la base de datos.

Elaborado por: Daniel Rea y Cristian Heredia.

1.5.1 Diseño Lógico

Para este tipo de diseño se ha propuesto exponer las siguientes tablas demostradas en la Figura 5, especificadas acorde para el cliente, la forma de visualizar los datos de manera accesible.

Figura 5
Diseño lógico de la base de datos.



*Nota. Diseño lógico implementado en la base de datos.
 Elaborado por: Daniel Rea y Cristian Heredia.*

1.6 Procedimiento y fases de desarrollo.

En este ítem se describe de manera breve las fases de desarrollo que va a adquirir el módulo para la detección y reconocimiento facial

- Observación de las librerías específicas de Python que son de código abierto y las herramientas que se pueden usar.
- Instalación del Software (Python).
- Se va a indagar las librerías que son necesarias para el proceso de detección y reconocimiento facial, se ha elegido “OpenCV”, ya que es una librería de código abierto que facilita el procedimiento para captar rostros, y su entrenamiento para el reconocimiento.
- También se tendrá que ingresar a varios directorios que se crean al momento de instalar las librerías, puesto que hay archivos donde se almacenan los códigos para el entrenamiento y reconocimiento.
- En la base de datos se va a usar MySQL mencionada anteriormente. Para conectar con Python se debe instalar la librería “pymysql”.

- Se procede a la instalación de las librerías, en muchas ocasiones, se ha verificado que existen errores entre las versiones de las librerías con el lenguaje de Python. Para no causar ningún inconveniente se ha elegido utilizar la versión más actual de Python que es la V3.
- Se ha considerado algunos aspectos importantes en el desarrollo del módulo y estos son:

1.6.1 Captura de rostros.

En esta etapa de captura de rostros se ha usado la biblioteca de OpenCV porque abarca más de 2500 de algoritmos, entre todos estos se ha empleado los paquetes “imutils” ya que contiene un grupo de funciones para el procesamiento de imágenes. Otro modulo a implementarse es el “os” debido a que facilita y expone los detalles y la funcionalidad del sistema operativo.

Para la información de entrada se va a llevar a cabo la captura del rostro mediante un video, el que contiene una duración de 2 minutos, dado que en el entrenamiento de la red neuronal es importante tener una gran cantidad de imágenes que van a ser extraídas del video.

El inicio para las capturas de las imágenes es el ingreso a los respectivos directorios ubicados en determinadas rutas, donde se alojará el video de la persona para posteriormente subirlo al programa como indica la Figura 6.

Al momento de cargar el video, se procede a crear una carpeta de manera automática donde se va a guardar las imágenes extraídas.

Figura 6

Capturas de los rostros utilizando el método EigenFaces

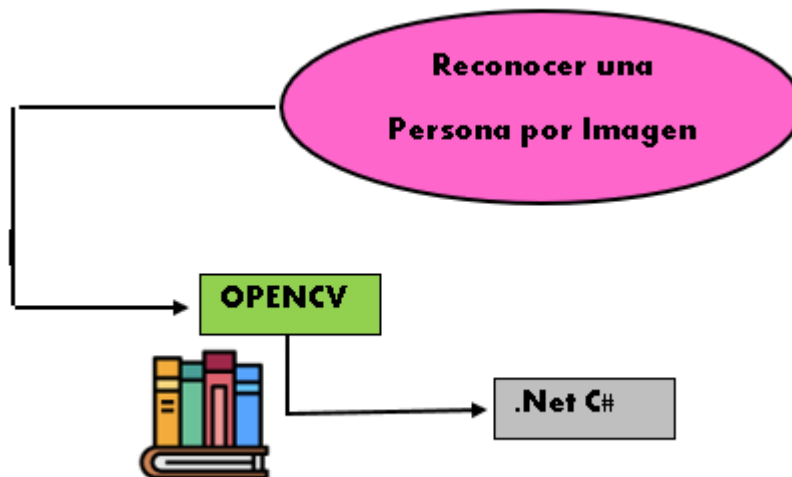


Nota. Captura y almacenamiento de rostros.

Elaborado por: Daniel Rea y Cristian Heredia.

En la biblioteca de OpenCV existe una librería “CascadeClassifier”, que es la encargada de la detección de los rostros mediante el método de cascada. Esta librería es delegada para la clasificación de objetos en el video, dividiendo entre objetos o imágenes que no son detectados como rostros. En esta librería está alojado un archivo XML llamado “haarcascade_frontalface_default.xml”, puesto que este documento abarca clasificadores que ya están pre entrenados para facilitar la detección del rostro.

Figura 7
Librería OPENCV.



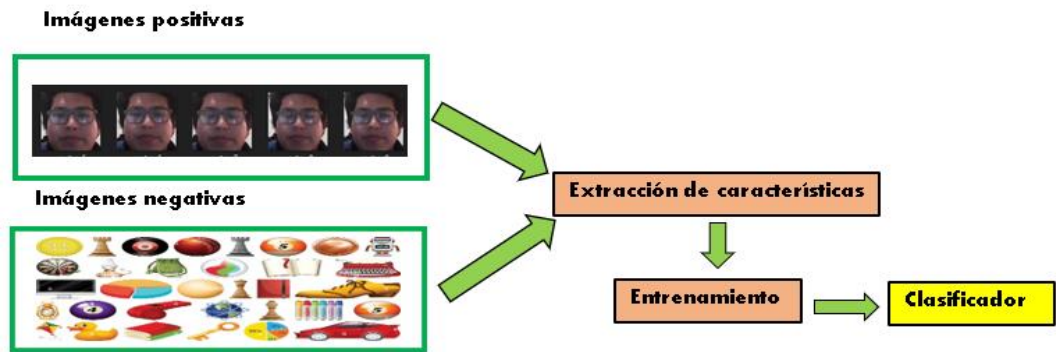
*Nota. Fase de reconocimiento por imagen a través de la librería OPENCV
Elaborado por: Daniel Rea y Cristian Heredia*

1.6.2 Entrenamiento de datos.

Para esta etapa se va a ingresar al “path” donde se alojan los rostros capturados. Se crean dos “arrays”, el primero se lo llama “labels” que posee números consecutivos iniciado desde 0, ya que se diferencia los rostros de cada una de las personas con las que la red neuronal se pueda entrenar, a cada rostro se atribuye una etiqueta única que va a ser de tipo numérico para que el algoritmo haga diferencia de cada uno de los rostros que se van almacenando. El segundo “array” que se va a utilizar es el denominado “rostroDatos”, en el que se almacenará cada una de las imágenes extraídas del entrenamiento.

Al momento de ingresar al “path” se empieza a recorrer cada una de las carpetas que va a contener los nombres de las personas a quienes correspondan las capturas, estas imágenes almacenadas en el “array” indicado anteriormente, y este se va a almacenar en el “array” de “labels”, donde cada uno estará definido por una etiqueta única. Al momento de finalizar el recorrido de cada una de las carpetas se va a autoincrementar el contador desde 1 y esto se repite hasta leer todas las carpetas.

Figura 8
Proceso para captura y entrenamiento de rostros.



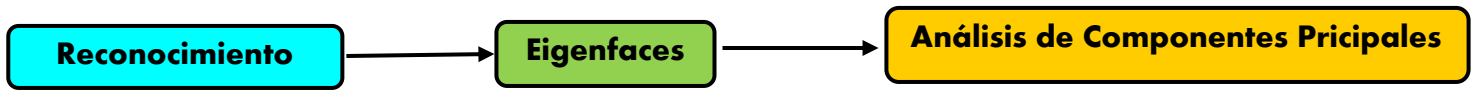
*Nota. Fases para capturar, entrenar y reconocer rostros.
Elaborado por: Daniel Rea y Cristhian Heredia.*

1.6.3 Reconocimiento Facial.

En esta etapa se procede a extraer las imágenes y se guardarán en una ruta específica, donde se encuentran alojadas las características de los rostros capturados. Se genera una instancia llamada “EigenFaceRecognizer_create()”, que permite la interpretación de los archivos con formato “XML” donde se localiza los rostros con sus respectivos datos que se entrenaron. Finalmente se usa la librería CascadeClassifier con la finalidad de establecer la ruta en donde se encuentra almacenado el rostro de la persona al momento de llevar a cabo la lectura del video.

En la predicción del rostro se va a emplear el método conocido como “predict” en el que recibe como parámetro una imagen dimensionada de 150 x 150 píxeles, puesto que estas medidas se emplearon en las respectivas pruebas de funcionamiento tal como se indica en la Figura 9.

Figura 9
Etapa de reconocimiento.



*Nota. Proceso de etapa de reconocimiento.
Elaborado por: Daniel Rea y Cristian Heredia.*

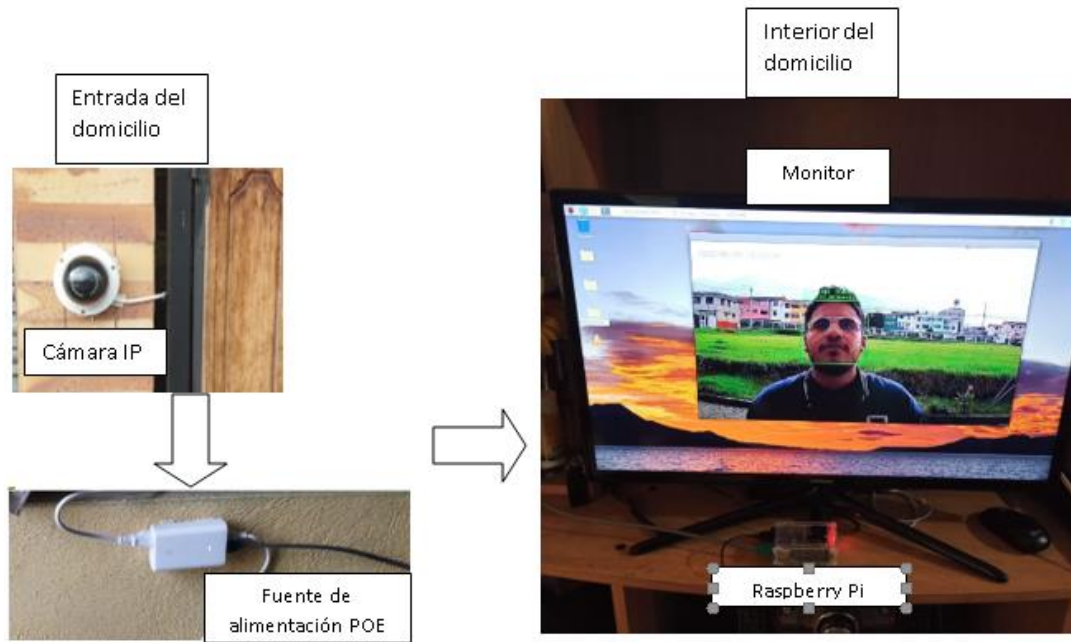
1.7 Precedentes

En el hogar en el que se va a proceder con la instalación del sistema de monitoreo y reconocimiento facial se tomaron en cuenta algunos requerimientos necesarios para el funcionamiento del módulo.

- Inteligencia Artificial.
- Similitud y concordancia con el sistema donde se alojará el módulo.
- Videos de seguridad del reconocimiento, alojados en un dispositivo de almacenamiento.
- Actualización y la visualización de los datos por medio de tablas.

1.8 Implementación física del sistema.

Figura 10
Implementación física del sistema.



*Nota. Implementación Física del Sistema en el Hogar.
Elaborado por: Daniel Rea y Cristhian Heredia.*

CAPITULO II

MARCO TEÓRICO

2. Fundamentación Teórica

En este capítulo se detallarán brevemente los temas planteados en el desarrollo del proyecto con una visión general del mismo.

2.1 Reconocimiento Facial

El reconocimiento para los humanos desde tiempos remotos hasta la actualidad siempre ha sido importante para la identificación de unos a otros, por medio de nombres, sobrenombres, etc. Así como el rostro que nos identifica de manera única es por eso, que, las nuevas tecnologías y sus algoritmos han implementado y aumentaron funcionalidades para reconocer un rostro.

Según David Espinoza y Peter Jorquera (2015) el reconocimiento facial es aquel que permite detectar automáticamente un rostro de una persona mediante imágenes digitales, de tal manera que permita clasificar y obtener características de un conjunto de rostros obtenidos como parámetros principales, al depender de un método que se utilizara con el fin de poder identificar entre un rostro registrado de uno desconocido.

Los métodos de detección son los clasificadores en cascada entre los más conocidos, por lo general no toman en cuenta la imagen en su fondo, por lo tanto, únicamente prioriza en identificar un objeto definido en este caso un rostro de una persona para reconocerla (Jiménez. J.G, 1999).

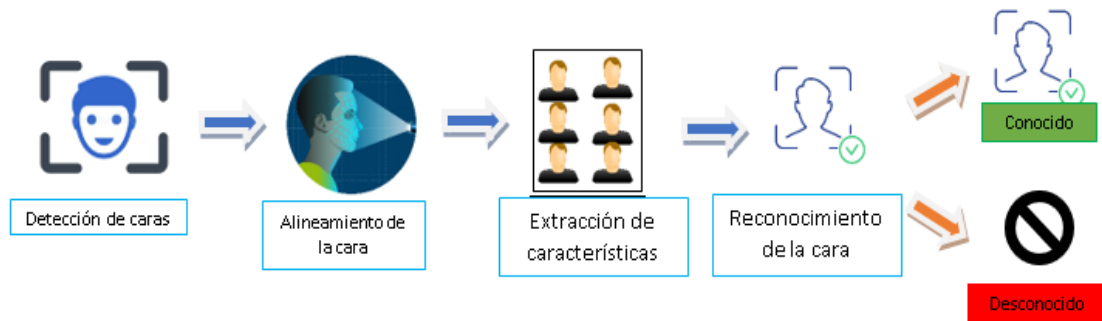
2.1.1 Detección de rostros

En esta Etapa, aparte de que el rostro este alineado a la cámara, también enfoca otros objetos o fondos, que estén alrededor de la cara, la principal función es marcar el rostro con un rectángulo para facilitar la localización.

2.1.2 Etapas del reconocimiento facial

Al momento de intentar reconocer rostros en un sistema, Hacemos uso de un caso particular de etapas, para que el reconocimiento facial obtenga las características faciales como se expone en la Figura 11.

Figura 11
Etapas del reconocimiento facial.



Nota. Etapas del reconocimiento facial.
Elaborado por: Daniel Rea y Cristian Heredia.

2.1.3 Alineamiento de la cara

Esta fase denominada normalización ya que se alinea la cara para que permita ajustar los elementos como su tamaño y forma para tener consistencia con todas las imágenes capturadas que son almacenadas, con el propósito de poder compararlas.

2.1.4 Extracción de características

En esta etapa, se obtiene imágenes con una secuencia que consiste en capturar toda la activación muscular del rostro. Conocido como un vector de características.

2.1.5 Reconocimiento de cara

En esta fase, se compara el vector de características, con el rostro que en ese momento este alineado al dispositivo de imagen, así buscará similitud entre ambos y podrá comprobar si es conocido o desconocido.

2.1.6 Métodos para la detección facial

La detección facial se la relaciona como un método biométrico, que, por medio de características faciales obtenidas, permite identificar individuos, facilitando reconocer rasgos de un rostro con la ayuda de videos o imágenes obtenidas por un dispositivo de imagen de cualquier tipo (Eslava Ríos. J, 2013).

2.1.7 Método Eigenfaces

Este método de detección facial eigenfaces, en concepto es aquel que ejecuta un alcance lineal del espacio de imágenes a un grupo de características de menor magnitud (Ottado. G, 2015).

Por ello a esta disminución, se la aplica usando PCA como técnica, ya que obtiene dicha proyección lineal, facilitando el incremento de la dispersión de todo el conjunto de imágenes proyectadas de la siguiente manera:

- Como primer paso, crea un grupo de N imágenes con diferentes valores, en el que se distingan los rostros, como las distintas posturas, ángulos y diferentes escenarios de iluminación, almacenado todas las características del rostro.
- Como segundo paso, se designa una matriz donde se contenga una imagen y el grupo de las imágenes almacenadas, para que se calcule los Eigenectores con la ayuda de un proceso matemático a través de la matriz de covariancia, convirtiéndose en la abstracción total de las imágenes con el fin que el método pueda comparar vectores entre la imagen original y el grupo de imágenes obtenidas, de tal forma que encuentre la distancia mínima para que indique cual es la imagen correspondiente (Pico, B. O. P., Rondón, P., & Arguello, H. 2011).

2.1.8 Método Fisherfaces

De acuerdo con Granja et.al (s.f) este método abarca lo mismo que Eigenfaces a comparación de otras técnicas conocidas, como por ejemplo la eficiencia, la velocidad

del sistema, la capacidad para operar en varios rostros en un corto tiempo. Eigenfaces tiene el método de reconocimiento de manera frontal y en condiciones análogas a la iluminación, por ello, Fisherfaces es menos sensible a variaciones que puede haber en la iluminación y a los ángulos de los rostros con las imágenes.

Fisherfaces indaga para incrementar la varianza de las distintas muestras entre clases de personas y disminuir entre las muestras que pertenecen a la misma clase, ya que se obtiene mejores resultados al presentarse distintos escenarios con iluminación y expresiones respecto con las imágenes que entreno el sistema.

2.1.9 Librería OpenCV

OpenCV es una librería libre tanto para proyectos comerciales o de investigación que se destaca en la visión artificial, OpenCV permite diseñar aplicaciones como: detección de movimiento, Reconocimiento de objetos, entre otros. Siendo multiplataforma para los diferentes sistemas operativos existentes y para múltiples dispositivos de hardware, computadoras, celulares y Raspberry Pi (Howse.J, 2013).

OpenCV es una biblioteca con demasiada eficiencia, principalmente orientada a objetos y desarrollada por el lenguaje de programación C++ pero permite incorporar conexiones con distintos lenguajes como: Python, Java, Matlab, Octave, Javascript.

2.1.10 Sistemas Embebidos

De acuerdo con Pedre (2012) un sistema embebido también conocido como integrado, es un sistema de computación diseñado con el fin de ejecutar distintas funciones, un sistema embebido este compuesto por un microprocesador ubicado en una placa base. además, está compuesto con interfaces de entrada y salida, y a su vez con una memoria integrada en la placa base, dichos incluyen lenguajes de programación como c o

c++ por medio de compiladores, su principal objetivo es el diseño y a su vez el desarrollo de distintas aplicaciones y prototipos, como lo es la Raspberry pi 4 utilizada en el proyecto.

2.2 Software

Es el componente interno que permite la funcionalidad dentro de un sistema operativo. Este posee un conjunto de programas tales como procedimientos, datos y pautas, que aparte de ser importantes, permiten realizar tareas en el sistema informático (Significados, S.f).

2.2.1 Sistemas Operativo para Raspberry pi

2.2.1.1 Pi-OS (Raspbian). El sistema operativo Pi-OS (Raspbian), es un software completamente gratuito, basándose en el sistema operativo de Debian que se optimizo con el fin que pueda funcionar en el dispositivo Raspberry Pi.

Sin embargo, se lo conoce como un sistema operativo puro, incluyendo más de 35000 paquetes, a su vez posee software en un formato agradable, recompilado que permite tener facilidad en su instalación (Sevilla Rodríguez. J, 2014).

2.2.2.2 Windows 10 Iot Cores. Según Solé (2021) microsoft ha creado la versión del sistema operativo enfocado en otros dispositivos con procesador ARM sea de 32 o 64 bits, este sistema carece del menú de inicio, Cortana entre otros, ya que fue diseñado para desarrollo de soluciones para el internet de las cosas.

2.2.2.3 LibreELEC. Este sistema operativo fue diseñado para un potente centro de multimedia con Kodi, esta distribución de Linux Just enough OS, pensada para ejecutar software (Kodi Sole,2021).

2.2.2.4 Inteligencia artificial. Según Lasse Rouhiainen (2018) Inteligencia artificial (IA) permite la facilidad de obtener capacidad en las máquinas para utilizar algoritmos permitiendo la facilidad de aprender, entender y razonar la información, utilizando lo aprendido al momento de la toma de decisiones parecido a como lo haría un ser humano.

Cuando se habla de IA, se convierte en un tema complejo, ya que implican varias capacidades de los seres humanos y de otros seres vivos que utilizan diariamente para obtener sus objetivos. Es decir, tienen la capacidad de aprender, procesar toda la información para iniciar una tarea y posteriormente tomar decisiones.

Llevando a la semejanza tanto de la inteligencia de otros seres vivos como la de los seres humanos. La IA trata de imitar las capacidades que tiene la mente al utilizar múltiples técnicas que facilitan realizar un conjunto amplio de tareas. Siendo las más conocidas como: aprendizaje automático o conocido como (machine learning), sistemas de expertos, redes neuronales, minería de datos, etc., (Boden, 2017).

La idea que los programas informáticos tengan la facilidad de captar y tomar decisiones es muy importante y algo sobre lo que el ser humano debería ser consciente, ya que día a día sus procesos crecen exponencialmente rápido (Lasse, 2018).

Los distintos lenguajes de programación se estiman como máquinas virtuales al tener instrucciones escritas que a su vez se deben traducir en un código, que el computador permita

interpretar para llevar al inicio de una determinada tarea. Un software de IA debe implementar un orden e inteligencia a las labores que estén por ejecutar, de tal forma que los procesos que fueron realizados siempre evalúen las alternativas con las cuales la toma de decisiones para que el razonamiento sea semejante a la de la mente humana.

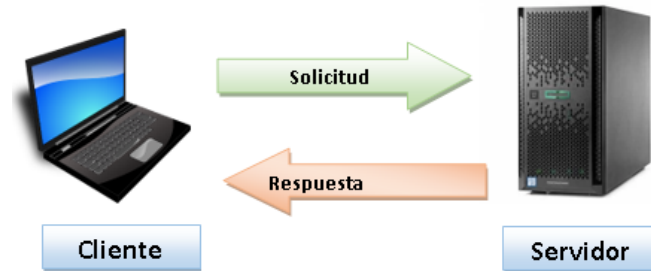
2.2.2.5 Lenguaje Python. Python es un lenguaje de nivel alto, que se orienta a objetos, enfocándose en su flexibilidad de uso. Según Fernández, A. (2013), “Python es un lenguaje interpretado, flexible, multipropósito, y su sintaxis clara convirtiéndolo en un lenguaje muy productivo” (p. 4).

Python es un lenguaje que se caracteriza por tener distintas formas de usar, una de ellas es por las bibliotecas, permiten realizar atajos para algunos tipos de proyectos, por lo tanto, Python se lo puede usar desde cero, para inventar cualquier tipo de programa para el uso de bibliotecas o atajos (Chazallet, 2016).

Python es open source, esto quiere decir que cualquiera puede aportar a su desarrollo y su transmisión. De igual manera no es primordial pagar licencia para distribuir el software desarrollado, permitiendo que su interprete distribuya gratuitamente para diferentes plataformas (Fernández, 2013).

2.2.2.6 MySQL. MySQL es de código abierto (RDBMS) de gestión de bases de datos relacionales, que ocupa un modelo cliente servidor, siendo un software que permite crear, administrar las bases de datos relacionales (DuBois ,2008). Como indica la Figura 12.

Figura 12
Modelo Cliente – Servidor.



*Nota. MySQL modelo cliente-servidor.
Elaborado por: Daniel Rea y Cristian Heredia.*

1. MySQL crea y almacena una base de datos y define la relación de cada tabla.
2. El cliente realiza una solicitud escribiendo instrucciones SQL específicas.
3. El servidor responderá con la información enviando una respuesta a los clientes.

2.3 Hardware

Es el componente físico de un equipo informático. Está conformado por partes eléctricas, electrónicas, electromecánicas y mecánicas como cables, luz, memorias y dispositivos (E/S) entre otros (Significados, S.f).

2.3.1 Entorno de la cámara

La cámara es un dispositivo que se puede colocar en cualquier superficie, pero, el entorno de una cámara depende de la ubicación en que se la coloque, la luz y el tipo de cámara de acuerdo con las necesidades como se describe a continuación.

2.3.2 Luz

Al hablar de luz, hacemos referencia a la luminosidad que tenga la cámara en la superficie que se la instalo, esto permite a que la cámara tenga calidad en la obtención de imágenes, permitiendo eficazmente funcionar la comparación de imágenes obtenidas por medio de métodos, obteniendo eficazmente las características importantes.

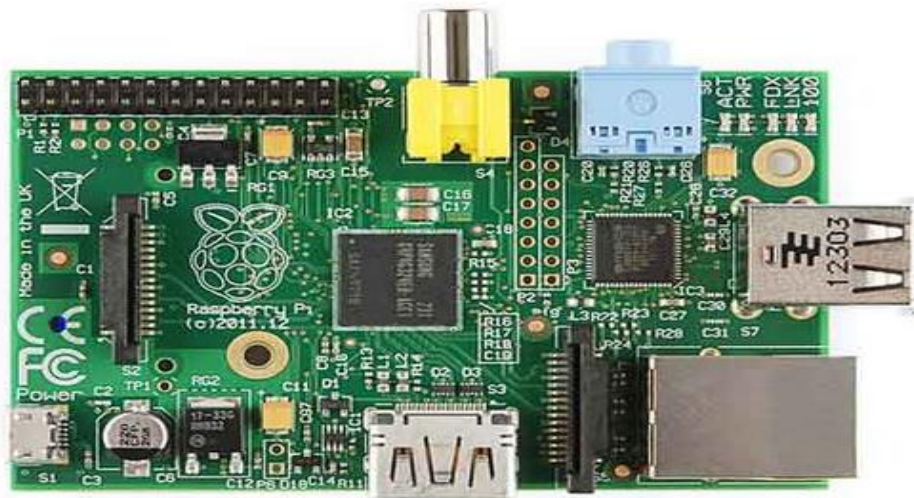
2.3.3 Ubicación

La ubicación en que se instale una cámara es de suma importancia, ya que si una cámara se la instala en una parte alta es mucho más difícil reconocer los rasgos faciales, a lo contrario de una cámara instalada en una posición alineada al rostro que se quiere capturar, ayudando a que a identificar fácilmente.

2.3.4 Modelos de Raspberry

2.3.4.1 Raspberry pi 1. Como característica principal tiene 256Mb de SDRAM, no tiene conectividad a Ethernet y también trae un puerto USB esto es en el modelo A. Según Llamas (2017) En el modelo B ya tiene 512Mb con un puerto a conexión Ethernet y 2 puertos USB, estos modelos traen salidas de video HDMI, DSI y RCA.

Figura 13
Raspberry pi 1



Nota. Vista del hardware de la Raspberry Pi 1.
Elaborado por: Daniel Rea y Cristian Heredia.

2.3.4.2 Raspberry Pi 2. Este modelo fue la versión que se pudo mejorar de la Raspberry Pi-B, el cambio principal fue el incremento en la potencia de cálculo que pudo ser mayor eficiente. Según Llamas (2017) Contiene un SOC Broadcom BCM2836, con procesador de 4 núcleos ARM C rtex A7 a 900 MHz, con Ram de 1Gb, con 4 puertos USB y un puerto de 10/100 Mb Ethernet.

Figura 14
Raspberry pi 2.



*Nota. Vista del hardware de la Raspberry Pi 2.
Elaborado por: Daniel Rea y Cristian Heredia.*

2.3.4.3 Raspberry Pi 3. Seg n Llamas (2017) este modelo se basa en el cambio de conectividad inal mbrica ya que a nadir n la conexi n Bluetooth 4.1 con una conexi n Wifi 802.11n, incrementando su potencia con un SOC Broadcom BCM2837 y tambi n con un procesador AMv8 1.2 GHz de 64 bits con 4 n cleos, 4 puertos USB y el puerto Ethernet.

Figura 15
Raspberry pi 3.



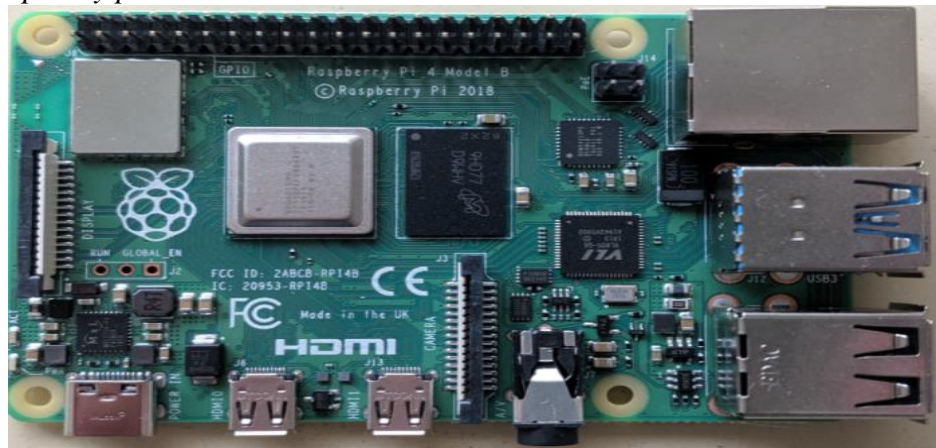
*Nota. Vista del hardware de la Raspberry Pi 3.
Elaborado por: Daniel Rea y Cristian Heredia.*

2.3.4.4 Raspberry Pi 4. La Raspberry Pi 4, incorpora un procesador conocido como Quad Core con una arquitectura de 64 bits con una velocidad de 1.5GHz y un videoCore VI 3D que permite tener una resolución con un códec H.265de 4K y 60 FPS y un códec H.264 a resoluciones de 1080p a 60FPS o también a 30 FPS, también incorpora cuatro puertos USB, dos puertos 2.0 y 2 puertos 3.0, Incorpora un puerto Ethernet con velocidades más de 900Mbps reales en redes locales

Mientras su memoria RAM de este modelo de Raspberry, ocupa el tipo LPDDR4 con velocidad de 2400MHz de 8 GB permitiendo explotar al máximo su potente hardware. En la parte de alimentación, la Raspberry Pi necesita una fuente que permita repartir tenciones de 5V o 5.1V, a su vez intensidades de 3 A para una correcta alimentación.

En cuanto a la conectividad inalámbrica, incorpora WIFI 802.11, bluetooth 5.0 con BLE, 2 micro puertos HDMI para salida de audio y video, también se puede conectar un display o una cámara gracias a sus puertos (2-lane MIPI DSI), asimismo posee un slot para mico SD, ya que en este se instalará el sistema operativo que va a funcionar en la Rspberry Pi.

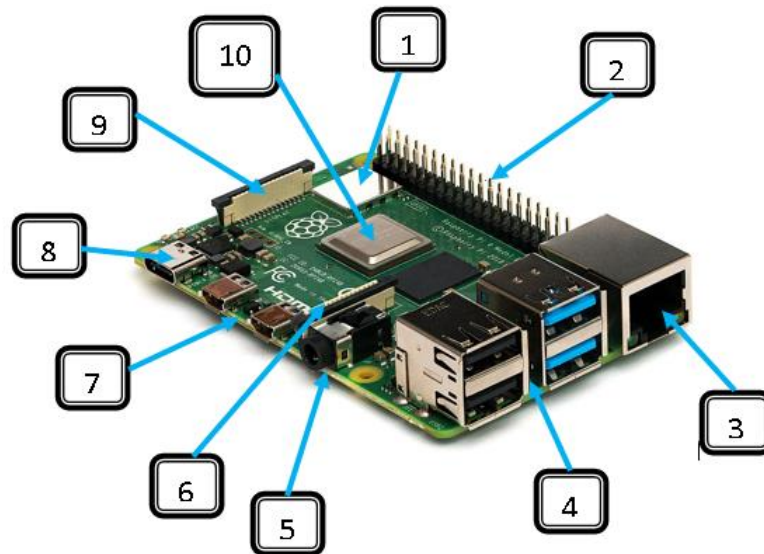
Figura 16
Raspberry pi 4



*Nota. Vista del hardware de la Raspberry Pi 4.
Elaborado por: Daniel Rea y Cristian Heredia.*

2.3.4.5 Partes de la Raspberry Pi 4

Figura 17
Partes de la Raspberry pi 4.



*Nota. Partes físicas de la Raspberry pi 4.
Elaborado por: Daniel Rea y Cristian Heredia.*

Tabla 1
Partes de la Raspberry pi.

Numero	Partes
1	Wfi 2,4 / 5 GHz, Bluetooth 5.0
2	Pines GPIO
3	2 USB 2.0 y 2 USB 3.0
4	Puerto Ethernet
5	Salida de Audio
6	Puerto de cámara MIPI CSI
7	2x Conectores HDMI
8	Entrada de alimentación Tipo-C
9	Puerto Display
10	Procesador Quad Core Cortex a 1.5GHz Memoria ram 8GB

Nota. partes principales de la Raspberry pi.
Elaborado por: Daniel Rea y Cristhian Heredia.

CAPITULO III

METODOLOGIA

3.1 Análisis De Software Y Hardware.

En este capítulo se va a describir sobre el software y hardware utilizado para el desarrollo del proyecto con sus respectivas características, modelo del dispositivo de Raspberry a utilizar, el uso de las bibliotecas, el lenguaje de programación utilizado y el funcionamiento en general.

3.1.1 Hardware

Tabla 2
Tipos de cámaras.

Tipos de Cámaras	Descripción
Webcams	Las webcams son cámaras más sencillas y económicas, se encuentran fácilmente en el mercado, ya que no tienen de tantos mecanismos y son de fácil instalación.
Cámaras IP	Las cámaras IP son dispositivos que tienen la facilidad de conectarse a internet, ya que en sus funciones principales tienen WI-FI, permitiendo captar las imágenes visualizando desde un dispositivo smartphone.
Cámara infrarroja o visión nocturna	Las cámaras infrarrojas son dispositivos que se utilizan para vigilancia para espacios de poca luz al tener un sistema infrarrojo y un sensor que permite captar imágenes en la noche, también llamados dispositivos de visión nocturna LED, que sus valores son los más altos en el mercado.
Cámaras de exterior	Estos dispositivos son estudiados y diseñados para soportar cambios climáticos ya que se utilizan en exteriores al ser diseñadas para aguantar contra el agua o viento estas cámaras tienen un precio elevado a las demás.

Nota. Se eligió la cámara IP para el proyecto.
Elaborado por: Daniel Rea y Cristhian Heredia.

El tipo de cámara que se escogió es la cámara IP, debido a que estos tipos de cámaras brindan altas resoluciones de imagen, conexión a internet, y al ser cámaras diseñadas para exteriores siendo resistentes al agua y polvo, ya que son cámaras completas, por lo cual permite aumentar satisfactoriamente el reconocimiento facial en este tipo de proyectos.

Tabla 3
Tipos de Raspberry pi.

Modelo Raspberry Pi 1	Este modelo consta de un procesador ARM11 de un núcleo 700MHz, con un SOC Broadcom BCM2835.
Modelo Raspberry Pi 2	Este modelo consta de un un procesador ARM Cortex A7 900 MHz de 4 núcleos, con una SOC Broadcom BCM2836.
Modelo Raspberry Pi 3	Este modelo consta de un procesador 1.2GHz ARMv8 de 4 núcleos de 64 bits incluyendo conectividad Wifi 802.11n, con Bluetooth 4.1
Modelo Raspberry Pi 4	Este modelo consta de un Broadcom BCM2711, 1.5 GHz de 4 núcleos a con conectividad wifi 802.11ac con Bluetooth 5,0.

*Nota. Se eligió Raspberry pi 4 para el proyecto.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Por otro lado, el modelo que se tomó fue Raspberry pi 4 puesto que el algoritmo seleccionado requiere de un mayor procesamiento, que consta de un procesador Broadcom BCM2711 de 4 núcleos y 8 Gb de Ram, permitiendo obtener un mejor rendimiento al sistema instalado en el dispositivo.

3.1.2 Software

Tabla 4.

Tipos de sistemas operativos en Raspberry pi 4.

Raspberry Pi OS (Raspbian)	Este sistema operativo es oficial de los sistemas Raspberry pi ya que los propios desarrolladores las están manteniendo y está basada en la distribución de Linux.
Windows 10 IoT Cores	Este sistema operativo es la nueva versión de Windows para dispositivos pequeños, pensada para desarrollo de soluciones para internet de las cosas
LibreELEC	Sistema operativo para quien desea crear un centro multimedia con Kodi, esta distribución de Linux enough Os que aprueba la ejecución de un software por defecto.

Nota. Sistemas operativos del Raspberry pi.

Elaborado por: Daniel Rea y Cristhian Heredia.

El sistema operativo elegido fue el PI-OS (Raspbian) debido a que es compatible con todas las versiones actuales de Raspberry Pi, y al tener la facilidad de utilizarlo como un sistema operativo convencional, en la cual este sistema permite el desarrollo de proyectos electrónicos, reconocimiento, etc, como el de este proyecto que al ocupar este dispositivo se pudo enlazar a una cámara IP, con el fin de probar el sistema de detección y reconocimiento facial.

Tabla 5

Tipos de lenguaje de programación.

Java Script	Permite automatizar y optimizar modificando elementos HTML. En la actualidad evoluciona y se encuentra disponible en aplicaciones móviles, necesita de otro programa para poder ejecutarlo.
Python	Lenguaje de programación más conocido a nivel mundial por ser de código abierto.

Nota. Se toma en cuenta el lenguaje de Python para la realización de este proyecto.
Elaborado por: Daniel Rea y Cristhian Heredia.

Se seleccionó el lenguaje de programación Python, puesto que este consta de librerías como OpenCV que implementa algoritmos para la detección de objetos y rostros. OpenCV posee una clase abstracta llamada face_recognition que trabaja con diferentes algoritmos, uno de ellos es Eigenfaces (PCA) debido a que facilita el incremento de la dispersión de todas las imágenes proyectadas.

3.2 Análisis económico del módulo.

Se analizó la economía para el uso del módulo, los requerimientos que necesita para tener el funcionamiento, teniendo en cuenta a software y hardware, se requirió de un precio real y el costo para dicho diseño del módulo de reconocimiento facial.

3.2.1 Requerimientos para la funcionalidad del sistema

El módulo requiere como principalmente una cámara que este en la parte exterior del domicilio, para la visualización de personas que estén rondando dicho hogar, por lo cual, se enlazará a una Raspberry Pi 4 que en este dispositivo permite alojar el sistema de reconocimiento facial, se recomienda que la cámara tenga una buena resolución y estabilidad de enfoque.

3.2.2 Costos de hardware y software del módulo.

Para analizar los costos, se tomó en cuenta precios referenciales, de todos los dispositivos que se usó y tanto como el desarrollo del módulo y su instalación, para el funcionamiento del sistema.

Tabla 6
Costos de Hardware.

Dispositivo	Función	Precio
Raspberry Pi4	Encargado de alojar el sistema de reconocimiento facial	\$190
Cámara IP	Captación de rostros Luz infrarroja Nocturna	\$150
Case Raspberry pi 4	Permite al cuidado de la Raspberry-PI con su respectivo ventilador para evitar temperaturas altas, por usos prolongados en la placa	\$15
Tarjeta de memoria 64GB	Permite alojar el sistema operativo con el que funciona la Raspberry-PI	\$12
	TOTAL	\$ 367

Nota. Valores de costos de hardware.
Elaborado por: Daniel Rea y Cristhian Heredia.

Tabla 7
Costos de Software e instalación del módulo.

Desarrolladores	Tareas	Tiempo	Valor
Desarrollador 1	Encargado de instalación y configuración del hardware del proyecto y programación del sistema.	5 meses	\$120
Desarrollador 2	Encargado de la instalación y configuración del hardware del proyecto y programación del sistema.	5 meses	\$120
	TOTAL		\$240

Nota. Valor total de desarrollo.
Elaborado por: Daniel Rea y Cristhian Heredia.

En la Tabla 7, se muestra cada requerimiento de todas las horas trabajadas seguidamente, por el tiempo de demora que tomo en la finalización del proyecto, también se valoró el precio de instalación que interviene la instalación de la cámara e instalación del dispositivo Raspberry pi para la funcionalidad del sistema.

Tabla 8
Costo total del proyecto.

Dispositivo	Precio
Costo de dispositivos de hardware.	\$367
Costo de desarrollo.	\$240
Instalación	\$30
TOTAL, COSTO PROYECTO	\$637

Nota. Se toman los valores de las Tablas 6 y 7 para obtener el valor total. Elaborado por: Daniel Rea y Cristhian Heredia.

CAPITULO IV

PRUEBAS Y RESULTADOS

En el capítulo se explica las diferentes pruebas que se van a realizar, también se muestra cada uno de los resultados obtenidos para comprobar el funcionamiento.

4.1 Análisis de pruebas

Para este prototipo se evaluó tres tipos de pruebas como la: funcionalidad, sensibilidad y confiabilidad con el fin de comprobar la calidad del software, midiendo los requerimientos para cumplir con las expectativas que medirán al desempeño.

4.1.1 Pruebas de funcionalidad del reconocimiento.

En esta sección se ejecutó pruebas que miden la funcionalidad del algoritmo, con el fin, de comprobar el porcentaje de reconocimiento de cada miembro familiar con distintos escenarios tales como:

- Análisis de detección facial y reconocimiento.
- Pruebas con números de fotos del rostro.
- Pruebas de luminosidad
- Pruebas con distancias

Para este apartado de pruebas se vio pertinente tomar 4 personas, cada una de ellas se colocó al frente de la cámara en un lapso de 5 min, por ende, se obtuvo un aproximando de 70 registros entre reconocidos y desconocidos, este grupo de personas están compuestos por dos hombres, una mujer y una persona desconocida con el fin de poder analizar si el sistema detecta rostros en la cercanía y comprueba si pertenecen o no al grupo familiar.

A continuación, se demuestra los resultados que se obtuvieron los distintos escenarios propuestos.

4.1.2. Análisis de detección facial y reconocimiento en miembros del grupo hogar.

Prueba 1

Como primer paso, se comenzó por la detección del rostro, después la extracción y análisis de características, por último, marca el nombre de la persona que se reconoció; por ejemplo, con esta persona del grupo familiar llamada Ricardo como se indica a continuación en la Figura 18.

Figura 18

Reconocimiento facial de Ricardo Rea.

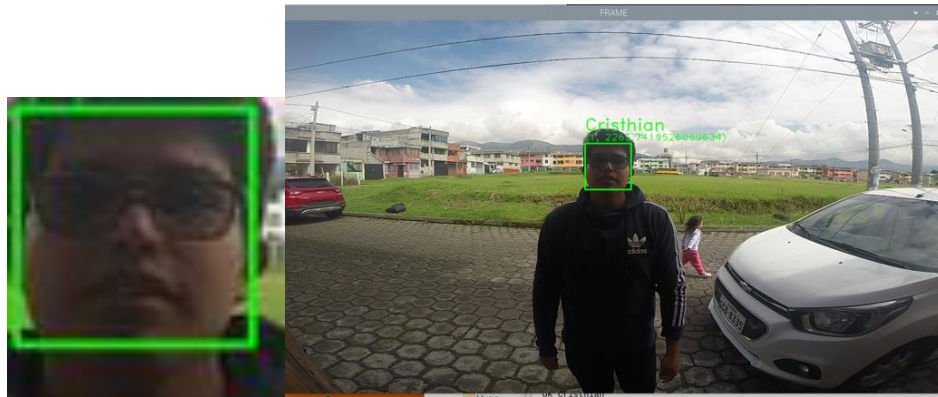


*Nota. Reconocimiento facial por medio de cámara ip.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Prueba 2

La segunda prueba se realizó con uno de los autores del trabajo de titulación, en este caso con Cristhian. Para este reconocimiento se observó que dicha persona utiliza lentes; como se muestra en la Figura 19, permitiendo comprobar que el sistema detecté y reconozca a esta persona que lleva puesto un objeto en el rostro.

Figura 19
Reconocimiento facial de Cristhian Heredia.

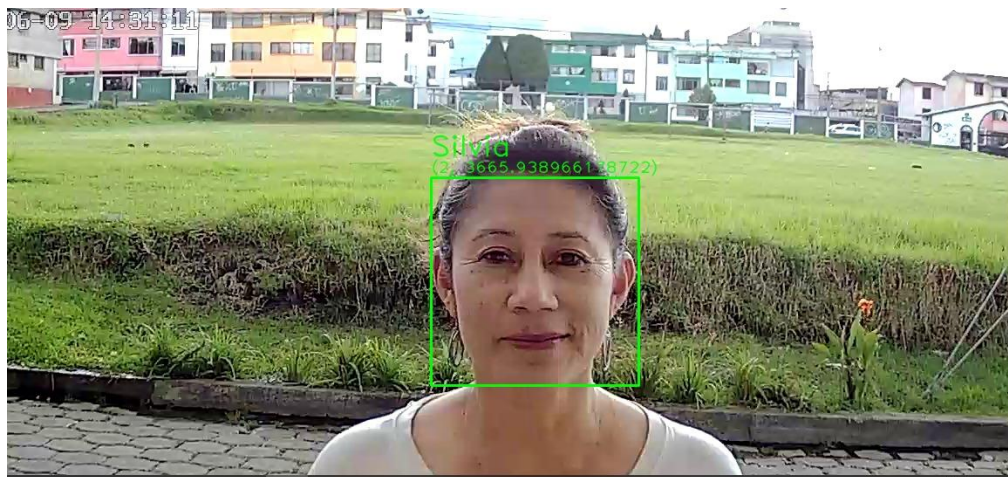


*Nota. Reconocimiento facial a diferentes distancias.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Prueba 3

La tercera prueba se realizó con una mujer llamada Silvia, la cual utilizó accesorios como aretes, este detalle permitió probar si el módulo detecta y reconoce al ocupar dicho accesorio. Tal como se evidencia en la Figura 20, donde se muestra que Silvia fue detectada y reconocida por el sistema.

Figura 20
Reconocimiento facial de Silvia.



*Nota. Reconocimiento facial a Silvia Rodríguez.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Prueba 4

Por último, se analizó un factor importante; pues al tratarse de la instalación de la cámara y su ubicación la cual debe estar en el exterior del domicilio, en esta parte se ejecutó una prueba de acercamiento; por lo cual el sistema de detección y reconocimiento facial, detecto al rostro más cercano por ende ignora al otro usuario que se encontró más alejado de la cámara como se indica en la Figura 21.

Figura 21

Detección del rostro más cercano.



Nota. Detección a corta distancia.

Elaborado por: Daniel Rea y Cristhian Heredia.

Para todas estas pruebas realizadas, se analizó que el sistema detecta el rostro de personas en la cercanía del hogar, así se comprobó, que las personas que reconoció forman parte del grupo hogar, al contrario de rostros desconocidos detectados en la cual el sistema no reconoce al no ser miembros del grupo hogar permitiendo alertar.

4.1.3 Pruebas de reconocimiento según el número de fotos.

Para el apartado de pruebas de funcionalidad, se comparó con 100, 400 y 600 características como lo muestra en la Tabla 9, que el sistema agrupo en el momento que capturó al rostro, así se comprobó con cuantos números de fotografías el sistema tiene mayor precisión en reconocer al usuario.

Tabla 9

Resultados de reconocimiento por el número de fotos.

Persona	N° fotos	N° Registros	N° de registros reconocidos	N° de registros desconocidos	% Reconocido	% Desconocido
Daniel	400	64	29	35	45.3%	54.6%
	600	68	62	6	91.1%	8.8%

Nota. Comparación según números de fotos extraídas.

Elaborado por: Daniel Rea y Cristhian Heredia.

- En el apartado de pruebas se comprobó como muestra en la tabla 6, con 400 fotos, el sistema no funcionó de manera óptima, ya que no obtuvo las suficientes imágenes para el reconocimiento a los miembros del grupo familiar, ya que obtuvo un porcentaje de reconocimiento del 45.3% incrementando al desconocimiento del rostro.
- Al contrario, con 600 fotos se comprobó que con este grupo de características almacenadas el sistema, obtuvo una mejor funcionalidad ya que el número de registros reconocidos se incrementó al 91.1%, comprobando que mientras más características almacenadas, incrementa la precisión de reconocimiento facial de los usuarios.

4.1.4 Pruebas de luminosidad.

Para esta etapa se realizó las respectivas pruebas de luminosidad en la cual se comparó dos escenarios como:

- luz natural es aquella que se mantiene en un día normal soleado.
- luz nocturna es aquella que está presente por un dispositivo luminoso, es decir, donde se utilizó la luz infrarroja de la cámara IP.

Se evaluó las pruebas en el día para la luz natural y en la noche, un sensor infrarrojo instalado en la cámara IP que se utilizó, como se indican en las Figuras 22 y 23 que muestra los reconocimientos en la luz natural y nocturna.

Figura 22

Reconocimiento facial luz nocturna.



*Nota. Vista cámara IP con luz infrarroja.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Figura 23

Reconocimiento facial luz natural.



*Nota. Reconocimiento facial durante el día.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Tabla 10

Pruebas de funcionalidad de luminosidad.

PRUEBAS DE LUMINOSIDAD						
Miembro familiar	Tipo de luz	Número de Registros	Número de registros reconocidos	Número de registros desconocidos	% Reconocido	% Desconocido
Daniel	Natural	72	68	4	94.5%	5.5%
	Nocturna	67	56	11	83.5%	16.4%
Ricardo	Natural	70	65	5	92.8%	7.2%
	Nocturna	68	56	12	82.3%	17.6%
Silvia	Natural	72	67	5	93%	7%
	Nocturna	65	55	10	84.6%	15.4%
Desconocido	Natural	62	0	62	0%	100%
	Nocturna	74	10	64	13.5%	86.5%

*Nota. Pruebas analizadas a cada uno de los miembros del grupo hogar.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Para el reconocimiento facial la iluminación es una característica de suma importancia, por lo tanto, entre más claros sean los rasgos faciales de las personas que se detectó, se obtuvo mayor porcentaje de reconocimiento, esto se evidencia en la Tabla

10, donde se mostró que con la luz natural obtuvo un mayor desempeño donde se realizó las pruebas de funcionalidad de reconocimiento con cada uno de los usuarios, al contrario de la luz artificial que descendió hasta los 10% de su porcentaje de reconocimiento de la luz natural, debido al cambio en la luminosidad en el rostro.

Tabla 11

Promedios de porcentajes de las pruebas de funcionalidad de los usuarios de reconocimiento por luminosidad.

RESULTADOS FINALES DE PRUEBAS DE LUMINOSIDAD		
Miembro familiar	Porcentaje reconociendo (%) luz natural	Porcentaje reconocimiento (%) luz Nocturna
Daniel	94.5%	83.5%
Ricardo	92.8%	82.3%
Silvia	93%	84.6%
Promedio total de reconocimiento de luminosidad	93.4%	83.4%
Persona desconocida	0%	13.5%

Nota. Se realiza una sumatoria de todos los porcentajes obtenidos de las pruebas de luminosidad.

Elaborado por: Daniel Rea y Cristhian Heredia.

Según en la Tabla 11 se obtuvo como resultados, que la luz natural logró un porcentaje de reconocimiento del 93.4%, comprobando que el algoritmo funciono tanto en pruebas de luz natural como de artificial al momento que reconoció el rostro, ya que el 83.4% de reconocimiento se obtuvo con la luz artificial, comprobando que su porcentaje de reconocimiento descendió un 10% menos que el de la luz natural, también la persona desconocida arrojó un reconocimiento en la luz artificial, debido a que por la resolución de la cámara el sistema, confundió con algún rasgo del miembro del hogar.

4.1.5 Pruebas de variación de la distancia.

Por último, se aplicó pruebas en variación de la distancia, consistió en alejarse de la cámara lo más lejos posible, se hizo referencia a la distancia en metros de 1m,

1.5m y 2m, así se comprobó hasta que distancia máxima el sistema funcionó para reconocer el rostro.

Tabla 12

Pruebas de reconocimiento facial con diferentes distancias.

PREUEBAS Y PORCENTAJES DE VARICION DE DISTANCIAS.						
Miembro familiar	Distancia (m)	Número de Registros	Número de registros reconocidos	Número de registros desconocidos	% reconocido	% Desconocido
Daniel	1	72	67	5	93%	7%
	2	69	51	18	73%	27%
	3	65	32	33	49.2%	50.8%
Ricardo	1	72	68	4	94.4%	5.6%
	2	69	49	20	71%	29%
	3	68	30	38	44%	56%
Silvia	1	72	66	6	91.6%	8.4%
	2	75	54	21	72%	28%
	3	66	31	35	47%	53%
Desconocido	1	68	1	67	1.5%	98.5%
	2	71	0	71	0	100%
	3	65	0	65	0	100%

Nota. Pruebas para el reconocimiento facial en varias distancias.

Elaborado por: Daniel Rea y Crithian Heredia.

En el apartado de pruebas, se comparó las distintas distancias, tal como se muestra en la Tabla 12, donde se evidencio que la distancia obtuvo un factor muy importante, por esta razón la persona que estuvo más cerca y alineada a la cámara el sistema obtuvo una mayor precisión de reconocimiento, a lo contrario de las demás distancias, que el sistema experimento elevar el esfuerzo para reconocer los rostros.

Tabla 13

Promedios de porcentajes de las pruebas de reconocimiento de los usuarios con variación de distancias.

Porcentajes			
Miembro familiar	1m	2m	3m
Daniel	93%	73%	49.2%
Ricardo	94.4%	71%	44%
Silvia	91.6%	72%	47%
Promedio total de reconocimiento con distancias	93%	72%	46.7%
Desconocido	1.5%	0%	0%

Nota. Se realiza una sumatoria de todos los porcentajes obtenidos de las pruebas de distancias.

Elaborado por: Daniel Rea y Cristhian Heredia.

Según la Tabla 13, los resultados se compararon y se obtuvo un promedio de los porcentajes obtenidos, de las distintas distancias que cada usuario se colocó para las pruebas, obteniendo como resultados:

- La distancia más lejana de 3 metros que se obtuvo un 46.7% de porcentaje bajo al igual que la persona desconocida, de tal manera que mientras más alejado estuvo el usuario el reconocimiento se dificultó en reconocer, ya que no logró captar los rasgos faciales completos a la variación de distancia, algo similar ocurrió con la distancia de 2 metros, solo que en esta distancia sí obtuvo menos registros desconocidos, ya que obtuvo el 72% de registros reconocidos, finalmente la distancia de mayor función entre el rostro y la cámara, fue el de la distancia de 1m ya que esta distancia arrojó un porcentaje 93% de reconocimiento, comprobando que a esta distancia con una correcta alineación a la cámara, el sistema logró captar los puntos faciales completos para el reconocimiento.

4.2. Pruebas de sensibilidad.

En base a las pruebas de funcionalidad se obtuvieron resultados de porcentajes de reconocimiento facial, en las distintas pruebas realizadas a cada uno de los usuarios del hogar, Por lo cual se ha tomado en cuenta las siguientes pruebas experimentales como:

- Análisis de detección según las distancias.
- Análisis de luminosidad

Esto con el fin de describir cuantitativamente los distintos experimentos, en base a la sensibilidad del sistema, como los contextos demostrados a continuación en la Tabla 14.

Tabla 14
Interpretación de resultados de pruebas de sensibilidad.

Resultado	Significado	Descripción
VP	Verdaderos Positivos	cuentan los rostros correctos que reconoció y registra las caras reconocidas.
FP	Falsos Positivos	cuentan los rostros correctos que reconoció, pero registra aquellas caras como incorrectas cuando en realidad estas son las correctas.
FN	Falsos Negativos	cuentan los rostros incorrectos que reconoció, pero registra aquellas caras como correctas cuando en realidad son las incorrectas
VF	Verdaderos falsos	cuentan los rostros incorrectos que detecto y registra las caras desconocidas.

*Nota. Interpretación de los resultados al realizar pruebas de sensibilidad.
Elaborado por: Daniel Rea y Cristhian Heredia*

Para las pruebas de sensibilidad del sistema se engloba la distancia y la luminosidad con el fin de demostrar los fallos y aciertos. Por lo tanto, en las siguientes tablas se ilustrará la sensibilidad, el valor predictivo y la efectividad.

La función de la sensibilidad es clasificar correctamente una imagen como en este caso es un rostro, por ello se debe tener en cuenta:

- VP - (verdaderos positivos)
- FP - (Falsos Positivos)
- FN - (Falsos negativos)
- VF - (verdaderos Falsos)

A continuación, se demuestran los resultados y porcentajes de los rostros capturados de todos los miembros del grupo hogar en los escenarios de luminosidad y distancias, estos obtenidos en las pruebas de funcionalidad mostradas en las Tablas 15 y 16 en donde se muestran los porcentajes de registros reconocidos y desconocidos de todo el grupo familiar, así facilitó la comparación para las pruebas de sensibilidad.

Tabla 15*Porcentajes totales de luminosidad del grupo familiar*

Luminosidad	Porcentajes reconocimiento		Porcentajes desconocidos	
	Luz natural	Luz nocturna	Luz natural	Luz nocturna
Promedio total del grupo familiar	93.4%	83.4%	6.6%	16.6%
Promedio personas desconocidas	0%	13.5%	100%	86.5%

*Nota. Pruebas de luminosidad realizadas al grupo familiar.**Elaborado por: Daniel Rea y Cristhian Heredia.***Tabla 16***Porcentajes totales de distancias del grupo familiar.*

Distancias	Porcentaje reconocidos			Porcentajes desconocidos		
	1 m	2 m	3 m	1 m	2 m	3 m
Promedio total del grupo familiar	93%	72%	46.7%	7%	28%	53.3%
Promedio personas desconocidas	1.5%	5%	0%	98.5%	95%	100%

*Nota. Porcentajes obtenidos en distancias determinadas.**Elaborado por: Daniel Rea y Cristhian Heredia.*

Tabla 17
Resultados finales de sensibilidad

Todos los miembros del grupo hogar		Distancias y porcentajes de reconocidos y desconocidos											
		1 m				2 m				3 m			
		VP	FP	FN	VF	VP	FP	FN	VF	VP	FP	FN	VF
Luminosidad	Luz Natural	90%	5%	3%	2%	72%	6%	6%	16%	42.5%	4.5%	5%	48%
	Luz Nocturna	85.5%	3.25%	4.2%	7.05%	65.5%	10.5%	5.6%	18.4%	35.3%	3.25%	4.25%	57.2%

Nota. Resultados obtenidos al momento de realizar las pruebas de sensibilidad.
Elaborado por: Daniel Rea y Cristhian Heredia

Con estos resultados se obtuvo que la luminosidad y distancia toma como un factor importante en los sistemas de detección facial ya que, al unir estas dos pruebas, se conoció que el sistema confundió rasgos faciales por parentesco familiar, estado de la luz, distancia, en ciertos periodos de tiempo, pero se comprobó que a pesar de todos esos escenarios el sistema reconoce al usuario registrado.

4.3. Pruebas de confiabilidad.

Este tipo de pruebas se evaluó por escenarios, que evidencio que el sistema brinda seguridad a los usuarios como es la monitorización y detección de gente desconocida, esto permitió alertar por medio de notificaciones que un desconocido se encuentra rondando el hogar, se evaluó la confiabilidad del módulo en cuanto a la seguridad del hogar.

Conociendo que el módulo envía datos desconocido, se analizó la confiabilidad del número de veces que el sistema mostro y notificó que persona desconocida rondo el domicilio así se obtuvo resultados de confiabilidad.

4.3.1 Pruebas de confiabilidad de monitoreo.

Métrica:

El módulo debe monitorear desconocidos en los lapsos de tiempo establecidos.

Escenario 1:

En la prueba se tomó lapsos de tiempo del día, tomando 2 horas en la mañana, medio día y tarde, así se observó que tan confiable es la detección y a cuantos individuos no registrados alerto como desconocidos.

Datos de tabla:

N°: Numero de prueba.

Tiempo: descripción si es en la mañana, tarde o noche.

Horario: Lapso de horas que se usó en las pruebas.

N° Detecciones: Número de veces que el sistema detecto una cercanía.

N° conocidos: Cantidad de veces que reconoció a un miembro familiar.

N° Desconocidos: cantidad de veces que detecto un desconocido rondando el hogar.

Tabla 18

Pruebas de confiabilidad de monitoreo.

N°	Tiempo	Horario	N° Detecciones	N° conocidos	N° Desconocidos
1	Mañana	8am-10am	15	13	2
2	Medio día	12am-2pm	5	4	1
3	Tarde	4pm-6pm	11	9	2

Nota. Pruebas de confiabilidad durante el día.

Elaborado por: Daniel Rea y Cristhian Heredia

Al finalizar la prueba, se evidencio, que los datos obtenidos como se muestran en la Tabla 18, son los requeridos y cumplen con la métrica, observando que en los distintos lapsos de tiempo que se probó el módulo, este detecto cercanía al hogar de gente desconocida que

rondo por la vivienda , se observó que en la mañana y tarde se alertó de 2 personas desconocidas rondando el hogar, y al medio día 1 persona rondo el domicilio, permitiendo tener una mayor seguridad para los usuarios de dicho hogar, ya qué se comprobó que el sistema registro datos al momento que detecto un desconocido.

4.3.2 Pruebas de confiabilidad de notificación.

Como se evidencio en la anterior prueba se observó el número de desconocidos que rondaba la vivienda, para esta fase de pruebas se hace referencia de las veces de detecto el sistema a los desconocidos y que tan confiable el sistema notifico al usuario, de las personas desconocidas que rondaron el domicilio y que el módulo detecto.

Datos de tabla:

N°: Numero prueba.

Tiempo: descripción si es en la mañana, tarde o noche.

Horario: Lapso de horas que se usó en las pruebas.

N° Desconocidos: cantidad de veces que detecto un desconocido rondando el hogar.

N° Detección desconocido: Cantidad de veces que detecto como desconocido depende del tiempo que se acercó al módulo.

N° de notificación: número de veces que notifico presencia del desconocido.

% de notificación: muestra el porcentaje de veces que notifico el sistema.

Tabla 19
Pruebas de confiabilidad de notificación.

N°	Tiempo	Horario	N° Desconocidos	N° de registros desconocidos	N° de notificación	% Notificación
1	Mañana	8am-10am	2	6	4	100%
2	Medio día	12am-2pm	1	3	2	100%
3	Tarde	4pm-6pm	2	7	3	85.71%

*Nota. Pruebas de confiabilidad de notificación.
Elaborado por: Daniel Rea y Cristhian Heredia.*

Al terminar la prueba se puede comprobar la confiabilidad ya que esta prueba se partió desde el Tabla 19, comparando las veces que se acercó una persona desconocida, esto permitió medir la confiabilidad, ya que se comprobó que el sistema notifica inmediatamente apenas detecta a una persona desconocida que se encuentra fuera del hogar, conociendo que el sistema notifico 2 veces por cada registro desconocido, esto se obtuvo como resultado que el sistema siempre mantuvo alerta al usuario.

4.4 Análisis de resultados

El sistema de detección facial se evaluó en varias pruebas de funcionalidad, sensibilidad y confiabilidad, obteniendo como resultados que:

En cuanto a los resultados de funcionalidad, se concluyó que el sistema no tuvo problemas para reconocer el rostro con accesorios, luminosidad, distancia y variaciones faciales como las muecas, por lo que se entrenó al rostro con esas características en el sistema. Se verifico la confiabilidad en el momento que el sistema detecto personas desconocidas y se envió una notificación de alerta a los usuarios; con la finalidad de mantener la seguridad del hogar. Seguido de las pruebas de sensibilidad en la cual se comprobó que el sistema en ciertos escenarios como la luz se confunde con el rostro de algunas personas también en las distancias con esto se comprobó los porcentajes de sensibilidad que obtuvo el prototipo.

Al analizar los resultados se concluyó que el sistema funciono correctamente ante todos los escenarios de pruebas. Sin embargo, el sistema pudo funcionar mejor si ocupara una cámara de mayor resolución, de grados de captación, esto permitiría incrementar la efectividad en los reconocimientos, también incrementar la detección y reconocimiento a mayores distancias, logrando mayor seguridad en el hogar.

CONCLUSIONES

- Tras la experimentación con una Raspberry pi 4, se concluye que el procesamiento del algoritmo con la cámara IP se vuelve demasiado alto ya que llega a ocupar todos los recursos del dispositivo Raspberry, por lo tanto, para usos experimentales de casos de estudios no se recomienda, ya que se necesita de mayor procesamiento y nivel de respuesta para el reconocimiento en tiempo real.
- Además, se analizó que ha mayor cantidad de fotos registradas almacenadas, aumenta la lentitud del entrenamiento de rostros, por ello se convierte en un video entrecortado por lo cual se comprobó, que, al utilizar la propia cámara de la Raspberry Pi, obtuvo un mayor rendimiento y fluidez debido a que su conexión es directa a este dispositivo y su calidad soporta al procesamiento.
- Para un mayor nivel de precisión al momento del reconocimiento facial se debe tomar en cuenta que los rostros deben de alinearse de manera frontal, así se le proporcionará al algoritmo mayor exactitud en el momento de comparar los rasgos faciales.
- Al analizar las pruebas de luminosidad se determinó que el sistema funciona de tal manera: Con la luz natural, ya que obtuvo el 93% de reconocimiento, al contrario de la luz nocturna que, debido al brillo que provoca la luz infrarroja de la cámara, así se obtuvo un descenso de reconocimiento del 10% obteniendo el 83% (determinado experimentalmente).
- En las pruebas de distancias del rostro, se comprobó que la resolución y calidad de la cámara es un aspecto importante para la detección, ya que a mayor distancia (2m) menor fue el reconocimiento, y a menor distancia (1m) mayor el reconocimiento, teniendo en cuenta que con 1 metro de distancia la efectividad de reconocimiento fue del 93% (determinado experimentalmente), así se comprobó que en esta distancia el sistema detecta mejor los puntos faciales para el reconocimiento.

RECOMENDACIONES

- Para un óptimo funcionamiento del entrenamiento y reconocimiento de rostros, se deben grabar videos con el mismo tamaño y calidad de imagen de la cámara que va a capturar el rostro.
- Para que el reconocimiento incremente la eficiencia se recomienda ocupar otro tipo de dispositivo con mayor procesamiento, para no tener problemas de videos entrecortados, por ejemplo, con una PC NUC.
- Para que el sistema de reconocimiento facial funcione de manera óptima se recomienda utilizar un modelo de Raspberry pi actual porque estos dispositivos poseen un mejor procesamiento y admiten el uso de memorias de mayor capacidad.
- Para mayor seguridad se recomienda utilizar una cámara 360 con sensor de movimiento, con este método se podrá monitorear y visualizar de una mejor manera al rostro de la persona que está alrededor del hogar.
- Se recomienda entrenar al sistema con varias posturas como: gestos, accesorios en el rostro, luminosidad y distancia ya que, al obtener todas estas características, ayudaría a la precisión del reconocimiento.

LISTA DE REFERENCIAS

Artículos académicos:

Cabrera, F., Granja, I., Moreno, D., Valle, P. (s.f.). *Procesamiento De Imágenes Para La Identificación De Personas Como Sistema De Seguridad En Zonas*

Domiciliarias. <https://knepublishing.com/index.php/KNEngineering/article/view/6233/11605>

Seguridad Y Alerta Para Vehiculos, Basado En Reconocimiento Facial Y Localización Gps, En Una Raspberry Pi B Plus. [Tesis de Ingeniería, Escuela Politécnica Nacional].

<https://bibdigital.epn.edu.ec/bitstream/15000/16667/1/CD-7270.pdf>

Lasse, R. (2018). Inteligencia artificial. Madrid – España. Editorial alienta.

https://static0planetadelibroscom.cdnstatics.com/libros_contenido_extra/40/39308_Inteligencia_artificial.pdf

Bibliografías:

Challenger-Pérez, I., Díaz-Ricardo, Y., & Becerra-García, R. A. (2014). El lenguaje de programación Python. *Ciencias Holguín*, 20(2), 1-13.

Chazallet, S. (2016). Python 3: los fundamentos del lenguaje. Ediciones ENI.

Espinosa, H. (Septiembre de 2016). *Diseño E Implementación De Un Sistema De*

Espinoza, D., Jorquera, P. (2015). Reconocimiento Facial. Chile-Valparaíso.

Universidad Católica de Valparaíso.

DuBois, P. (2008). MySQL. Pearson Education.

Fernández, A. (2013). Python 3 al descubierto. Alfaomega Grupo Editor.

Harmon, P., King, D. (1988). Sistemas de Expertos. Madrid-España. MonoComp, S.A

Hidalgo, L. (2014). *Inteligencia Artificial y Sistemas de Expertos*. Córdoba. Servicio de publicaciones de la Universidad de Córdoba.

Howse, J. (2013). *OpenCV computer vision with python*. Birmingham: Packt Publishing.

Jiménez, J. G. (1999). *Visión por Computador*. Paraninfo.

Llerena Izquierdo, J. (2020). *Codifica en Python*.

Ottado, G. (2015). *Reconocimiento de caras: Eigenfaces y Fisherfaces*.

Reconocimiento de caras: Eigenfaces y Fisherfaces.

Pico, B. O. P., Rondón, P., & Arguello, H. (2011). Sistema de reconocimiento facial basado en imágenes con color. *Revista UIS Ingenierías*, 10(2), 113-122.

Ríos, J. (2013). *Reconocimiento facial en tiempo real* (Bachelor's thesis).

Sevilla, J. L. (2014). *Instalando un Sistema de Detección de Intrusos Inalámbrico (WIDS) en Raspbian-I*.

Páginas web:

Llamas, L. (17 de noviembre de 2017). *Modelos y Características de Rasperry PI*. <https://www.luisllamas.es/modelos-de-raspberry-pi/>

Raspberry Pi Company <https://www.raspberrypi.com/>

Solé, R. (10 de septiembre de 2021). *Cuál es el mejor sistema operativo para la Rasperry Pi*. <https://www.profesionalreview.com/2021/09/10/sistema-operativo-raspberry-pi/>