

**Universidad Internacional de La Rioja (UNIR)**

**ESIT**

**Máster Universitario en Industria 4.0**

# Sistema de reconocimiento facial mediante inteligencia artificial para registro de asistencia de personal en campo

**Trabajo Fin de Máster**

**presentado por:** Villafuerte Carrillo, Ricardo Patricio

**Director/a:** Herrera Ruiz, Leticia

## Resumen

La revolución industrial 4.0 se encuentra en auge y muchas empresas se han dedicado a digitalizar sus procesos operativos, y las que no, deben hacerlo para mantenerse competitivas. El objetivo de esta investigación fue el generar una propuesta de implementación de un sistema de reconocimiento facial para el registro de asistencia de personal en campo basado en la IA. Para lograr esto, se implementaron dos etapas, en una, se evaluó las capacidades y requerimientos técnicos de la empresa en base a tecnologías disponibles y visitas al lugar de implementación y en otra, el diseño de un plan de implementación que incluyó las indicaciones técnicas y la programación necesaria. Se logró crear un plan de implementación que incluye además de las indicaciones la programación del algoritmo de IA requerido. Se alcanzó el objetivo de generar una guía que sirva a la empresa para la implementación del sistema de reconocimiento facial.

**Palabras clave:** Inteligencia artificial, Internet de las cosas, Sistemas de reconocimiento facial, Aprendizaje profundo

## Abstract

The 4.0 industrial revolution is on the rise and many companies have taken steps to digitize their operating processes, and those that do not need to do so in order to remain competitive. The objective of this research was to create a proposal for the implementation of a facial recognition system for the attendance record of personnel in the field based on AI. To achieve this, two stages were implemented: in one, the capabilities and technical requirements of the company were evaluated based on available technologies and visits to the place of implementation, and in another, the design of an implementation plan that included the technical indications and the necessary programming. It was possible to create an implementation plan that includes, in addition to the indications, the programming of the required AI algorithm. The objective of generating a guide that serves the company for the implementation of the facial recognition system was achieved.

**Keywords:** Artificial intelligence, Internet of things, Facial recognition systems, Deep learning

## Índice de contenidos

1.	Introducción .....	9
1.1.	Motivación .....	10
1.2.	Planteamiento del trabajo .....	11
1.3.	Estructura de capítulos .....	13
2.	Contexto y estado del arte .....	14
2.1.	Descripción general del contexto del proyecto .....	14
2.1.1.	Internet de las cosas y Big Data .....	15
2.1.2.	Arquitectura IoT .....	17
2.1.2.1.	Hardware del sistema .....	17
2.1.2.2.	Dispositivo de software .....	18
2.1.2.3.	Ruta de comunicación .....	19
2.1.2.4.	Plataforma .....	19
2.1.3.	SRF.....	20
2.1.3.1.	Flujo de trabajo de un sistema de reconocimiento facial .....	21
2.1.3.2.	Problemas con los algoritmos de reconocimiento facial .....	22
2.1.3.3.	Tipos de sistemas y métodos para reconocimiento facial .....	23
2.1.3.3.1.	Detección de rostros .....	23
2.1.3.4.	Algoritmos de reconocimiento facial.....	24
2.1.3.4.1.	Métodos basados en el conocimiento .....	24
2.1.3.4.2.	Comparación de plantillas.....	25
2.1.3.4.3.	Métodos basados en la apariencia .....	25
2.2.	Estado del arte.....	30
2.2.1.	El reconocimiento facial, aplicaciones.....	30
2.2.2.	Procedimiento para el reconocimiento facial .....	37
2.2.2.1.	Ubicación de puntos en el rostro.....	38

2.2.2.2.	Ubicación y detección de ojos y características de la piel.....	39
2.3.	Tecnologías relacionadas con el SRF .....	39
2.3.1.	Big Data.....	40
2.3.2.	Visión Artificial .....	41
2.3.3.	Machine Learning .....	42
2.3.4.	Deep Learning.....	42
2.4.	Conclusiones sobre el estado del arte.....	43
3.	Descripción general de la contribución del TFM.....	45
3.1.	Objetivos.....	45
3.2.	Metodología del trabajo .....	46
3.3.	Descripción general de las partes o componentes de la propuesta .....	47
3.3.1.	Descripción general .....	47
3.3.2.	Alcance y limitaciones.....	49
3.3.3.	Listado de participantes.....	50
3.3.4.	Tecnologías implicadas .....	50
3.3.5.	Arquitectura, componentes e integración de tecnologías.....	50
3.3.6.	Resultados esperados .....	50
3.3.7.	Presupuesto y retorno esperado de la inversión .....	51
4.	Desarrollo específico de la contribución del TFE .....	51
4.1.	Especificación de la arquitectura a emplearse:.....	51
4.1.1.	Elección de la topología correcta para poder realizar el sistema reconocimiento facial y registro. ....	52
4.1.2.	Descripción del componente de inteligencia artificial empleado.....	53
4.1.3.	Definir cada una de las partes del sistema.....	54
4.2.	Selección de la tecnología .....	54

4.2.1.	Elección del hardware de video.....	55
4.2.2.	Elección del hardware de procesamiento .....	56
4.2.3.	Elección del algoritmo para el entrenamiento del reconocimiento facial, basado en la inteligencia artificial.....	57
4.2.4.	Entrenamiento y testeo sobre el SRF .....	59
4.2.5.	Diseño de la base de datos .....	60
4.2.6.	Diseño del software (HMI).....	61
4.2.7.	Diseño de la estrategia para el Match entre el algoritmo de reconocimiento facial y el HMI .....	64
4.3.	Configuración del sistema de video empleado en el reconocimiento facial.....	65
4.3.1.	Configuración inicial y conexión entre sistema de captura de video y la base de datos local.....	66
4.3.2.	Programación del script para poder realizar el Deep learning con la base de datos de rostros establecida. ....	66
4.4.	Configuración de la plataforma IoT .....	67
4.4.1.	Registro de credenciales y Generación de usuarios.....	67
5.	Conclusiones y trabajo futuro .....	68
	Referencias bibliográficas.....	71
Anexo A.	Código implementado .....	82
	Índice de acrónimos .....	125

## Índice de Ilustraciones

Imagen 1. Internet de las cosas .....	15
Imagen 2. Diagrama operacional general asociado con el funcionamiento de un SRF.....	20
Imagen 3. Etapas del reconocimiento facial .....	35
Imagen 4. Áreas de aplicación para SRF.....	37
Imagen 5. Puntos seleccionados del modelo inicial .....	38
Imagen 6. Ecuaciones para los diferentes tonos de piel .....	39
Imagen 7. Cantidad de dispositivos conectados cada año.....	40
Imagen 8. Representación de capas neuronales.....	43
Imagen 9. Topología del SRF propuesto .....	53
Imagen 10. Estructura de las partes que conforman el SRF propuesto.....	54
Imagen 11. ECam 8000 Full definición alta 1080p .....	56
Imagen 12. Construcción de clasificadores en cascada usando boosting.....	58
Imagen 13. Características de Haar usadas en Viola-Jones.....	59
Imagen 14. Script específico para la conversión de las imágenes a gris.....	60
Imagen 15. Primera Interface.....	62
Imagen 16. Segunda Interface.....	62
Imagen 17. Tercera Interface .....	64
Imagen 18. Diagrama de los procesos involucrados en la estrategia para el Match.....	65

## Índice de tablas

Tabla 1. Comparativa de técnicas de reconocimiento biométrico .....	36
Tabla 2. Ventajas de la visión humana vs la artificial .....	41
Tabla 3. Características de Haar .....	59



## 1. INTRODUCCIÓN

Las nuevas demandas del mundo moderno, donde el acceso a la información es fácil en todas partes mediante el uso de Internet ha sido clave para encaminar la modernización del sector industrial desde diversos puntos de vista (Goncalves et al., 2017).

Partiendo de esta idea, la última revolución industrial, conocida como Industria 4.0, se sustenta en parte en la digitalización de los procesos y la información empresarial por medio de la posibilidad de interconexión de sensores, dispositivos y líneas de producción con personas a través de un sistema central (software) que pueda analizar la productividad en tiempo real, a partir de los datos recolectados de ellos, por lo cual, representa una fusión de varias tecnologías avanzadas bajo un mismo concepto, como lo son, el llamado Internet de las Cosas (IOT por sus siglas en ingles), Computación en la Nube, Integración de Sistemas, Simulación, Fabricación Aditiva, Autónomo Robots, Big Data, Realidad Aumentada, cada una de ellas con un rol bien definido para lograr esta idea.

En las empresas que asumen este concepto de industria 4.0, la digitalización de los diversos y más mínimos aspectos inherentes al proceso productivo, el personal autorizado puede interactuar en tiempo real con los equipos o con toda la información que en la empresa se genera (Morrar et al., 2017). Al digitalizar, la información está a un clic de distancia y se puede acceder tanto dentro como fuera de una empresa.

La tecnología mínima necesaria en el camino hacia la digitalización es IoT o IIoT si nos referimos al ámbito industrial, utilizada junto con un software como Application Programming Interface (API) capaz de recopilar datos, interpretarlos y procesarlos, y finalmente enviarlos de vuelta al equipo los nuevos requisitos, para que pueda cumplirlos. IIoT, es uno de los principales puentes de conexión entre las aplicaciones físicas y digitales, lo que lleva al concepto de I4.0 (Schwab, 2016).

Con las tecnologías de la nueva era causando revuelo en el mundo competitivo actual, las direcciones de RRHH en las empresas, también necesita alinear sus procesos con el panorama tecnológico que cambia rápidamente (Bondarouk & Brewster, 2016).

No pueden depender únicamente del enfoque tradicional, por lo que es necesario que aborden un concepto de Recursos humanos inteligentes 4.0, en el cual, las tecnologías de nueva generación dirigidas a los empleados tienen el potencial de transformar los procesos de recursos humanos y ofrecer varios beneficios a la organización (Shaw & Varghese, 2018; Sivathanu & Pillai, 2018).

De manera general, en la transición de la Industria 3.0 a la 4.0, se generan nuevas necesidades, entre las que se encuentra la mejora y optimización de los procesos con el objetivo de tener sistemas inteligentes y que se pueda generar la interacción hombre-máquina. El término Industria 4.0 abarca tecnologías que faciliten y vuelvan más eficientes los procesos.

Uno de los aspectos relacionados con los recursos humanos y la transición a las empresas 4.0 es precisamente el del control de asistencia del personal, el cual, hasta hace unos pocos años, se realizada casi de manera exclusiva por métodos físicos, poco o escasamente digitalizados e interconectados, pero que en la actualidad, debido a los avances en tecnologías como el del reconocimiento facial, es susceptible a ser modernizado, salvándose de esta manera un escaño más en el proceso de la digitalización empresarial.

## 1.1. MOTIVACIÓN

Unos de los objetivos básicos de toda empresa, es la de estar a la vanguardia en su campo, con lo cual se pueda incrementar la rentabilidad de esta por medio de la disminución de los costos asociados a la actividad productiva y el incremento de la eficacia productiva. Aunque son diversas las opciones que puede abordar una empresa para encaminarse en esta vía de la modernización, uno de los procesos más fáciles de modernizar es el del control de asistencia del personal, esto, porque dicho proceso de modernización, posiblemente sea el más económico de todos, y podría mejorar significativamente el tema del ausentismo laboral, con lo cual, se puede monitorear de manera más fiable la producción y el control de recursos económico que implica el pago de salarios a empleados que no cumplen su jornada laboral completa.

Sin embargo, aunque no se ha realizado un censo al respecto, puede que, en Ecuador, la gran mayoría de las empresas no hallan implementado esta sencilla solución tecnológica, entre estas, se encuentra la empresa ecuatoriana “3A Composites”.

Esta empresa, se encuentra en un área rural, por lo que el acceso a internet es muy limitado, además, el personal trabaja en campo y el sistema de monitoreo de asistencia de personal es manual, por medio de tarjetas de control, lo que implica que en ocasiones los empleados no cumplan con sus jornadas laborales adecuadamente, se engañe al departamento de recursos humanos al falsificar entradas y salidas del puesto de trabajo, y esto, con la consecuencia de la pérdida de dinero para empresa que paga un salario por una jornada laboral no cumplida.

De esta manera, se estima que la carencia de un sistema de registro que no pueda ser vulnerado por los empleados, es necesario para optimizar los procesos productivos, y reducir la pérdida de dinero en pago de jornadas laborales que nunca se realizaron.

En este sentido, los avances tecnológicos relacionados con el reconocimiento facial son la solución a esta situación. A pesar de la carencia de internet de manera continua, es posible la ejecución de un Sistema de Reconocimiento Facial (SRF) que almacene los registros en un hardware local y que luego, de manera periódica, pueda ser cargada a la nube para que sea analizada de manera adecuada por el personal de recursos humanos.

## 1.2. PLANTEAMIENTO DEL TRABAJO

El presente proyecto consiste en solventar la necesidad de una empresa ecuatoriana, denominada 3A Composites ubicada en la ciudad de Quevedo km 4 ½ vía a Valencia, la cual requiere realizar un eficiente sistema de registro de personal en campo.

El personal de esta empresa trabaja en una zona fuera de la cobertura de internet, por lo cual, el sistema propuesto debe trabajar de forma offline, en este caso, los datos deben poderse guardar en un almacenamiento local, y posteriormente, deben enviarse de manera periódica a la nube para su almacenamiento y análisis.

En base al problema previamente planteado, la propuesta del presente trabajo se orienta en el empleo de integración de diversas tecnologías actuales, tales como: dispositivos móviles,

computadores, software de reconocimiento facial, servidores y conexión a internet. Todas estas tecnologías pueden interconectarse con el departamento de nómina para el monitorear la asistencia del personal a su jornada laboral.

El modelo de implementación a seguir está basado en el concepto de cliente-servidor, ya que el dispositivo de detección captura la imagen la cual se enviará a través de la conexión disponible a un servidor de reconocimiento, lo cual, permitirá la identificación del individuo, a través de un proceso básico que implica, la capturar de la imagen, la detección de las caras, y el posterior envío de esta al servidor seleccionado y esperar la respuesta de la coincidencia.

Se plantea instalar un servidor que trabaje de forma pasiva a la espera de conexiones con los dispositivos enlazados, las tareas del servidor de reconocimiento serian; procesar la imagen recibida, realizar el reconocimiento y generar información, mejorar las características de datos y emitir una respuesta al cliente. Para el reconocimiento facial, existe una serie de software que podrían considerarse para su implementación estos son: FacePro, FaceFirst, KeyLemon y TrueKey.

Dado que no existe una conexión constante a internet, los datos recabados de los registros faciales y las respectivas identificaciones del personal deben poder ser guardados en dicho servidor, y luego de manera periódica, pueden reenviarse al departamento de recursos humanos para su respectivo análisis.

De manera general, el objetivo de este trabajo es el de presentar una propuesta viable para la implementación de un SRF basado en la inteligencia artificial, para el control de la asistencia del personal de la empresa 3A Composites, y que se sustente en la integración de hardware y software de fácil acceso para la empresa.

### 1.3. ESTRUCTURA DE CAPÍTULOS

Este trabajo está compuesto por los capítulos que se detallan a continuación:

- Capítulo 1.- Introducción: en este capítulo se presenta el problema de estudio previo a una breve contextualización del aporte en base a lo que plantea la transición de las empresas 3.0 a las 4.0- Se expone de manera general cual es la motivación del trabajo y se plantea la ruta básica de abordaje y solución del problema planteado.
- Capítulo 2.- Contexto y estado del arte: En este capítulo se describen conceptos importantes sobre la Industria 4.0. Se describen diversos proyectos relacionados con el SRF, el IIoT y Big Data en el ámbito Industrial que han permitido, mediante la unión de tecnologías, generar soluciones. Finalmente se emiten conclusiones sobre lo investigado con el fin de dar a conocer los detalles más relevantes que se ha generado durante el presente trabajo.
- Capítulo 3.- Descripción general de la contribución: En ese capítulo, se señala el objetivo general y los objetivos específicos. En esta etapa del proyecto se explican los pasos de la metodología aplicada para la generación de la propuesta de implementación.
- Capítulo 4.- Desarrollo específico de la contribución: En el presente capítulo se detalla claramente el proyecto y su desarrollo. Se indican primeramente los recursos con los que cuenta la empresa y en base a estas, las necesidades que deben subsanarse para lograr la correcta implementación, seguidamente se detallan los aspectos a implementar y las directrices específicas para lograr esto.
- Capítulo 5: Conclusiones y líneas de trabajo futuro: Se plantean las conclusiones alcanzadas en el trabajo y se presenta líneas de trabajo futuro para poder complementar la implementación del SRF.

- Anexos: Como anexos se incluyen las especificaciones técnicas de los dispositivos utilizados y los programas empleados para llevar a cabo el TFM.

## 2. CONTEXTO Y ESTADO DEL ARTE

En este capítulo se expondrá de forma general los temas teóricos relacionados con la presente investigación, así mismo se describirán abordajes similares y sus aportes a nivel empresarial y social, tecnologías habilitadoras y la propuesta de recursos tecnológicos empleados.

### 2.1. DESCRIPCIÓN GENERAL DEL CONTEXTO DEL PROYECTO

La última revolución industrial, la 4.0, se encuentra en pleno desarrollo, al respecto, Perasso (2016), explica que esta no está definida por el conjunto de las tecnologías aplicables en la modernización de todos los aspectos de la empresa, sino que se define por la transición hacia un conjunto de nuevos sistemas cimentados y estructurados sobre la estructura tecnológica que pretenden reemplazar.

En este punto de vista, se manifiesta que la Industria 4.0 es el cambio a nuevas tecnologías, pero en base a los sistemas ya establecidos de la Industria 3.0.

Cuando se habla de las industrias 4.0, se hace referencia a una nueva etapa de la revolución industrial, caracterizada por la interconexión entre equipos y centros de datos, así como la automatización y sistematización de diversidad de procesos, tanto productivos como operativos (EPICOR, 2022).

La Industria 4.0 surge de la fusión de diversas tecnologías en un único ecosistema digital, por ello, los términos 'digitalización' e 'Industria 4.0' a menudo se emplean de forma asociada, pese a que la digitalización ya ha causado un gran efecto en gran parte de la sociedad, tiene mucho mayor impacto en el sector Industrial.

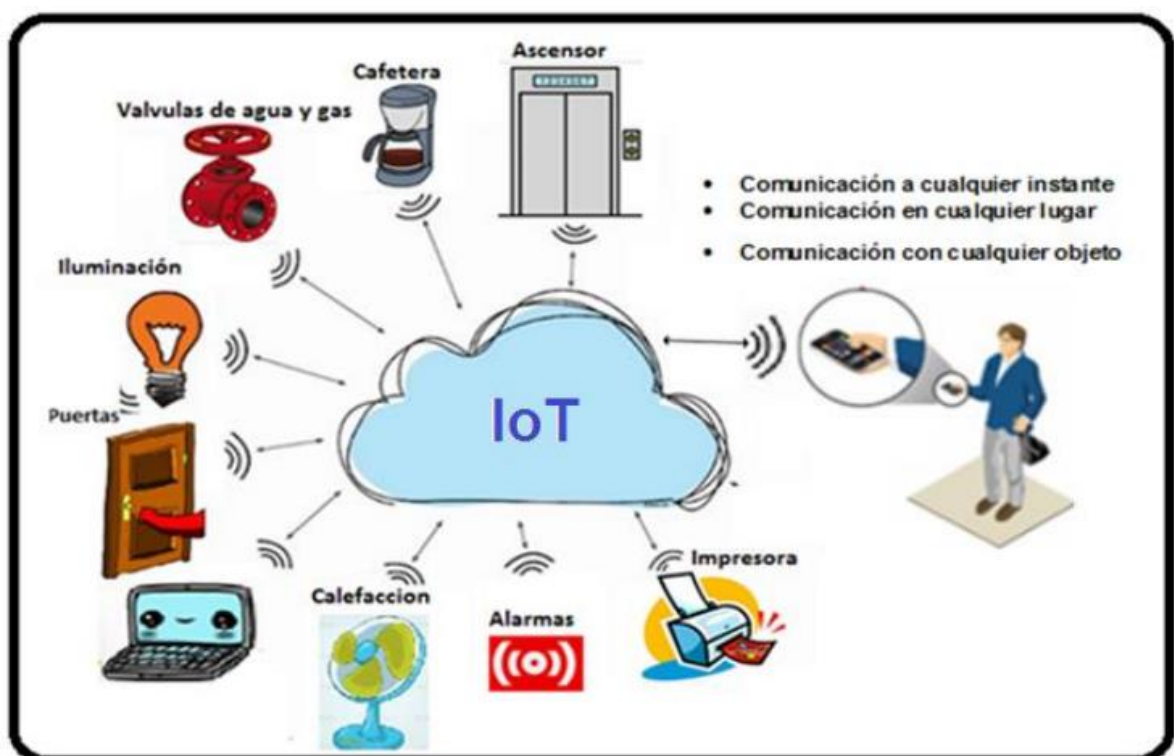
En relación con las Industrias 4.0, el proceso de digitalización está asociado con la conectividad y la interacción entre máquinas y humanos, esperando una transformación en el diseño,

operación y sistema de producción Industrial. Un ejemplo de mejora en los procesos operativos, es la toma de decisiones y reconocimiento de imágenes a través de un análisis de Big Data, lo cual, contribuye a la mejora de la productividad (Carlos Suárez, 2019).

### 2.1.1. Internet de las cosas y Big Data

El IoT (*Internet of Things*) se ha incorporado en la vida cotidiana poco a poco, lo cual ha traído grandes beneficios para la sociedad y la Industria, por ejemplo, disponer de infraestructuras interconectadas, procesos más eficientes, generación de empleo, reducción de costes e incremento de ganancias (Imagen 1). De acuerdo a las investigaciones realizadas por la empresa Cisco®, hasta el año 2020, habían 50.000 millones de dispositivos conectados, número que irá creciendo con el paso de los años (Ramírez Madrid & Rodríguez Hernández, 2016).

Imagen 1. *Internet de las cosas*



Fuente: Ramírez y Rodríguez (2016)

La gran cantidad de datos que se manejan debido a la conectividad de los dispositivos se lo maneja a través de Big Data, la cual junto con la inteligencia artificial y los sistemas de computación permite crear una nueva información, conocimiento clave para que las empresas vuelvan sus procesos mucho más eficientes y competitivas.

Por su parte, el almacenar grandes cantidades de información existe desde hace algunos años, pero el concepto de *Big Data* empieza a cobrar impulso desde el año 2000 y hace referencia al manejo de una gran cantidad de datos y a las velocidades de transmisión de estos en la red, los cuales, son prácticamente imposibles de procesarlos con un método tradicional (Ramírez Madrid & Rodríguez Hernández, 2016).

Como es posible apreciar, el IoT se está convirtiendo en un paso hacia la revolución en el mundo moderno y en un hito logrado en el campo de la inteligencia artificial (Wanasinghe et al., 2020). Desde electrodomésticos hasta vehículos se puede controlar mediante el uso de teléfonos inteligentes, ya que IoT es una gran red que consta de dispositivos conectados a ella.

Los dispositivos conectados recopilan datos y comparten cómo están siendo operados y realizan las tareas asignadas (Sahni et al., 2017). Todo esto es gracias a los sensores, los mismos, están integrados en nuestros teléfonos móviles y en varios otros aparatos eléctricos y dispositivos basados en señales que se conectarán a la red de IoT. Un ecosistema de IoT involucra dispositivos inteligentes, los cuales se encuentran habilitados para la web, además presentan hardware, procesador y sensores para poder enviar y procesar sobre los datos que obtienen (Gaber et al., 2019).

Los dispositivos de IoT revelan los datos del sensor que ensamblan al involucrar una entrada de IoT u otro dispositivo de borde donde los datos se envían a la nube para su análisis, luego, estos dispositivos se comunican con otros que también están conectados y actúan en base a la información que obtienen entre sí. Sin exista la mano del hombre, los dispositivos hacen la mayor parte del trabajo, aunque las personas pueden interactuar con ellos, por ejemplo, para configurar, instruir o acceder a datos (Zhou et al., 2019).

Los protocolos de comunicación, redes y conectividad utilizados con estos dispositivos habilitados para la web dependen en su mayor parte de las aplicaciones específicas de IoT implementadas. IoT también puede ser capaz de aprendizaje automático, lo que se denomina



inteligencia artificial que puede ayudar a que los procesos de datos sean más convenientes y mucho más dinámicos (Hussain et al., 2020).

Dado que IoT crea y analiza grandes cantidades de datos, es un motor importante de los proyectos de análisis de big data (Marjani et al., 2017). En particular, puede trabajar con grandes cantidades de datos en tiempo real (Akbar et al., 2018). A través de varios dispositivos IoT, es posible visualizar el desempeño de todos los empleados, así como avanzar en las operaciones en todas las ubicaciones.

### **2.1.2. Arquitectura IoT**

Diferentes estructuras han sido propuestas por diferentes especialistas; sin embargo, no existe una arquitectura única que esté estandarizada por ninguna organización para IoT, lo cual es aceptado en su mayor parte (Bader et al., 2019). Es una tarea difícil delinear la tecnología que controla la gran cantidad de inventos y descubrimientos que nos rodean, sin embargo, es posible diferenciarlos en torno a cuatro ilustrarlo cuatro grupos diferentes, todos, involucrados en la creación del IoT (Fortino et al., 2018), a saber:

- Hardware del sistema
- Dispositivo de software
- Vía de comunicación
- Plataforma

#### **2.1.2.1. Hardware del sistema**

Estos forman el cuerpo físico del sistema; son los componentes a los que se hace referencia como "cosas" en el título de IoT («IoT Hardware Development Platforms», 2017). Actúan como puerta de entrada entre la realidad y el mundo digital. Se observan de muchas formas, como un sensor en un teléfono o un automóvil de conducción automática (Petrov et al., 2018).

Cualquier dispositivo físico cableado que pueda volverse "inteligente" al agregar un chip, un sensor o una placa para controlarlo a través de señales se denomina hardware IoT. Estos dispositivos se utilizan mucho para recopilar datos sobre sus respectivos campos. Muchos sensores y microprocesadores se trazan dentro de los dispositivos para que sean inteligentes, pero surge un pequeño problema que puede solucionarse más adelante en el futuro, es decir, el alto costo de compra, generalmente se requiere una gran cantidad de dinero para adquirirlos (Lee, 2017). Otro problema es la compatibilidad del dispositivo; muchos no son compatibles con los sensores recién inventados pero esta situación, tiende a reducirse con el tiempo (Efremov et al., 2015).

#### 2.1.2.2. Dispositivo de software

El segundo paso hacia la construcción de IoT es el dispositivo de software (Rahman et al., 2018). La palabra 'inteligente' puede usarse al referirse a este componente ya que funciona como el cerebro del sistema IoT (Zhu et al., 2010).

La implementación de datos y comunicación en la nube recae en este grupo. Son responsables de la integración de dispositivos y la recopilación de datos del sistema, y el análisis de estos dentro de la nube realizado por el software IoT. También permite que el usuario interactúe con el sistema convirtiendo los datos en un nivel de aplicación capaz de ser visualizados por el usuario (Zhu et al., 2010).

Los datos recibidos de los dispositivos están en forma analógica y se convierten en flujos digitales mediante Sistemas de Adquisición de Datos (DAS por sus siglas en inglés) al realizar todas las operaciones analíticas y teóricas lógicas. Después de procesar los datos antes de ingresar al centro de datos, se repite el mismo proceso de operaciones para hacerlo más exacto e informativo (Zhu et al., 2010).

### 2.1.2.3. Ruta de comunicación

Es la tercera base del sistema IoT (Hamrioui & Lorenz, 2017). Después de completar la configuración del sistema de hardware y el software del dispositivo y prepararlo, llega la etapa de comunicación, que es responsable de proporcionar la ruta para la transmisión de los dispositivos de hardware inteligentes recién generados.

El sistema de comunicación se basa en hardware y software, y existen varias opciones incluyen: Wi-Fi, LAN, Bluetooth, o controles satelitales. Cada uno de ellos es apto para la transmisión si tienen instalado su respectivo hardware y software. Cada chip de comunicación tiene su propia configuración y estructura en la nube, y no solo decidirá la forma de envío, sino también cómo se administran y se comunican los dispositivos (Rosas et al., 2017).

### 2.1.2.4. Plataforma

Este es el cuarto y último del sistema IoT, hace referencia a la plataforma, donde se almacenan todos los datos recopilados y procesados anteriores. Desde aquí, los datos se presentan al usuario en un formulario fácil de usar. En esta sección, los datos se clasifican y dividen según su propósito, se etiquetan y almacenan en diferentes archivos (Benammar et al., 2018).

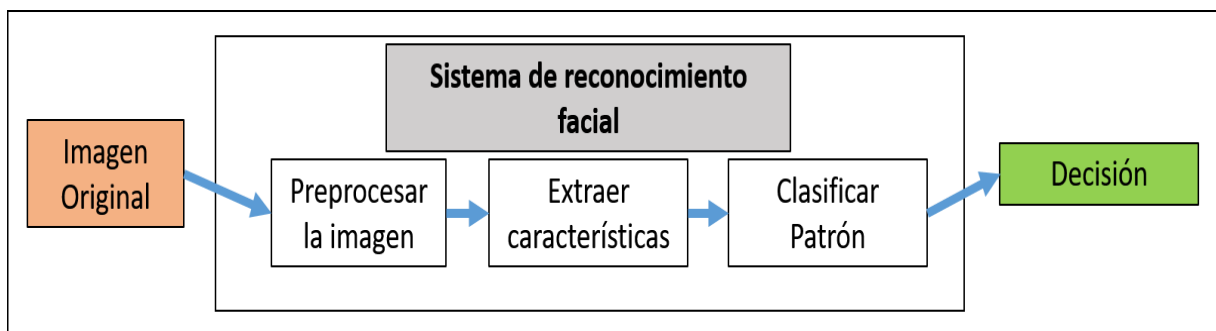
A través de esto, el usuario podría elegir la información exacta. Hay muchas empresas en el mercado que ofrecen plataformas dependiendo de las características y la estructura de la plataforma (Hejazi et al., 2018). Las opciones pueden variar según el costo y los requisitos. Hay muchas empresas que ofrecen plataformas, que pueden instalarse de manera local usando un dispositivo de almacenamiento adicional o empleando almacenamiento en la nube.

### 2.1.3. SRF

En las tecnologías antes mencionadas, se encuentra la asociada con el reconocimiento facial mediante el empleo de la inteligencia artificial. Durante los últimos años, el reconocimiento facial se ha convertido en una de las aplicaciones más relevantes en el campo de la biometría mediante el procesamiento y reconocimiento de patrones (Muskopf-Stone et al., 2022).

Este proceso de reconocimiento de los rasgos faciales por un sistema computacional debe tener en cuenta primero la detección facial y luego el reconocimiento del individuo de acuerdo a una base de datos previamente creada, esta última etapa, implica el reconocimiento de patrones faciales y el sistema debe ser capaz de clasificar e identificar las imágenes de la “cara” dependiendo de sus características propias (Hernández, 2010). En la Imagen 2 se observa un diagrama general del proceso de reconocimiento facial.

**Imagen 2.** Diagrama operacional general asociado con el funcionamiento de un SRF.



Fuente: Hernández (2010)

### 2.1.3.1. Flujo de trabajo de un sistema de reconocimiento facial

Los pasos básicos involucrados en el reconocimiento facial son los siguientes (López-Sánchez, 2022):

- Captura de una imagen: la imagen capturada también se conoce como imagen de prueba. La imagen puede capturarse a través de una cámara fija o de circuito cerrado de televisión, con o sin el conocimiento del sujeto.
- Detección de rostros: en este paso, se detecta el rostro del sujeto de toda la imagen capturada.
- Extracción de características: en este paso, a partir del rostro detectado, se extraen las características específicas y únicas para unirlos con las imágenes correspondientes en la base de datos. En este paso, se genera una plantilla de cara.
- Emparejamiento: Se realiza el emparejamiento de la imagen de la plantilla con las imágenes de la base de datos. Si los rasgos faciales coinciden, se realiza la verificación de la persona.
- Verificación/identificación: Estas son las dos tareas principales del reconocimiento facial. El último paso consiste en identificar al individuo. Si el objetivo es la verificación, se realiza una coincidencia 1:1 y, para la identificación, se realiza una coincidencia 1: N.

Después de la detección de rostros, la identificación de la persona es el último paso en los sistemas de reconocimiento de rostros, y se realiza una coincidencia 1:1 para permitir la identificación de personas.

En un estudio, se propuso el sistema de reconocimiento facial para identificar a una persona en el contexto de un material didáctico (Nasution et al., 2014). El sistema se basó en la comunicación entre un servidor estático y un dispositivo móvil, y proporcionó una tasa de precisión de >95 %. Los métodos proporcionados por Open Source Computer Vision Library se utilizaron para el reconocimiento de detección de rostros. La identificación de personas se basa en las capacidades de detección de rostros de los algoritmos de reconocimiento utilizados.

#### 2.1.3.2. Problemas con los algoritmos de reconocimiento facial

Desde que se desarrolló, el área de reconocimiento facial a menudo ha tenido que superar obstáculos, existiendo varios factores que hacen que el reconocimiento preciso sea una tarea desafiante, como se describe a continuación:

- **Iluminación:** las condiciones variables de iluminación pueden tener una serie de efectos diferentes en la imagen facial de una persona (M. Singh & Arora, 2016).
- **Pose:** el problema de la pose se ha identificado como un tema clave en el reconocimiento facial y ha sido objeto de mucho interés entre las investigaciones. comunidad en las últimas décadas (Zhang & Gao, 2009).
- **Oclusión:** entre muchos problemas asociados con el reconocimiento facial preciso, el manejo de la oclusión facial es uno de los principales problemas (Sharma et al., 2013). Cuando se ocluye cualquier parte de la cara, las características faciales no se pueden ver por completo y, por lo tanto, la autenticación por parte del sistema de reconocimiento facial está en riesgo.
- **Expresiones:** Los rostros humanos siempre contienen expresiones de un tipo u otro, generalmente sustentadas por el estado emocional del individuo. Los sistemas de reconocimiento facial también se ven afectados por las expresiones humanas, que resultan del movimiento de los músculos faciales, lo que provoca cambios en las imágenes faciales (Samadiani et al., 2019). Sin embargo, algunos sistemas de reconocimiento facial no pueden procesar

diferentes expresiones de la misma persona, lo que significa que un problema puede surgir con la identificación (Li & Da, 2012).

- **Cabello:** El cabello a menudo cubre la frente. Por lo tanto, en la mayoría de los sistemas de reconocimiento facial, las imágenes de la base de datos tienen cabello neutralizado para evitar que actúe como una barrera para el reconocimiento de rostros. Se están realizando investigaciones para tratar de resolver este problema (Marqués, 2010).
- **Edad:** con el avance de la edad, las características faciales pueden cambiar enormemente. Sin embargo, esta área de investigación aún se encuentra en una etapa inicial y aún queda mucho por hacer (Abate et al., 2007).

### 2.1.3.3. Tipos de sistemas y métodos para reconocimiento facial

Existen dos categorías de sistemas de reconocimiento facial, estos son, los sistemas supervisados y los no supervisados. La diferencia entre ellos es que el primero conoce a quien corresponde la imagen previamente y asigna una etiqueta de clasificación, mientras que lo dicho no sucede con el no supervisado (Morcillo, 2014).

#### 2.1.3.3.1. Detección de rostros

La detección facial forma parte de las tecnologías de visión computarizada que permite determinar el tamaño de rostros en videos o en imágenes (Kumar et al., 2019). Es una tarea que representa dificultades para la computadora debido a la gran cantidad de variables como: posiciones del rostro, presencia de elementos estructurales (bigote), expresiones faciales, oclusiones (gorra, lentes), cantidad de luz en el ambiente, por lo cual, el sistema debe ser capaz de detectar rostros independientemente de los factores antes mencionados (Espinoza & Jorquera, 2015).

Hasta la fecha, dada las dificultades antes expuestas, se sabe de la existencia de varios métodos que permiten que el reconocimiento facial sea realizado. Al respecto, Espinoza y

Jorquera (2015), explican los fundamentos de los mismos, en estos, exponen que en el método basados en conocimiento, se consideran las distancias existentes en las características ergonómica y faciales humanas, por ejemplo, la distancia entre ojos-nariz, nariz-labios, entre otros.

También explican que, en el Métodos basados en las características invariantes de las personas, las identificaciones se orientan en la evaluación de las características faciales que no son afectadas por los cambios de luz, como por ejemplo, las cejas, la textura de la piel, la nariz, la línea de cabello, etc.

Con este método, se detecta uno de estos componentes y se establece un modelo estadístico con el cual se verifica la existencia del rostro en la base de datos con la que se esté trabajando.

Por su parte, en el método basado en patrones, se considera las relaciones entre una imagen y un patrón ya definido mediante el cual se identifican las características del rostro escaneado.

Así mismo, indican que, en el método de identificación basado en apariencias, se requiere el entrenamiento del sistema mediante la incorporación de imágenes previamente tomadas de la persona. La diferencia de este con el método basado en moldes, los patrones son determinados por el aprendizaje obtenido durante el entrenamiento por medio de imágenes.

#### 2.1.3.4. Algoritmos de reconocimiento facial

En las últimas décadas se han desarrollado varios tipos de algoritmos de reconocimiento facial. Algunos de los métodos comúnmente utilizados se muestran a continuación:

##### 2.1.3.4.1. *Métodos basados en el conocimiento*

Estos también son ampliamente conocidos como los "métodos basados en reglas". En estos métodos, se hacen esfuerzos para capturar y aprender de las caras y luego, posteriormente, se traducen en un conjunto de reglas. Por ejemplo, hay rasgos faciales que están ubicados simétricamente y también áreas en la cara que difieren en intensidad. La mayor dificultad con



los métodos basados en el conocimiento está en la formulación de un conjunto apropiado de reglas, que no debe ser ni demasiado general ni demasiado detallada (Marqués, 2010).

Una posible solución para superar estos problemas es desarrollar métodos jerárquicos basados en el conocimiento. Sin embargo, en el caso de imágenes complejas, este método no puede funcionar de manera eficiente. Se debe emplear un método robusto para superar esta dificultad (Marqués, 2010).

#### 2.1.3.4.2. *Comparación de plantillas*

Este es un intento inicial de identificar rostros teniendo en cuenta solo ciertas regiones representadas en las plantillas. Los píxeles de la imagen de entrada se comparan con una imagen de plantilla utilizando una medida métrica como la distancia euclidiana (Zhang & Gao, 2009).

En los intentos iniciales, se seleccionó un conjunto de cuatro características (ojos, nariz, boca y cara completa) de una plantilla y se aplicó a todas las caras disponibles (Brunelli & Poggio, 1993). En la comparación de plantillas, el área de la imagen de entrada se compara con las imágenes en la base de datos para cada plantilla. Finalmente, se calculan las puntuaciones de coincidencia y se toma la decisión de reconocimiento en base a ellas. Se han informado altas tasas de precisión del 90 %, y una de las ventajas clave de este método es que su implementación es simple, aunque es inapropiado para variaciones en la pose y la iluminación.

#### 2.1.3.4.3. *Métodos basados en la apariencia*

Este método es una forma de coincidencia de plantillas en la que las plantillas se toman del conjunto de ejemplos en las imágenes. En general, este método se basa en técnicas de estadística y aprendizaje automático para encontrar las características específicas de las imágenes de rostros (Marqués, 2010). Algunos de los algoritmos más empleados en este método son los siguientes:

- **Algoritmo PCA de métodos basados en Eigenface:** El método más conocido y ampliamente aceptado para el reconocimiento de rostros se conoce como el método Karhunen-Loeve. Es el método más estudiado para el reconocimiento facial, siendo su principal utilidad la reducción de la dimensionalidad de la imagen. Primero se utilizó para el reconocimiento facial y luego se aplicó a la reconstrucción facial. Para que se utilice PCA, la plantilla y las imágenes de la galería deben tener el mismo tamaño y primero deben normalizarse para alinearse con los ojos y la boca de los sujetos en las imágenes. Luego, utilizando los conceptos básicos de compresión de datos, se reducen las dimensiones de los datos. En este paso, la información que no es útil se reduce y la estructura de la cara se reduce con precisión en componentes no correlacionados, conocidos como Eigenfaces, que representan una base que tiene un espacio de baja dimensionalidad (Abate et al., 2007).

Cada imagen facial puede representarse como una suma ponderada (vector de características) de las caras propias, que se almacenan en una matriz unidimensional. Luego, se realiza la comparación entre la galería y las imágenes de la sonda en función de la distancia entre los vectores de características. La principal ventaja de esta técnica es que reduce los datos por 1000 veces. Además, este método es muy rápido, ya que se usa solo para entrenar la muestra (Abate et al., 2007). Sin embargo, este enfoque requiere una imagen frontal completa de la cara del sujeto y, en situaciones en tiempo real, esto rara vez sucede. Además, el escaso poder de discriminación dentro de la clase es uno de los mayores problemas del método PCA.

- **Análisis discriminante lineal:** El análisis discriminante lineal (LDA), también conocido como análisis discriminante de Fisher, es la técnica de reducción de la dimensionalidad. Entre los métodos basados en apariencia, LDA se usa para la selección de características. Las limitaciones del método PCA se superan con este método. De todos los métodos basados en la apariencia, LDA es la técnica más utilizada y comprensible para la selección de características. Similar a PCA, se basa en la distancia euclidiana. Es un método supervisado y, a diferencia de PCA, utiliza la información de la etiqueta para mejorar la separabilidad entre diferentes clases (Zhang & Gao, 2009).

Además, también tiene como objetivo minimizar la variación dentro de la clase (Mankar & Bhele, 2012). En varios enfoques de reconocimiento facial dependientes de LDA, inicialmente, PCA se utiliza para la reducción de

dimensionalidad, y luego se aplica LDA para maximizar el poder de discriminación de la selección de características.

Este método ha demostrado su valía en el reconocimiento, seguimiento y detección de rostros ya que no aborda las dependencias estadísticas de alto orden, por ejemplo, la relación entre múltiples píxeles (Agrawal & Singh, 2015). Sin embargo, LDA sufre el problema del tamaño de muestra pequeño. De hecho, debería poseer muestras grandes para una buena discriminación de la extracción de características. Por lo tanto, la extracción de características discriminatorias es deficiente.

- **Vectores comunes discriminantes:** Este método es básicamente el siguiente paso a LDA. El concepto detrás de los vectores comunes discriminantes es que apunta a recopilar las similitudes entre los elementos que pertenecen a la misma clase y minimizar las diferencias. Por lo tanto, de esta manera, cada clase se puede representar mediante un vector común, que se calcula desde dentro de la matriz de dispersión. Para el reconocimiento de un rostro desconocido, se calcula el vector de características y se asocia a la clase que tiene el vector común más cercano (Abate et al., 2007).
- **Análisis de componentes independientes:** Este método es adecuado para encontrar factores a partir de datos estadísticos multivariados. En los sistemas que ya existían, había una brecha, ya que no eran adecuados para imágenes faciales que poseían diferentes condiciones, como orientaciones faciales, poses y condiciones de iluminación variables (Cevikalp et al., 2005).

Este enfoque busca componentes que no sean gaussianos y sean estadísticamente independientes. En el análisis de componentes independientes (ICA), antes de calcular los componentes independientes, cada imagen facial se transforma inicialmente en un vector. Este método se ha utilizado con varios otros modelos para el reconocimiento facial (Alfalou & Brosseau, 2011).

ICA ha demostrado ser un método más poderoso que PCA, ya que tiene como objetivo proporcionar una descomposición y representación de imágenes independientes, a diferencia de PCA, donde los componentes de la imagen facial se dividen en componentes individuales no correlacionados. Se ha demostrado el análisis de componentes principales independientes/análisis de componentes independientes (IPCA-ICA), y demostró ser muy eficiente (Kar et al., 2011).

IPCA-ICA también tuvo una tasa de éxito más alta en comparación con varios otros métodos, como los métodos Eigenface, Fisher face y Fast ICA (Dagher & Nachar, 2006). ICA se considera una generalización de PCA, pero ofrece varias ventajas sobre PCA, que incluyen una caracterización superior de los datos y la captura de factores discriminatorios al considerar estadísticas de alto orden. Para la identidad facial, sin duda, ICA supera a PCA, pero entre ICA, la arquitectura ICA II ha mejorado el rendimiento de la arquitectura ICA I (Draper et al., 2003).

- **Técnica de concordancia de gráficos de racimos elásticos:** Esta técnica se basa en estructuras de enlaces dinámicos. En la coincidencia de gráfico de grupo elástico (EBGM), se genera un gráfico para cada cara individual. En la cara, se seleccionan nodos específicos, conocidos como puntos de referencia. Estos se describen mediante componentes de longitud de onda (Alfalou & Brosseau, 2011).

Cada punto fiduciario está etiquetado con respuestas de filtro Gabor aplicadas a una ventana alrededor de cada punto fiduciario. Dichos gráficos se combinan en una pila conocida como gráfico de grupo de caras. Una vez que se genera, se pueden generar nuevas imágenes de rostros utilizando EBGM (Jafri & Arabnia, 2009).

En la fase de reconocimiento, simplemente se comparan dos gráficos de imágenes. La ventaja de EBGM sobre los métodos de análisis lineal ya existentes es que tiene en cuenta varias características no lineales, como la iluminación, la pose y la expresión. La tasa de reconocimiento puede alcanzar hasta el 98 % con este método, y también se han proporcionado varias variaciones de este enfoque (Dagher & Nachar, 2006).

La localización de puntos de referencia con precisión es bastante difícil con este método. Esto puede facilitarse combinando PCA y LDA con EBGM (Cevikalp et al., 2005). Además, requiere tamaños de imagen facial grandes, que pueden ser difíciles de obtener en situaciones del mundo real. Por ejemplo, en los sistemas de vigilancia, el área de la cara en la imagen es bastante pequeña (Zhao & Chellappa, 2002). En un método propuesto recientemente, no se requiere la colocación manual del gráfico. Esto se hace mediante el uso de modelos paramétricos basados en plantillas deformables.

- **Redes neuronales:** La solución a los problemas encontrados en los métodos lineales fue proporcionada por varios métodos no lineales como las redes neuronales. Básicamente, se considera una red con una neurona en cada píxel. El paso de

extracción de características es más eficiente que los métodos lineales debido a la no linealidad de la red. Usando este método, se informó una tasa de precisión del 96,2%. El tiempo de entrenamiento es más en comparación con el tiempo de clasificación (Rowley et al., 1998).

Las redes neuronales se han combinado con diferentes modelos, como los modelos ocultos de Markov (HMM) y los filtros Gabor. Se utilizó un método semisupervisado para reconocer el rostro humano (Intrator et al., 1996). Los investigadores utilizaron métodos no supervisados para la extracción de características y métodos supervisados para encontrar aquellas características que pueden reducir el error de clasificación. Utilizaron redes neuronales de avance para la clasificación.

La red neuronal probabilística basada en decisiones se modeló para tres aspectos diferentes: detección de rostros, localización de ojos y reconocimiento de rostros (Lin et al., 1997). Se introdujo un enfoque híbrido en el que PCA se usó inicialmente para la extracción de características y luego la salida se usó como entrada en un sistema radial. red neuronal de función básica(Er et al., 2002).

Este método funcionó de manera eficiente para el reconocimiento facial. Los autores se encontraron con el problema del sobreajuste. Los investigadores propusieron una nueva clase de red neuronal convolucional. En esto, las células de procesamiento son neuronas inhibitorias de derivación (Tivive & Bouzerdoum, 2003).

Anteriormente, estas neuronas se han utilizado en la arquitectura de avance convencional para la clasificación y la regresión no lineal. Se había demostrado que eran más poderosos que  $n$  perceptrones multicapa difusos (MLP)(Kar et al., 2011). Se propuso un enfoque para el reconocimiento facial que implementa el concepto de un MLP difuso. El objetivo principal del enfoque era capturar superficies de decisión en variedades no lineales. Este trabajo no podía ser realizado por un MLP simple y, por lo tanto, se combinó con la red neuronal (Li & Da, 2012).

La ventaja básica y más importante de la red neuronal es su capacidad para capturar la clase compleja de patrones faciales. Sin embargo, cuando aumenta el número de clases, se vuelve difícil implementar redes neuronales. Además, este enfoque no es adecuado para un solo modelo (AL-Allaf, 2014).

- **Aprendizaje profundo e inteligencia artificial:** En este ámbito tecnológico, las técnicas de aprendizaje automático, aprendizaje profundo e inteligencia artificial han

impactado y afectado posteriormente la amplia gama de servicios y operaciones lógicas del día a día. Incluso en el dominio del reconocimiento facial, posteriormente han demostrado tener un impacto indispensable.

En un estudio publicado recientemente por Chang y col. (2019), se desarrolló un algoritmo para determinar la experiencia de los clientes en un restaurante sin personal. Sobre la base de las expresiones, se califican la comida y el entorno. El sistema evaluado en dicho estudio utilizó los modelos de Red Neuronal Convolucional (CNN).

- **Máquinas de vectores soporte (SVM):** Las SVM, son generalmente clasificadores lineales que amplían el margen entre el hiperplano de decisión y el conjunto de entrenamiento. Esto se logra encontrando un hiperplano que separe y maximice la distancia de cualquiera de la clase al hiperplano, donde un conjunto de puntos pertenecen a dos clases diferentes (Kakade, 2016). Por lo tanto, un hiperplano óptimo debería funcionar minimizando el error de clasificación de la prueba no vista patrones. En primer lugar, la extracción de funciones se realiza mediante PCA y, a continuación, la discriminación entre las funciones se realiza mediante SVM. La ventaja de las SVM sobre otros enfoques clásicos radica en el hecho de que pueden lograr un rendimiento mejorado.

## 2.2. ESTADO DEL ARTE.

En la presente sección se realiza una recuperación bibliográfica de los aspectos reportados en la literatura científica que están relacionados con el tema de estudio, así como, se incluyen referencias a investigaciones asociadas a cada uno de los temas planteados.

### 2.2.1. El reconocimiento facial, aplicaciones

En los últimos años, la literatura científica sobre el reconocimiento facial en el entorno empresarial ha incrementado en gran medida, y los ámbitos empresariales que abarcan también son extensos, e incluye aspectos como manufactura, seguridad, vigilancia, turismo e

inclusive tocan el área de la domótica (Matias et al., 2010; Hernández, 2010; Gualdrón & Suárez, 2014; Espinoza Olguín & Jorquera Guillén, 2015; Rivas Perea et al., 2016; Garcés Nuñez, 2017; Aldana Porras, 2018; Bastidas Gavilanes, 2018; Bravo et al., 2018; González-Muñiz, 2018; xmart, 2019; Chacua Criollo, 2019; Torres et al., 2021; Carrión & Yandry, 2022).

Una de las aplicaciones del reconocimiento facial se encuentra en los ambientes no colaborativos donde existen diversos problemas de iluminación. En este tipo de ambientes, las personas pueden estar siendo identificadas de forma involuntaria, es decir sin cooperación a su identificación. Para poder combatir estas situaciones se emplea la teoría de redes neuronales artificiales y la lógica difusa, los cuales han sido utilizados para resolver múltiples tareas de reconocimiento y obteniendo muy buenos resultados (Rivas Perea et al., 2016).

Cuando la identificación facial es necesaria en eventos públicos a gran escala, donde la iluminación es un factor clave para la correcta identificación, por lo general, las cámaras son ubicadas en los sitios donde se expenden alimentos, el sistema en estos casos, al identificar el rostro de una persona en particular puede sugerir los alimentos que podrían ser deseados por la persona identificada, debido a que luego de reconocer la imagen, revisan el historial de compras o preferencias al que tenga acceso el sistema y se presenta lo que habitualmente la persona solicita para comer (Orange, 2021).

Hoy en día las empresas trabajan constantemente para lograr un equilibrio entre su inversión y las demandas laborales por parte de sus colaboradores, por lo cual, se está adoptando la implementación de una aplicación como es *XmartClock* la cual se puede instalar en un smartphone o Tablet y de esta manera controlar el correcto cumplimiento del horario laboral y pago de roles (Duocom Europe, 2022).

Entre los beneficios que brinda dicha aplicación se encuentra la gestión efectiva de forma remota del tiempo de trabajo, ahorro de tiempo instalando la aplicación de control de asistencia laboral, disminución de cotes y ahorro de dinero, disminuye el ausentismo y la impuntualidad en el trabajo, potenciar la imagen y la transparencia como empresa, controlar el contexto exterior en la compañía, el funcionamiento de app es sencillo (xmart, 2019).

El SRF también se aplica en tiempo real, en especial cuando se requiere realizar vigilancia. Por ejemplo, mediante la app *ImaGeplus*, la cual es empleada por la UPC de la ciudad de Barcelona-España (Universitat Politècnica de Catalunya, 2013), la cual consiste en detectar a

los individuos que ingresan en la sala por determinado trámite/delito, en este caso, la aplicación se la ejecutó con la ayuda de un video ya obtenido de dicha sala y poder obtener una base de datos de las personas a ser identificadas.

En esta aplicación, al utilizar como base un video, no muestra los resultados instantáneos de forma individual, sino que se hace uso del resultado acumulado en las última N imágenes (N=10), con lo cual, se logra que el resultado mostrado en el video sea más estable, evitando variaciones constantes y generar un reconocimiento incorrecto (Universitat Politècnica de Catalunya, 2013).

Al ser en tiempo real, las imágenes utilizadas en este aplicativo se encuentran sin recortar y presentan caras en diferentes posiciones y expresiones, para lo cual, es necesario para su procesamiento, el empleo de un algoritmo (detector facial) conocido como detector facial Viola-Jones (Hernández, 2010).

En Ecuador, el SRF también ha sido introducido como método de apoyo al ECU911, por ahora, se trata de un prototipo funcional implementado por la mencionada entidad de socorro en la ciudad de Ambato-Ecuador para encontrar a personas que se encuentran solicitadas por la justicia y que están dentro de los más buscados en Ecuador (Garcés Nuñez, 2017).

El mencionado prototipo de sistema empleado por el ECU911 requiere de una base de datos, por lo que previamente fue ingresado al servidor las imágenes de dichos individuos. Mediante el empleo de un dispositivo de video conectado al sistema, se logra la captura en imagen de la cara de la persona, se realiza luego una evaluación de los rasgos faciales a través del empleo de técnicas y métodos de procesamiento de imágenes, y de ésta forma, se identifica a la persona que se buscaba, con lo cual, se pueden enviar alertas en tiempo real que indican a las autoridades el lugar donde esta se encuentra o por donde se desplaza que incluyen datos de GPS para facilitar su ubicación, esta información se muestra en un mapa que es visualizado en una interfaz de monitoreo.

Mediante el uso de las técnicas actuales empleadas en el reconocimiento facial, se creó una aplicaciones denominada Análisis de Componentes Principales (PCA) que es utilizada para el reconocimiento facial por medio de funciones, siendo aplicada a través de la herramientas de programación Visual Studio (Garcés Nuñez, 2017).



En los últimos años el *Deep Learning* (o aprendizaje profundo) se ha posicionado como una de áreas con más desarrollo dentro del campo de la inteligencia artificial, obteniendo resultados más relevantes en comparación con otras técnicas de aprendizaje automático (Chen & Ran, 2019).

Gracias a los resultados de dichas técnicas, grandes empresas como Google, Facebook, Microsoft, entre otras, ya las han incluido como instrumentos imprescindibles para la evolución de nuevos productos (Bisong, 2019; Liu y col., 2020; Park y col., 2018; Wang y col., 2020). Sin embargo, en la Industria es un campo que aún se encuentra en fase de investigación y por explorar.

Entre las investigaciones que se han realizado en el área industrial, se encuentra el modelado de procesos Industriales, predicción de comportamiento, detección de rostros, diagnóstico de fallos, entre otros, para lo cuales se usa una combinación de librerías *open source* Keras-Tensorflow (González-Muñiz, 2018).

En diversos países desarrollados, se poseen sistemas de control de asistencia para el personal que trabaja en la Industria, sea en el sector público o privado, el principal objetivo de estos es verificar la asistencia de determinado colaborador. En cada institución se requiere un exhaustivo control de la asistencia del personal, para determinar el tiempo dedicado a la actividad laboral para lo cual se emplea tecnologías que vayan de acuerdo a la situación económica y tecnológica de cada empresa (Unal & Tecim, 2018).

Para poder realizar el reconocimiento de imágenes, además de emplear una base de datos, se puede hacer mediante redes neuronales, las cuales, tienen la habilidad de adquirir conocimientos basados en un entrenamiento adecuado, mismo, que implica lograr introducir ciertos datos que resulten en una adecuada salida como parte de sus resultados ingresar datos a la entrada y obtener óptimos resultados a la salida (Awoamin & Brand, 2020). Las Redes de este tipo (neuronales), internamente organiza los datos aportados de una manera óptima, lo que desliga al operador humano de realizar esta tarea.

Es de gran utilidad ya que estas redes pueden manipular de manera óptima todos los cambios generados por la información introducida, tales como ruido o perturbación que pueda generar alguna falla, y al trabajar con un hardware de este tipo, se puede trabajar en tiempo real debido a que la red neuronal es montada en paralelo (Bastidas Gavilanes, 2018).

Para llevar a cabo el registro de asistencia del personal, se puede desarrollar aplicaciones como por ejemplo una app de *Amazon Web Service* (Amazon, 2022) para integrarla a la aplicación de reconocimiento facial, de igual manera se puede generar una red convolucional para el reconocimiento del hablante y entrenamiento de los audios.

Debido a que el personal no siempre se encuentra en la misma geolocalización para realizar su trabajo, se puede definir previamente las coordenadas de ubicación de la empresa y el área (radio) sobre el que se puede realizar las marcaciones de registro y así conocer el lugar de trabajo de determinado colaborador (Carrión & Yandry, 2022).

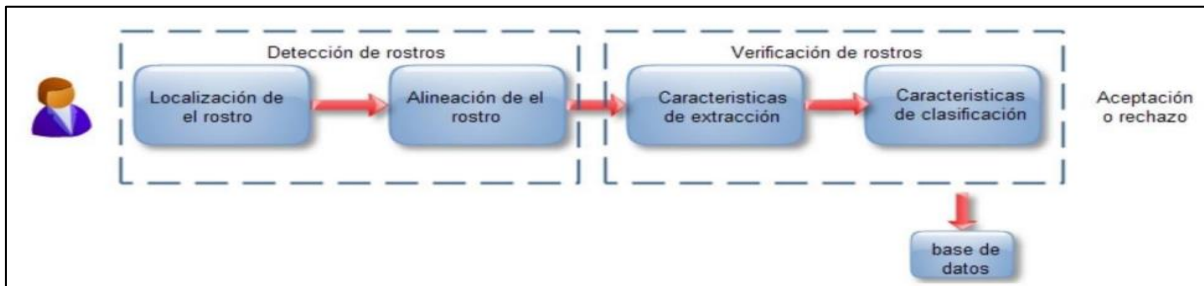
La seguridad de la implementación del reconocimiento facial como método de identificación preferencialmente sobre otros métodos se ha demostrado también en trabajos previos, al respecto, el trabajo de Espinoza y Jorquera (2015) en una industria chilena, y observaron que el reconocimiento por huella dactilar es más fácil de vulnerar ya que las huellas pueden ser clonadas, lo cual no ocurre tan fácilmente con un rostro.

A pesar de que la tecnología ya forma parte de la vida cotidiana, existen personas e incluso empresas que se niegan al cambio, al respecto, Bravo y col. (2018), en su investigación evaluaron la aceptación de los ciudadanos en cuanto al reconocimiento facial como un mecanismo de seguridad. Esa investigación, se llevó a cabo en Chile y se sustentó en el índice de predisposición tecnológica (TRI) y el modelo de aceptación de tecnologías (TAM).

En la misma, aplicaron una encuesta a 220 chilenos, los resultados del análisis indicaron en un 50% de los casos, las variables sociales, modo en que se percibe la responsabilidad entre otras analizadas, explicaron las observaciones sobre la utilidad del reconocimiento facial que poseían los evaluados.

El desarrollo de este trabajo de identificación, es posible si se consideran unos pasos básicos, al respecto, trabajos como el de Espinoza y Jorquera (2015), aportan una línea de procesos básicos que deben estar presentes y que abarcan desde la detección del rostro hasta la verificación del mismo empleando una base de datos previa para aceptar o rechazar el mismo como parte del registro (Imagen 3).

Imagen 3. *Etapas del reconocimiento facial*



Fuente: Espinoza y Jorquera (2015).

Como se aprecia, las tres etapas básicas que conforman el proceso son: detección del rostro, extracción de características y clasificación. Primero se realiza un análisis y documentación del sistema, realizando los diagramas correspondientes para el desarrollo.

Luego, se realiza la interacción con el usuario mediante una interfaz gráfica con el módulo inicial de captura de imágenes. Se realiza una descripción de la base de datos de rostros a partir de un conjunto de imágenes y se indican las técnicas en cada etapa del procedimiento.

Para poder detectar el rostro, se emplean algoritmos, los cuales, se basan en una representación global del mismo con la segmentación de piel, aplicar plantillas y detección de ojos. Son representaciones que se clasifican mediante una red neuronal, exactamente un perceptrón multicapa, identificando a la persona en cuestión. Por último, se exponen los experimentos realizados y se muestran los resultados obtenidos (Matias et al., 2010).

Unas de las ventajas del reconocimiento facial por sobre otros métodos de chequeo de personal (Tabla 1), es que esta tiene la posibilidad de registrar a más de una persona a la vez, que el registro sea verídico de la persona reconocida por el sistema y no de alguna otra más, agiliza el proceso de chequeo, mejorar el control del personal que trabaja en campo además de las ventajas de la digitalización de los datos relevantes identificativos de los trabajadores y la consecuente innovación de los procesos de la empresa (Garcés Nuñez, 2017).

**Tabla 1.** *Comparativa de técnicas de reconocimiento biométrico*

<b>Técnica</b>	<b>Coste</b>	<b>Requiere muestra</b>	<b>Contacto con el dispositivo de medición</b>	<b>Mantenimiento</b>
<b>Reconocimiento facial</b>	Medio	No	No	Si
<b>Huellas digitales</b>	Medio	Si	Si	Si
<b>Geometría de la mano</b>	Bajo	Si	Si	Si
<b>Voz</b>	Bajo	No	No	Poco
<b>Iris</b>	Alto	Si	No	Si

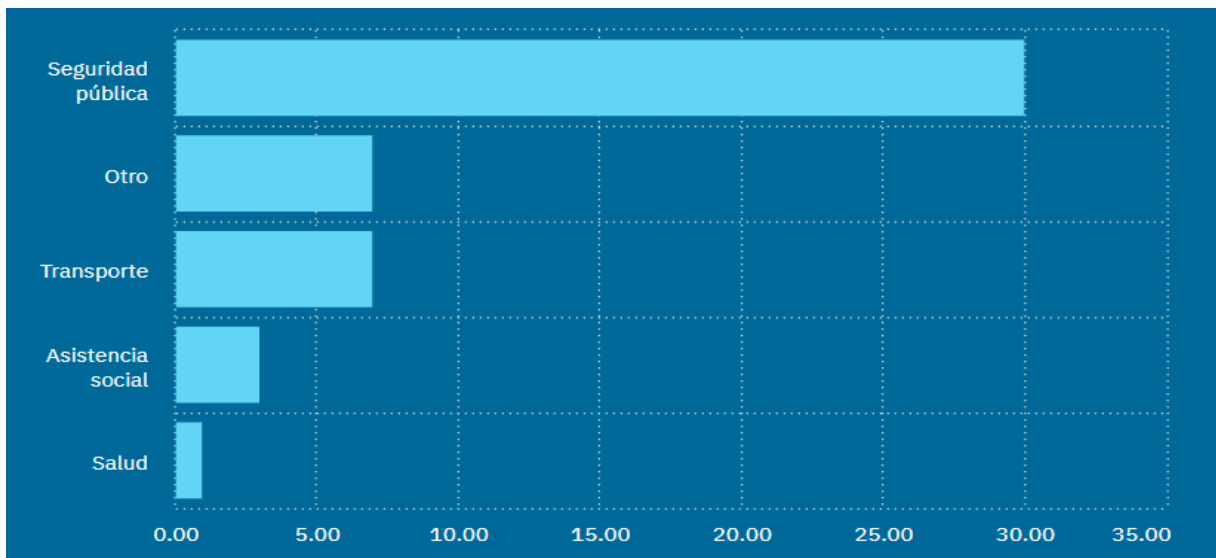
Fuente: Garces (2017).

Para poder llevar a cabo el cambio de cualquier parte de los procesos relacionados con la Industria hacia un entorno 4.0, se deben tomar en cuenta varios aspectos, por ejemplo, que las tecnologías han de ser compatibles con los hardware que ya posea la empresa, además de tener la posibilidad de comunicación con proveedores.

En este aspecto, al emplear tecnologías como internet Industrial, es importante incluso considerar los aspectos de ciberseguridad, de tal manera que el reconocimiento facial identifica cumple con parte de este cometido al identificar los rasgos de las personas mediante una tecnología biométrica, la misma que ha sido mejorada gracias al avance del *Big Data* y técnicas de inteligencia artificial como *Deep Learning* (Veiga García, 2021).

Con la Industria 4.0 se ha logrado integrar diversas tecnologías con lo cual se ha conseguido obtener sistemas inteligentes facilitando procesos, agilizando operaciones y garantizando buenos resultados. Entre los objetivos del SRF se encuentra incrementar los índices de seguridad, vigilancia y control del personal (Imagen 4) (Derechos Digitales, 2021).

Imagen 4. Áreas de aplicación para SRF.



Fuente: Derechos Digitales (2021)

### 2.2.2. Procedimiento para el reconocimiento facial

Para poder llevar a cabo el reconocimiento facial, como se ha indicado es necesario la generación de algunos algoritmos específicos, además de considerar la implementación de técnicas o modelos activos. Estos modelos, son una forma de poder determinar la posición de los puntos de interés con los que es posible el establecimiento de un modelo estadístico empleando el algoritmo de ASM, el cual, incluye dos rutinas (Elaboración del modelo y la búsqueda de los rostros (Johnston & Chazal, 2018; C. B. Singh et al., 2021).

Con el algoritmo ASM, la creación de un modelo, necesita que sean marcados  $n$  puntos relevantes de las tipologías faciales en cada imagen del conjunto de entrenamiento (Iqtait et al., 2018).

### 2.2.2.1. Ubicación de puntos en el rostro

En la primera etapa de este proceso de generación del modelo, para poder establecer los puntos clave en la imagen del rostro, se emplea ASM, con el cual, se crea un modelo de patrones que luego será evaluado. La preparación se realiza empleando los bordes de la imagen que previamente fue obtenida y establecida los puntos (Imagen 5) (Gualdrón & Suárez, 2014).

**Imagen 5.** *Puntos seleccionados del modelo inicial*



Fuente: Gualdrón y Suárez (2014)

#### 2.2.2.2. Ubicación y detección de ojos y características de la piel

La ubicación de los ojos constituye puntos estratégicos para la generación del modelo y que permiten posteriormente el reconocimiento facial para seguimiento de mirada. Así mismo, el modelo del tono de la piel, se basa en un cálculo que detalla la repartición de colores o tonos de la piel (Imagen 6) (Gualdrón & Suárez, 2014).

**Imagen 6.** Ecuaciones para los diferentes tonos de piel

$$\begin{aligned} \min(c) &= \min(R, G, B) / 255 \\ \text{med}(c) &= \text{med}(R, G, B) / 255 \\ \max(c) &= \max(R, G, B) / 255 \end{aligned}$$

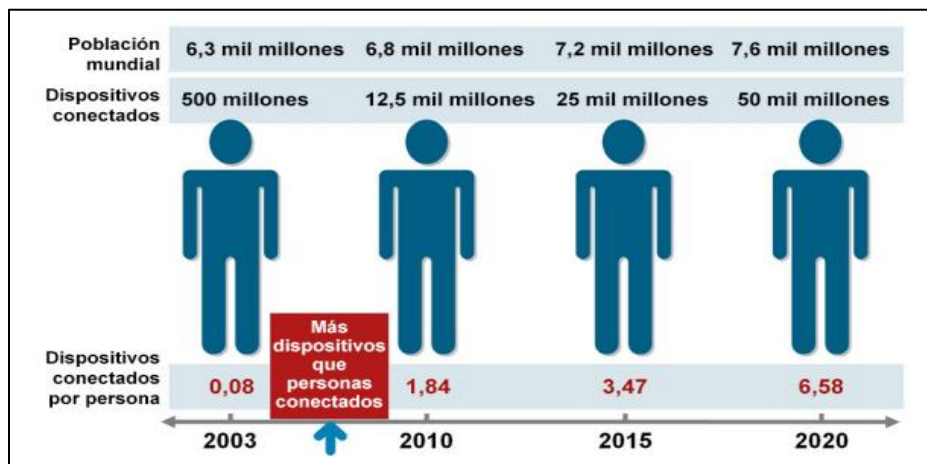
Fuente: Gualdrón y Suárez (2014)

### 2.3. TECNOLOGÍAS RELACIONADAS CON EL SRF

El IoT (Internet of Things) ha cambiado muchas cosas y su impacto se ha visto reflejado en diferentes ámbitos de la vida diaria de las personas, entre estos, se puede mencionar el espacio educativo, la comunicación, las operaciones en las empresas, la ciencia, los procesos gubernamentales entre otros más (Evans, 2011).

Esta forma de interconectar “las cosas” se perfila como la próxima evolución del internet cuando logre desarrollarse por completo, ya que cada vez existen muchos más dispositivos conectados entre sí, al punto que en la actualidad se estima que más de 50 mil millones de dispositivos se encuentran conectados respecto de una población de 7.6 mil millones de habitantes en todo el mundo (Imagen 7) (Evans, 2011).

Imagen 7. Cantidad de dispositivos conectados cada año



Fuente: Evans (2011).

Gracias al avance exponencial en materia de reconocimiento facial que se está experimentando, nuevos sistemas embebidos serán pronto capaces de conectarse a internet y realizar una transferencia de datos entre la nube y los clientes, al respecto, Cabas-torres (2019) afirman que muchas de las tecnologías que se usan hoy en día son de software libre, lo que permite a las personas con conocimiento en electrónica crear sus propias soluciones interconectadas.

### 2.3.1. Big Data

Al encontrarnos con una variedad de dispositivos conectados, la cantidad de datos que deben ser transferidos también aumenta de forma en gran medida, por lo que se requiere de la tecnología Big Data, para recopilar, analizar y gestionar los datos que se encuentran en internet, en estos casos, el objetivo principal es recopilar los datos masivos y poder procesarlos posteriormente (Fernández, 2021).

Los usos que se les puede dar a Big Data en las empresas son diversos, entre estos se encuentran: Mejora de productos, experiencia de los clientes, validación y detección de fraudes, entre otros. Con dicha tecnología, las máquinas en lugar de ser programadas, pueden aprender por sí mismas gracias a la disponibilidad de gran cantidad de información que funciona como base para la generación de modelos de aprendizaje automático (APD, 2019).



Para poder llevar a cabo un aprendizaje automático se requiere de inteligencia artificial, la cual, tiene por objetivo estudiar el comportamiento inteligente de las máquinas. De la misma forma, una conducta inteligente supone interactuar de diversas maneras con entornos complejos de acuerdo con los datos obtenidos. Entre las metas de la IA es el desarrollo de máquinas que puedan ejecutar las tareas mencionadas igual o mejor que los humanos (Chacua Criollo, 2019).

### 2.3.2. Visión Artificial

La visión artificial corresponde a un área de estudio de la IA y comprende un grupo de algoritmos de procesamiento para las características de una imagen que permite que esta pueda ser analizada usando una cámara de video (Paraskevoudis et al., 2020).

La visión artificial necesita de dos elementos principales, el hardware que toma las imágenes y el software que la procesa, y este proceso global, es muy eficaz en la realización de tareas que suelen ser repetitivas y tediosas para el hombre, estas facilidades se logran por medio de los avances en torno a los sensores visuales que, en la actualidad tienen diversas aplicaciones en el mundo real como por ejemplo la interacción entre personas y las computadoras, la robótica, etc.

Como contraparte se tiene que la desventaja principal, es que la visión artificial no tiene la capacidad total de entender el mundo en 3D fácilmente (Tabla 2), por lo cual, existen algoritmos capaces de reconstruir un modelo en 3D preciso usando asociaciones complejas en múltiples vistas, sin embargo, en la actualidad todavía no es una tarea fácil de ejecutar (Chacua Criollo, 2019).

**Tabla 2.** *Ventajas de la visión humana vs la artificial*

<b>Sistema Humano</b>	<b>Sistema artificial</b>
<b>Mejor reconocimiento de objetos</b>	Mejor midiendo magnitudes físicas
<b>Mejor adaptación a situaciones imprevistas</b>	Mejor para la realización de tareas rutinarias
<b>Utilización de conocimiento previo</b>	Mejor en tareas de procesamiento de imágenes de bajo nivel
<b>Mejor en tareas de procesamiento de imágenes visuales de alto nivel</b>	

Fuente: Tomado de Chacua (2019)

### 2.3.3. Machine Learning

El *Machine learning*, o entrenamiento de máquinas, constituye uno de los subcampos de la Inteligencia Artificial, sobre este, Román (2019) explica que es una ciencia que permite que los sistemas computarizados logren aprender, y en consecuencia, puedan actuar de manera similar a los humanos, facilita que estas puedan mejorar su proceso de aprendizaje autónomamente a lo largo del tiempo gracias a la constante interacción de estas con grandes bases de datos e informaciones provenientes del mundo real.

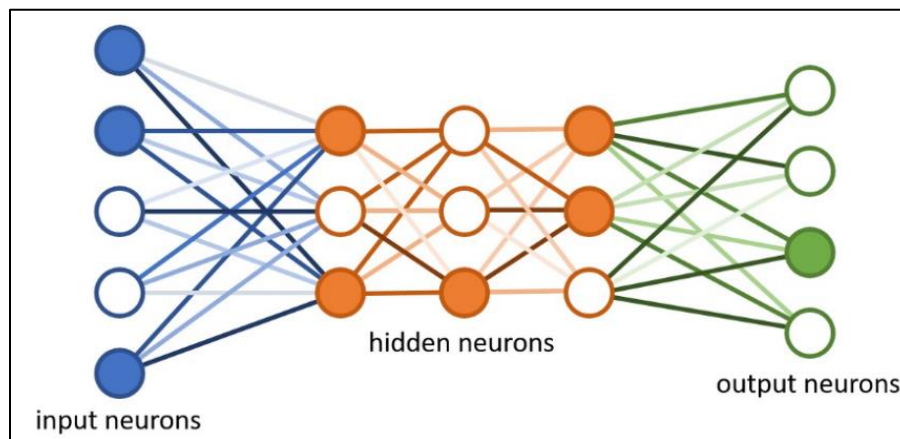
Este mismo investigador (Roman, 2019), indica que para poder mejorar de forma gradual, el rendimiento de *Machine Learning* se orienta en parte, en el reconocimiento de imágenes y de voz, conducción automática de vehículos y filtros antispam de correo electrónico, por lo que en la actualidad han evolucionado tres variantes de este método, a saber; el aprendizaje supervisado, el no supervisado y el aprendizaje profundo (*Deep Learning*).

### 2.3.4. Deep Learning

Como se mencionó anteriormente, uno de los subcampos de *Machine learning* es *Deep Learning*, el cual, presenta una estructura de redes neuronales artificiales que forman de manera similar a la estructura neuronal del cerebro (Chen & Ran, 2019).

En el *Deep Learning* las redes se dividen por capas y los datos en bruto son datos de entrada que son procesados y de los cuales se extrae la información que es transferida a la siguiente capa en forma de una “salida”, estos, se repite en cada una de las capas para al final obtener la predicción, la cual se compara con el resultado conocido (Imagen 8) (Román, 2019).

**Imagen 8.** Representación de capas neuronales



Fuente: Román (2019)

## 2.4. CONCLUSIONES SOBRE EL ESTADO DEL ARTE

Para que las empresas puedan desarrollarse competitivamente hoy en día la evidencia muestra que deben apuntarse en el camino de la evolución a un sistema industrial marcado por los estándares de las Industrias 4.0, para que, de esta manera, sus procesos y productos mejoren y, por ende, logren mantenerse competitivas en las nuevas formas de desarrollar y sobrellevar los negocio.

Gracias a la ayuda de las nuevas tecnologías habilitadoras de la Industria 4.0 tales como el IoT, logran tener un ahorro de costes y, además se disminuye el trabajo de diseño e implementación técnica.

Se debe considerar todas las ventajas que ofrece la Industria 4.0 para que la empresa sea competitiva y se encuentre siempre a la vanguardia tecnológica, ya que en la actualidad las empresas que no logran adaptarse a los avances simplemente están destinadas a desaparecer si se dejan sobrepasar en este aspecto de la innovación por parte de otras más decididas a dar el cambio.

Las investigaciones y trabajos consultados muestran diversas soluciones para poder resolver el problema planteado. Garcés (2017), en su investigación, indica que el reconocimiento facial es de gran ayuda para la sociedad debido al análisis y procesamiento de imágenes. La mayor parte de los trabajos académicos encontrados, se enfocan en el reconocimiento facial para diversas aplicaciones.

En este sentido, Rivas (2022) orienta su investigación hacia el control de asistencia del personal mediante el uso de tecnologías habilitadoras que permiten desarrollar el reconocimiento facial.

Así mismo, la mayoría de los proyectos consultados, se centran en soluciones particulares sustentadas en las necesidades propias de cada entorno, pero todos poseen un factor común que es el uso de dispositivos IoT como base para poder llevar a cabo el reconocimiento facial, por lo tanto, esta es una de las tecnologías que comúnmente se considera para ayudar a solventar los problemas específicos de cada empresa.

También se encontraron evidencia en el trabajo de Chacua (2019) sobre el uso de la inteligencia artificial para el desarrollo de sistemas viables de reconocimiento facial. Por otra parte, Aldana (2018) propone un sistema de control biométrico empleando técnicas de *Machine Learning* y *Deep Learning*, además de *Open CV* y otras librerías que permitan generar un código abierto para tratar el tema de la inteligencia artificial.

La solución que se propone en el presente TFM se sustenta en las evoluciones tecnológicas características de los entornos industriales 4.0, y presenta un enfoque que puede ser adaptable a cualquier ámbito en el que se requiera de un SRF, un sistema de vigilancia, seguridad o control de asistencia.

Finalmente, cabe mencionar que el uso de las tecnologías consultadas permite la generación de una propuesta para un SRF tal como puede verse en trabajos de Torres y col. (2021), Bastidas (2018) y Rivas (2022).

De igual manera, estos son trabajos que pueden ser escalables mejorando y actualizando las tecnologías empleadas y llevados a un nivel Industrial en el cual puedan ser usados de forma offline ya que, en muchas ocasiones, por diversas circunstancias como las geográficas, no se dispone de acceso a internet, por lo tanto, el sistema propuesto debe ser capaz de trabajar en dichas condiciones de escasa o nula interconexión a la red global de internet, lo cual, hace que la propuesta sea novedosa y distinta a los sistemas IoT presentados previamente en los trabajos citados.

### 3. DESCRIPCIÓN GENERAL DE LA CONTRIBUCIÓN DEL TFM

En este capítulo se expone la manera en la que el presente trabajo contribuye en la Industria 4.0 mediante tecnologías habilitadoras a través del planteamiento y cumplimiento de objetivos.

#### 3.1.OBJETIVOS

##### **Objetivo general:**

Crear una propuesta de implementación de un SRF usado en el registro de asistencia de personal en campo basado en la inteligencia artificial.

##### **Objetivos específicos:**

Los objetivos específicos del presente TFM son:

- Evaluar las capacidades y necesidades de la empresa en cuanto a requerimientos técnicos para la implementación de un SRF basado en un dispositivo móvil conectado a una plataforma IoT.
- Diseñar un plan de acción para el desarrollo por parte de la empresa, de un SRF para el chequeo del personal de trabajo.

### 3.2.METODOLOGÍA DEL TRABAJO

En la presente investigación, dado que no es un proyecto de implementación, sino una propuesta para implementar un SRF, se asumió un método de trabajo constituido en etapas, con el cual, fue posible la planificación, diseño y finalmente la generación de la propuesta de implementación.

No se implementó un método de trabajo común en el desarrollo y ejecución de propuestas tecnológicas porque en el presente trabajo no se realizó una implementación, y, por tanto, no fue posible el seguimiento secuencial de los avances en el proyecto en el terreno, ni las correcciones del mismo cada vez que se localice un nodo a solventar y corregir, como, por ejemplo, se contemplan en metodologías como la SCRUM.

**Primera etapa:** Evaluación de las capacidades y necesidades de la empresa en cuanto a requerimientos técnicos para la implementación de un SRF basado en un dispositivo móvil conectado a una plataforma IoT.

1. Cuantificación de los recursos tecnológicos con los que cuenta la empresa
2. Identificación de las necesidades tecnológicas (software y hardware) necesarias para la implementación de un SRF funcional

**Segunda etapa:** Diseñar un plan de acción que pueda implementar la empresa objeto de estudio, basado en un SRF para el chequeo del personal de trabajo.

- Descripción de la arquitectura a emplearse
- Elección de la tecnología
- Configuración del sistema de video empleado en el reconocimiento facial
- Identificación de la plataforma IoT

En todas las etapas del diseño de la propuesta, se abarcaron los pasos y subpasos que conformen cada etapa, y en la misma, se describieron todos los aspectos relacionados con cada uno, de manera que quedaran claramente descritos y sirvieran para una implementación sin dificultades.

### 3.3.DESCRIPCIÓN GENERAL DE LAS PARTES O COMPONENTES DE LA PROPUESTA

A lo largo de este capítulo se discutirá acerca de las tecnologías habilitadoras de la industria 4.0 las cuales permitirán realizar un proceso de registro de asistencia del personal para su correcto manejo en nómina por parte de la empresa 3A Composites.

#### 3.3.1. Descripción general

La propuesta que se presenta consiste en el diseño metodológico y procedimental que abarca desde el reconocimiento de la situación tecnológica actual de la empresa con el fin de reconocer las debilidades que afectan a la implementación y transcurre por el diseño sistematizado y ordenado de un sistema implementable, funcional y de bajo costo que permita el monitoreo del personal que labora en el campo.

Se considera para esto, el empleo de tecnologías accesibles en la actualidad, que permitan el reconocimiento facial de los empleados, y que el departamento de recursos humanos pueda llevar el control adecuado del personal que se presenta realmente a la jornada laboral, este SRF tiene la particularidad de que funcionará sin estar conectado a la RED de internet, por lo tanto, se considera también la implementación de una base de datos local donde se puedan gestionar todas las operaciones de reconocimiento de los trabajadores, y que, de manera periódica, por ejemplo, una vez a la semana, un operador representante de la empresa pueda hacer la transferencia de los registros de asistencia al departamento de recursos humanos y que este pueda en ese momento, manejar dicha data de la manera que estime conveniente.

En base a lo anterior, se tiene que la presente investigación, que se fundamenta en una propuesta de implementación viable, conste de las siguientes partes y características:

**Primera etapa.** Evaluación de las capacidades y necesidades de la empresa en cuanto a requerimientos técnicos para la implementación de un SRF basado en un dispositivo móvil conectado a una plataforma IoT.:

1. Cuantificación de los recursos tecnológicos con los que cuenta la empresa: Se verificara con la ayuda de la dirección de la empresa y el departamento de recursos humanos, la disposición de computadores, dispositivos móviles y posibilidades de conexiones a internet con las que la empresa cuenta en el lugar de la implementación o que pueda destinar para este proceso.
2. Identificación de las necesidades tecnológicas (software y hardware) necesarias para la implementación de un SRF funcional: En base a los hallazgos del paso anterior, se realizó un listado de las necesidades tecnológicas y de software que deben adquirirse, así como, se detalló cuáles son los elementos de programación necesarios.

**Segunda etapa:** Diseñar un plan de acción para el desarrollo por parte de la empresa, de un SRF para el cheque del personal de trabajo.

1. Especificación de la arquitectura a emplearse:
  - Elección de la topología correcta para poder realizar el sistema reconocimiento facial y registro.
  - Descripción del componente de inteligencia artificial empleado
  - Definir cada una de las partes del sistema.
2. Selección de la tecnología
  - Elección del hardware de video
  - Elección del hardware de procesamiento
  - Elección del algoritmo para el entrenamiento del reconocimiento facial, basado en la inteligencia artificial
  - Entrenamiento y testeo sobre el SRF
  - Planeación de la base de datos



- Creación del software (HMI)
  - Generación de la estrategia para el Match entre el algoritmo de reconocimiento facial y la base de datos
  - Diseño de la estrategia para el Match entre el algoritmo de reconocimiento facial y el HMI
3. Configuración del sistema de video empleado en el reconocimiento facial
- Configuración inicial y conexión entre sistema de captura de video y la base de datos local.
  - Programación del script para poder realizar el Deep learning con la base de datos de rostros establecida.
4. Configuración de la plataforma IoT
- Registro de credenciales.
  - Generación de usuarios

### **3.3.2. Alcance y limitaciones**

Con el sistema propuesto, se incluye la implementación de un hardware para la captación de video, así como la implementación de una estación de cómputo que sirva de servidor local y que además de contener la base de datos de los trabajadores, contenga el algoritmo de IT que realiza el reconocimiento facial de los empleados, así mismo, contara con un acceso a una plataforma IoT que permitirá el intercambio periódico de los registros de asistencia de los empleados con el departamento de Recursos Humanos de la empresa ubicada en otro lugar alejado.

La principal limitación del sistema propuesto se traduce en la capacidad y/o disponibilidad de la empresa para lograr establecer un punto de acceso a internet en el lugar de implementación de la propuesta con el cual sea posible de manera periódica que el supervisor del personal logre enviar los respectivos reportes de asistencia.

### **3.3.3. Listado de participantes**

En la presente investigación han participado en forma de asesores el departamento de recursos humanos de la empresa ecuatoriana “3A Composites”.

### **3.3.4. Tecnologías implicadas**

Las tecnologías implicadas en la propuesta son IOT, Big Data y Deep Learning.

### **3.3.5. Arquitectura, componentes e integración de tecnologías.**

El sistema, de reconocimiento facial se conecta mediante la tecnología IoT a los dispositivos utilizados (cámara, y ordenador), la integración con la tecnología Big Data es realizada por medio de la plataforma Google Cloud para poder manejar la gran cantidad de registros que se tendrán a lo largo de las jornadas laborales donde se emplee el reconocimiento facial para el chequeo del personal. Esta información, será posteriormente empleada por el departamento de recursos humanos para el debido procedimiento de control de asistencia.

### **3.3.6. Resultados esperados**

Se espera que con esta implementación que se propone, la empresa objeto de estudio logre resolver sus deficiencias tecnológicas y operativas con respecto al chequeo de personal en una de sus instalaciones en el campo, donde carecen de conexión a internet y no hay ningún proceso de la empresa automatizado hasta el momento.

### 3.3.7. Presupuesto y retorno esperado de la inversión

Con la presente propuesta de implementación se espera que la empresa objeto de estudio, tenga a disposición los lineamientos e indicaciones de hardware y software necesario para lograr la implementación efectiva de un SRF de bajo costo, pero enteramente funcional que le permita llevar a cabo el monitoreo del personal que ingresa y/o se retira de sus jornadas laborales en una de sus instalaciones que carece de acceso a internet.

## 4. Desarrollo específico de la contribución del TFE

En el capítulo 4 se dará a conocer la manera en la que se empleará la tecnología seleccionada, así como la configuración tanto de los dispositivos como de la plataforma IoT.

### 4.1.ESPECIFICACIÓN DE LA ARQUITECTURA A EMPLEARSE:

Para poder llevar a cabo la implementación propuesta en este trabajo, se requiere un “ambiente controlado” el cual, corresponde a un ambiente cerrado, en el cual la luz, la humedad, la temperatura entre otros, son controlados completamente, con esto, se facilitará el poder tener imágenes con características ambientales iguales. Entre los componentes que conforman parte de dicha arquitectura se tiene:

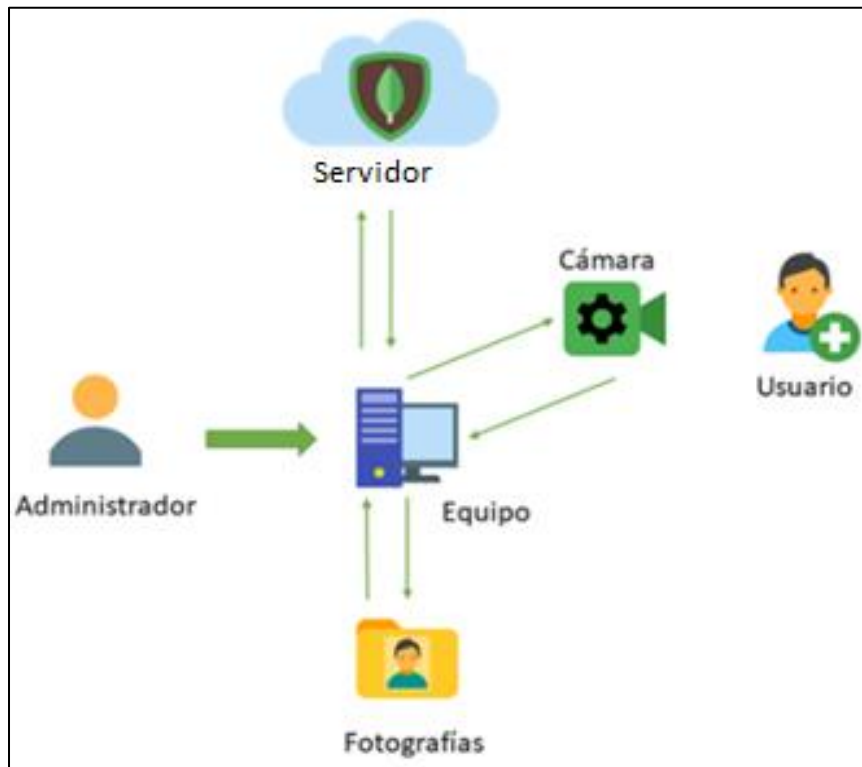
- ✓ Fondo de color azul: Empleado para mantener una uniformidad en la imagen captada generando este color un mayor contraste entre el rostro y su entorno, lo cual, mejora el proceso de reconocimiento facial.
- ✓ Dispositivo de video: un (01) dispositivo de captura de video, el cual, se debe instalar en el entorno seleccionado para la captura de las imágenes.

- ✓ Unidad de procesamiento (ordenador): Donde se almacenará la base datos local de los rostros de los empleados, así como también, realizará la ejecución del algoritmo y permitirá la visualización de la interfaz.
- ✓ Servidor que permita el almacenamiento de los registros de asistencia creados en base al algoritmo de reconocimiento facial (plataforma IoT).
- ✓ Iluminación basada en la luz natural del día: al ser jornadas laborales únicamente diurnas, se recomienda emplear la luz natural como contexto de iluminación para la captura de las imágenes, lo que garantiza que todas las capturas se encuentren en condiciones similares y permite el ahorro de dinero que se podría requerir para la instalación de sistemas de iluminación.

#### **4.1.1. Elección de la topología correcta para poder realizar el sistema reconocimiento facial y registro.**

Para que el sistema pueda funcionar adecuadamente, debe ser instalada una cámara, la cual, será la encargada de realizar el reconocimiento facial en base al análisis ejecutado por el algoritmo de reconocimiento facial que se implementará, mismo, que emplea una base de datos local que almacena las imágenes de muestra de los empleados, y con las cuales se realizara el Match para el chequeo. En la Imagen 9, se observa la topología a seguir para el SRF y registro.

Imagen 9. Topología del SRF propuesto



Fuente: Creado por el autor.

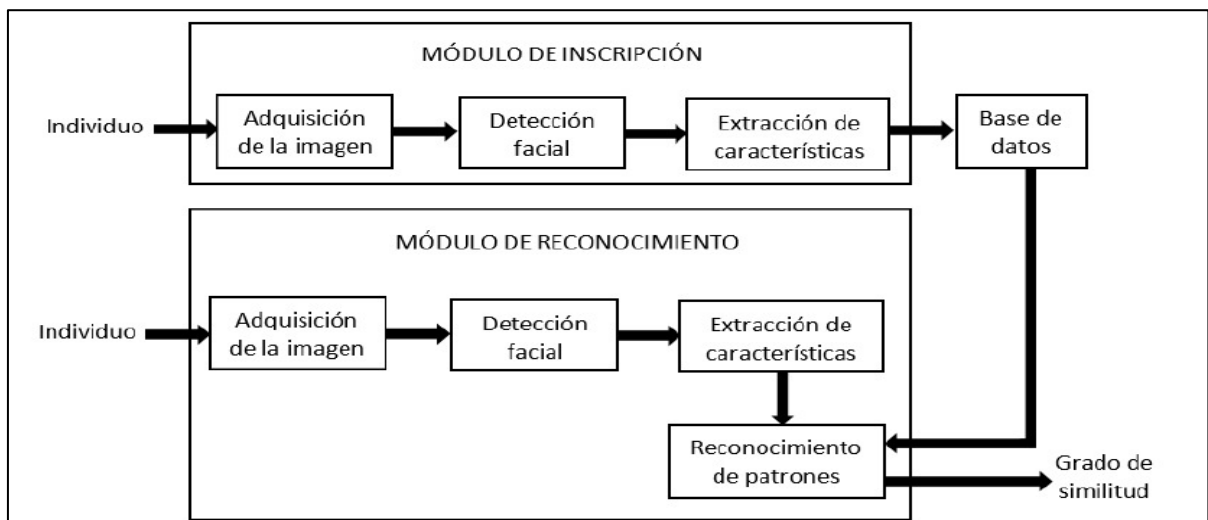
#### 4.1.2. Descripción del componente de inteligencia artificial empleado

El componente de inteligencia artificial empleado es el aprendizaje automático, ya que dicho aprendizaje se fundamenta en algoritmos para que la computadora logre aprender en base a los datos suministrados. Se empieza con los análisis de los datos para averiguar sobre la existencia de patrones en estos, en este caso patrones en las fotos de la base de datos. El aprendizaje automático brinda a las computadoras la capacidad de analizar y clasificar grandes volúmenes de datos.

#### 4.1.3. Definir cada una de las partes del sistema.

El SRF está conformado por dos módulos, uno, el de inscripción, en cual se registran las imágenes que se encuentran en la base de datos con sus respectivas características (patrones del rostro: distancia entre ojos, distancia entre nariz y boca, distancia entre nariz y ojos y tonos de piel), y otro modulo, el de reconocimiento, el cual tendría la función en si misma de realizar el Match del rostro capturado en la imagen de la cámara. En la Imagen 10 se muestra diagramadas las partes que constituyen el SRF propuesto.

**Imagen 10.** Estructura de las partes que conforman el SRF propuesto



Fuente: Elaboración propia

## 4.2. SELECCIÓN DE LA TECNOLOGÍA

A continuación, se presenta una descripción detallada de los aspectos abordados en la selección de la tecnología requerida para la implementación del sistema de reconocimiento facial que se propone.

#### 4.2.1. Elección del hardware de video

Las capacidades de la cámara son sumamente importantes porque permiten ir más allá de la simple toma de una imagen, las prestaciones de la cámara facilitarán la captura de imágenes. De esta manera, una de las características vitales es la resolución de la misma.

Las imágenes captadas con las cámaras de video pueden diferir enormemente en este parámetro, por tanto, todas las imágenes captadas no son esencialmente iguales ni de la misma calidad, lo que conlleva que para el caso de la presente implementación se pueda contar con un único tipo de cámara que presente una resolución en HD (1920x1080 píxeles), y que, por tanto, permita obtener imágenes nítidas con las que sea más fácil la verificación de las características faciales de las personas.

Por las razones descritas, se debe emplear una cámara ECAM8000, la cual, es un dispositivos que copta imágenes en HD y mucha nitidez, adicionalmente, este dispositivo, posee la capacidad de reducir el efecto de contraluz al emplear la tecnología WDR (Wide Dynamic Range), adicionalmente, el dispositivo cuenta con abertura de lente bastante amplia, situación que redundará en el beneficio de la captación de luz en la imagen en circunstancias fortuitas de baja luminosidad (Imagen 11). Esta cámara es fabricada por la empresa Genius y se encuentra en el mercado a un costo accesible.

El fabricante de la cámara ECam 8000, la empresa Genius, describe en su página web las características operativas de este hardware (Genius, 2020), mismas que son las requeridas para este proyecto de implementación, y las cuales, la empresa que implemente dicha propuesta, siempre debe de mantener como un estándar mínimo- Las características que indica el fabricante son las siguientes:

**Imagen 11.** ECam 8000 Full definición alta 1080p



Fuente: Genius (2020)

- Unidad de sensor CMOS de píxeles de alta definición 720p.
  - Lente de foco fijo.
  - Formato de archivo MPEG/WMV.
  - Micrófono Digital.
  - Resolución (DPI) 2MP, 1920 x 1080, 1280 x 720, 640 x 480 píxeles.
  - Resolución de video CIF/VGA: hasta 30fps / 720p HD: hasta 30 fps / 1080P hasta 30 fps.
  - Ángulo de visión, arriba y abajo de 90 ° / 360 ° de rotación.
  - UVC (Plug & Play).
  - Peso 82 g (Incluye clip y cable).
  - Dimensiones (A x A x P) 54,5 x 90,6 x 67,5 mm (2,15 x 3,57 x 2,66 pulgadas).
- (Genius, 2020)

#### **4.2.2. Elección del hardware de procesamiento**

Para la implementación de este proyecto se debe instalar un computador con las siguientes características:

- Windows 10 Pro de 64 bits.
- Lector y grabador de DVD 9.5 mm
- 1 HP Single Unit Packaging



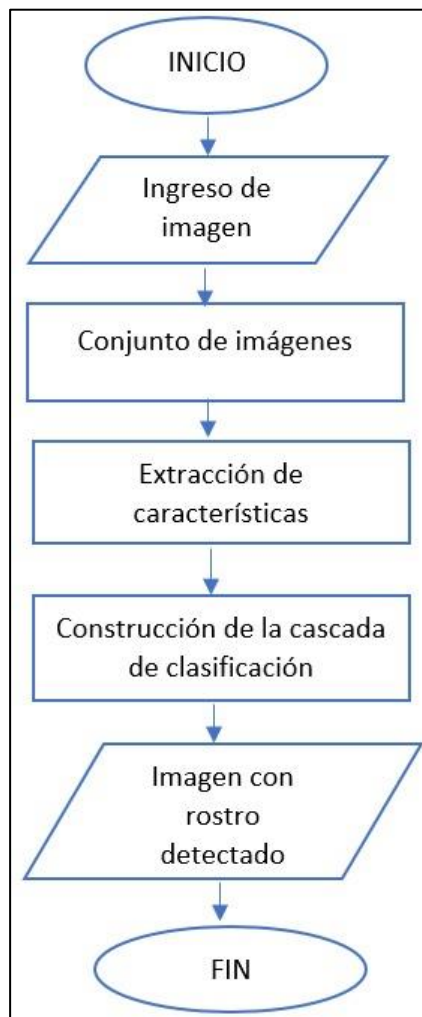
- 1 procesador Intel Xeon modelo E5-2667 v4 (8 núcleos de 3.2 GHz, 25 MB cache, 2400 MHz)
- 1 TB SATA SSD
- 1 case HP Z640 925W
- HP USB Optical Mouse
- 64GB de memoria RAM DDR4-2400 (4x16GB)
- Contratación mínima de la garantía de 3 años de HP.
- Intel® Ethernet I210-T1 Network Interface Card – PCIe
- Tarjeta GPU NVIDIA GeForce GTX 1080 8GB
- Licenciamiento SQL: Licenciamiento SQL server standard para un servidor y las cal necesarias de acuerdo con los siguientes códigos de solicitud: 228-04628 SQLSvrStd SNGL LicSAPk OLP NL(X1) + 359-01005 SQLCAL SNGL LicSAPk OLP NL UsrCAL(X1).

#### **4.2.3. Elección del algoritmo para el entrenamiento del reconocimiento facial, basado en la inteligencia artificial**

El método Viola-Jones es uno de los procesos más empleados hoy día (Gonzalez & Velásquez, 2019) porque permite dividir múltiples rostros en una sola imagen con un mínimo requerimiento de tiempo de ejecución. Viola y Jones basaron el algoritmo en características simples en lugar de píxeles.

Para ello utiliza una base tipo Haar que sirve para obtener las particularidades de cada rostro y el algoritmo Adaptive Boosting (Adaboost) para la elección y ordenamiento de características útiles en el reconocimiento, de manera general, este algoritmo se divide en tres etapas a saber: (1) Transformación, en el cual se genera una nueva imagen llamada “imagen integral”; (2) Extracción de características: uso de filtros basados en Haar. Y (3), que es la Construcción de clasificadores en cascada: Usando boosting (Imagen 12).

**Imagen 12.** Construcción de clasificadores en cascada usando boosting



Fuente: Almeida (2017), diagramado por el autor

Tal como se aprecia, el algoritmo de Viola-Jones es el responsable del procedimiento de detección que tiene como ventajas con una gran rapidez computacional y bajo coste, de acuerdo a los autores de dicho algoritmo, quienes son citados por Niola y Sanango (2019, p. 10), este es un procesador Pentium III de 700 MHz, con el detector facial se puede procesar una imagen de 384x288 píxeles en aproximadamente 0.067 segundos, de esta manera se tiene una detección mucho más confiable y robusta en tiempo real. Este algoritmo trabaja con una clasificación mediante características en vez de píxel a píxel y permite detectar objetos específicos, es por ese motivo que se aplica la detección facial, este es el primero paso para el reconocimiento facial.

#### 4.2.4. Entrenamiento y testeo sobre el SRF

Con el algoritmo ya seleccionado y tras la implementación del mismo, se genera una imagen integral, la cual, es una imagen acumulativa basada en la imagen original y que permite calcular la suma de los valores de los píxeles en cualquier área rectangular de una imagen dada (Platero, 2015).

Seguidamente el algoritmo realiza la extracción de características, misma que se ejecuta aplicando filtros de imagen basados en Haar sobre la imagen integral. Estas características se calculan como la diferencia de la suma de los píxeles de dos o más zonas rectangulares adyacentes (Gonzalez & Velásquez, 2019) (

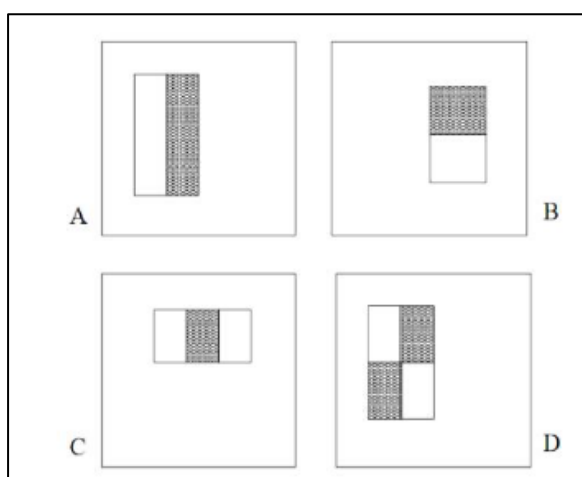
Imagen 13). Este algoritmo utiliza tres tipos de características de Haar:

**Tabla 3.** *Características de Haar*

Característica de Haar	Descripción
La función de dos rectángulos	Diferencia entre la suma de los píxeles de dos Rectángulos. Estos rectángulos tienen la misma forma y son adyacentes vertical u horizontalmente.
La función de tres rectángulos	Calcula la suma de los píxeles dentro de dos rectángulos exteriores, a su vez, restados de la suma de un tercer rectángulo interior
La función de cuatro rectángulos,	Calcula la diferencia de rectángulos emparejados en diagonal.

Fuente: Almeida (2017), diagramado por el autor

**Imagen 13.** *Características de Haar usadas en Viola-Jones*



Fuente: Niola y Sanango (2019, p. 11)

Y finalmente se ejecuta la construcción de clasificadores en cascada Boosting, que es un método de clasificación que utiliza clasificadores básicos (Adaboost) para formar un único clasificador más complejo y preciso (Platero, 2015). Este algoritmo agrega clasificadores simples uno tras otro, cada uno con una precisión ligeramente mayor que una clasificación aleatoria y los ordena para obtener un elemento de clasificación mucho más preciso.

#### 4.2.5. Diseño de la base de datos

Se debe almacenar en un archivo las imágenes de los rostros de 32 empleados, cada una debe poseer 18 tomas fotográficas en diferentes posiciones, lo cual significa que, en total, esta base de datos estaría conformada por 576 imágenes de rostros. Todas estas, deben ser convertidas a escala de grises, este proceso de conversión a gris es realizado por un script específico para esto (Imagen 14) que forma parte del algoritmo de reconocimiento facial empleado.

**Imagen 14.** *Script específico para la conversión de las imágenes a gris*

```
% Area de trabajo
work_area2=imresize(recorte2,[120
120]); % Cambia el tamaño de la
imagen a 120x120 pixeles.
img=rgb2gray(work_area2); %
Transforma la imagen a escala de
grises.
```

Fuente: Elaboración propia.

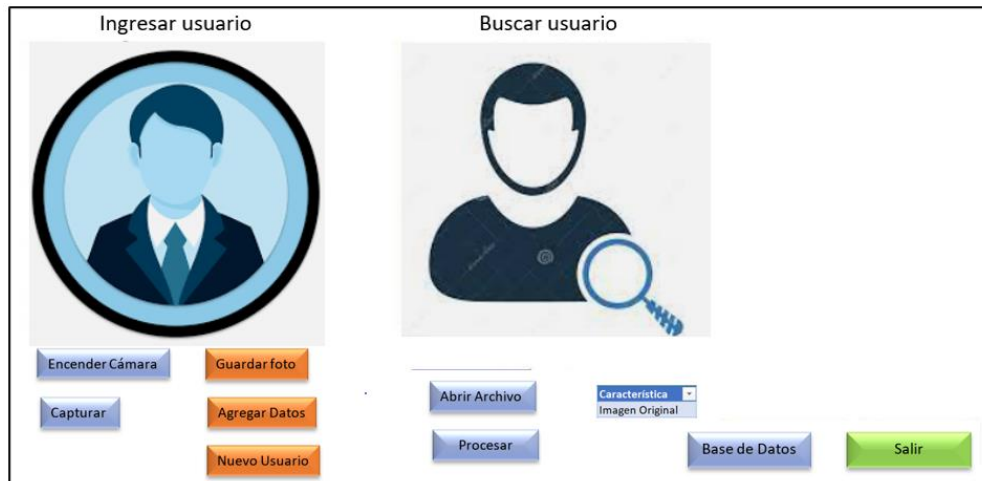
#### 4.2.6. Diseño del software (HMI)

El software que debe emplearse para poder diseñar el HMI es Matlab, gracias a sus prestaciones se puede crear tres interfaces que dispongan de los siguientes componentes (ver Imagen 15):

##### Primera Interfase:

- 1 botón para “Encender Cámara”, mismo que prendera la cámara
- 1 botón de “Capturar”, con el cual, se toma la imagen del empleado para su registro.
- 1 botón para “Guardar Foto”, con este, se almacena una nueva imagen dentro de la base de datos.
- 1 botón llamado “Agregar Datos”, que, al ser presionado, dirigirá al operador a la 3 interfase.
- 1 botón llamado “Nuevo Usuario”, con el cual, se puede acceder a la 4 interfase este botón nos redirige a la cuarta interfase.
- 1 botón denominado “Abrir Archivo”, que sirve para abrir archivos tipo .jpg que se encuentren en la computadora.
- 1 botón llamado “Procesar” el cual realiza el proceso: de reconocimiento facial.
- 1 botón llamado “Características de imagen”, con el cual, se expande un menú que contiene las opciones:
  - Imagen Original
  - Detección facial: esta opción genera un recuadro en cada una de las caras detectadas
  - Área de trabajo: esta opción realiza un recorte en la imagen de las caras detectadas
  - Escala de grises: esta opción convierte las imágenes detectadas en escala de gris
- 1 botón llamado “Base De Datos”: el cual redirecciona al operador a 2 interfase.
- 1 etiqueta de información: con esta, se logra visualizar la información del usuario (La cual incluye los siguientes datos: nombre, edad, domicilio, y antecedentes)
- 1 etiqueta de tiempo de ejecución. Con esta, se obtiene el tiempo que le toma al software realizar la identificación.
- 1 botón llamado “Salir”, el cual culmina la sesión y cierra todas las ventanas abiertas.

Imagen 15. Primera Interface



Fuente: Elaboración propia

## Segunda interfaz

Para que exista la posibilidad de interactuar con el usuario se debe establecer una interfaz que muestre la Base de Datos Existente, la cual despliega la información de acuerdo con el personal cargado. Un ejemplo de la interfaz de base de datos se muestra en la Imagen 16:

Imagen 16. Segunda Interface



Fuente: Elaboración propia

Con esta interfaz se logra la visualización de las diferentes características de la Base de Datos y posee los siguientes elementos:

- 1 botón llamado “Características”, con el cual, se expande un menú que contiene los siguientes elementos:
  - Base de Datos: con esta opción, se revisa el conjunto de imágenes que conforman la base de datos.
  - Normalizar: con esta opción se ejecuta el normalizado de las imágenes existentes con la finalidad de reducir los errores en ellas.
  - Media: con esta opción se recupera obtiene la media de las imágenes contenidas en el sistema.
  - Eigenfaces: esta opción sirve para revisar los vectores y valores propios de las imágenes
- 1 botón llamado “Registrar”: esta opción realiza una revisión de la asistencia del personal identificado.
- 1 botón “Terminar” con el cual se cierra la interfaz actual para regresar a la principal.

### **Tercera interface:**

Se debe crear una tercera interfaz, la cual, permite ingresar y guardar datos de un nuevo operador. De acuerdo a los cambios y rotaciones del personal se empleará dicha interfaz (**Imagen 17**). Esta interfaz se compone de:

- 1 botón llamado “Guardar Datos”, este, guarda en el sistema los datos que hayan sido ingresados
- Etiqueta de datos: esta es la sección de la interfase en la cual se realiza el registro de los datos asociados al nuevo usuario
- 1 botón “Terminar”: con el cual, se cierra la ventana de la interface y se retorna a la interfaz principal.

**Imagen 17.** Tercera Interface

The image shows a user registration form within a rectangular frame. On the left side, there is a button labeled "Crear Nuevo Usuario" and below it, a blue button labeled "Guardar Datos". On the right side, there are three input fields: "Nombre:" followed by a text box, "Edad:" followed by a text box, and "Domicilio:" followed by a text box. Below the "Domicilio" field is a longer, empty text box. At the bottom center of the frame is a blue button labeled "Terminar".

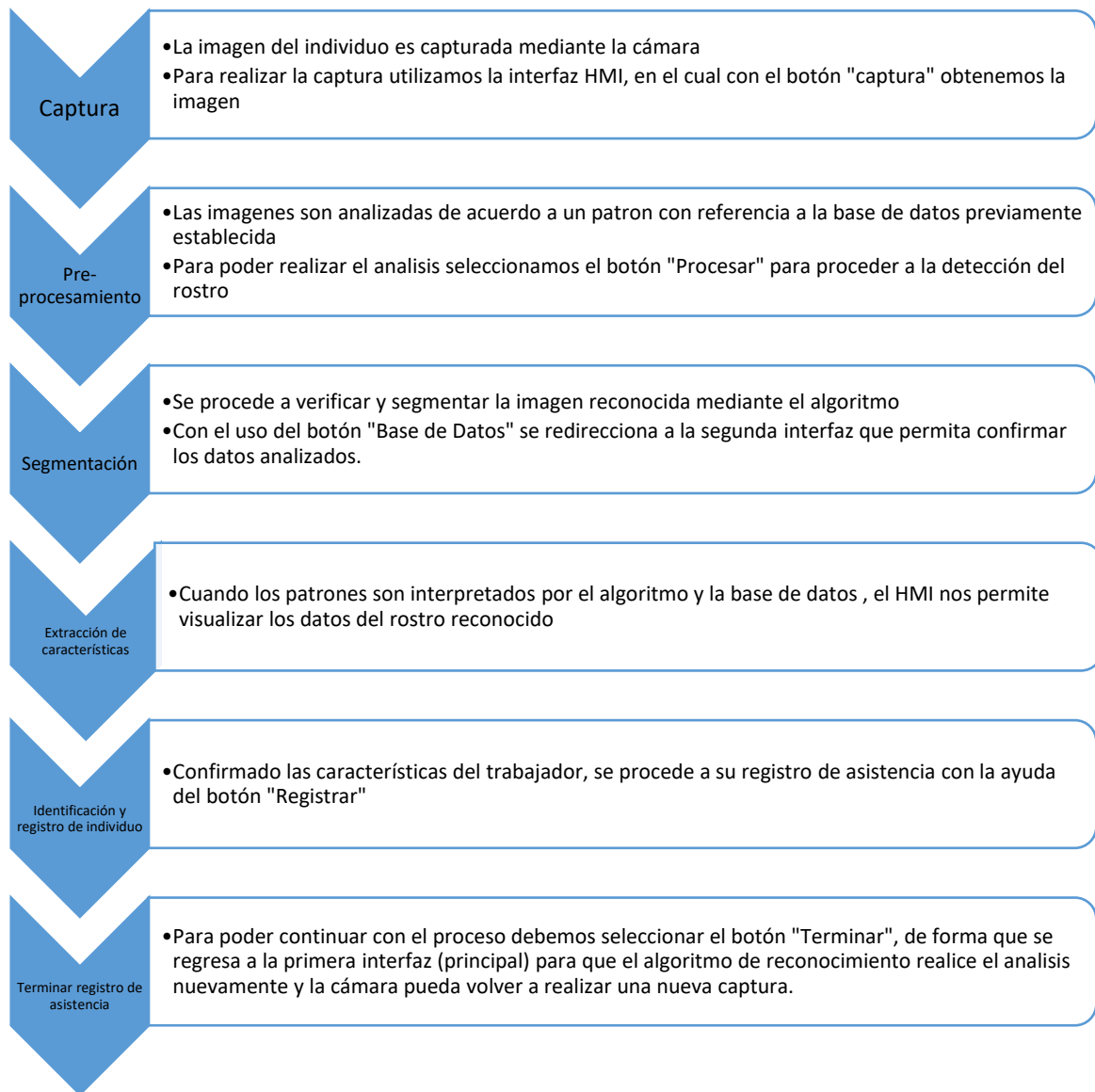
Fuente: elaboración propia

#### **4.2.7. Diseño de la estrategia para el Match entre el algoritmo de reconocimiento facial y el HMI**

El algoritmo de reconocimiento facial está programado para que ejecute una serie de tareas que comienzan con la detección de la imagen, luego, son realizadas varias secuencias de acciones que involucran a la base de datos que almacena a las imágenes y concluyen con una salida específica en las interfaces creadas. El diagrama de los procesos involucrados en la estrategia para el Match se puede observar en la Imagen 18:



Imagen 18. Diagrama de los procesos involucrados en la estrategia para el Match



Fuente: Elaboración propia

### 4.3. CONFIGURACIÓN DEL SISTEMA DE VIDEO EMPLEADO EN EL RECONOCIMIENTO FACIAL

Se procede a indicar la configuración del sistema de video, ya que es la base para que el reconocimiento facial se pueda llevar a cabo, en este apartado se indica la conexión del mismo, así como la programación del script propuesto.

#### **4.3.1. Configuración inicial y conexión entre sistema de captura de video y la base de datos local.**

La configuración inicial se realiza mediante el PCA-Eigenfaces para reconocimiento facial. El PCA (Análisis de componentes principales) permite reducir la cantidad de dimensiones y construye grupos de vectores.

Este algoritmo disminuye las dimensiones de las variables o cualquier otro dato que sea incluido en el programa al momento de la captura de las imágenes. De manera general, las operaciones que realiza el PCA son:

- Transforma de manera lineal todos y cada uno de los datos obtenidos, en una matriz de coordenadas ortogonales.
- Las variables resultantes, de proyectar datos de entrada sobre los vectores de las imágenes obtenidas se denominan componentes principales.
- Permite bajar la dimensión de los datos resultantes, el cual tiene una menor varianza

Al tenerse una imagen con  $n \times n$  pixeles, esta se transforma en una matriz en la cual, a cada uno de los pixeles se le asigna un valor que va desde 0 a 255 y que representa la intensidad de los mismos tras la conversión de la imagen de un formato de tipo RGB a otro de escala de grises. Son estos los datos a ser ingresados en el software y en algoritmo de reconocimiento facial.

#### **4.3.2. Programación del script para poder realizar el Deep learning con la base de datos de rostros establecida.**

Para esta etapa del proceso de implementación se debe adoptar el script de reconocimiento facial creado para esta propuesta, el mismo, se puede observar en el anexo A de manera íntegra.

## 4.4.CONFIGURACIÓN DE LA PLATAFORMA IOT

Una de las plataformas a ser empleadas es Google Cloud IoT core (Google Cloud Platform). Google Cloud permite ejecutar modelos tensoriales en las conocidas como TPU (Tensor Processing Unit) en la nube, lo cual permite manejar grandes volúmenes de datos.

Google es además creador de librerías tensor Flow para poder aplicar técnicas de Deep Learning como las que están siendo aplicadas en este proyecto mediante lenguaje (Python, Java, C++, Go, Rust). A nivel específico de IoT Google permite actuar como un bróker MQTT para el ingreso de datos en los diversos dispositivos, que pueden ser procesador por Cloud Dataflow, así como los servicios Machine Learning as a Service.

### 4.4.1. Registro de credenciales y Generación de usuarios

Las personas que deben tener acceso a la plataforma son los supervisores encargados de llevar a cabo el registro del personal en campo y jefe de departamento de recursos Humanos para poder descargarse la información de los datos registrados y de esta manera realizar el proceso de pago pertinente de acuerdo con las horas laboradas y llevar un control de asistencia.

## 5. CONCLUSIONES Y TRABAJO FUTURO

En la introducción de la presente propuesta de implementación, se planteó la necesidad de que las empresas en la actualidad deben de una manera u otra enfocarse en la modernización de sus procesos operativos y que esta opción va de la mano con la progresiva adaptación del sector empresarial con los preceptos y las bondades de la última de las revoluciones industriales, la industria 4.0.

Esta revolución, como se indicó en los acápites previos y, como particularmente se resaltó en la introducción de este manuscrito, se caracteriza por la digitalización de la mayoría de los procesos que las empresas poseen, esta digitalización no se circunscribe únicamente a la automatización de los procesos productivos, va más allá, y toca aspectos tan esenciales como los procesos de índole operativo, entre los cuales, se encuentran los relacionados con los recursos humanos de la empresa.

En el caso particular que compete a esta investigación, el problema que se planteó fue la carencia de un proceso para lograr automatizar y digitalizar la asistencia del personal de campo de una empresa en Ecuador, dicha empresa, posee una instalación operativa situada en una zona rural con difícil o escaso acceso a herramientas como el internet, además que por ser un emplazamiento alejado de la sede principal de la compañía, se dificulta controlar efectivamente la asistencia del personal a sus jornadas laborales, para dicho control, la empresa en la que se centra este estudio, en la actualidad cuenta con un anticuado sistema de registro por fichas que semanalmente es recogido y enviado en un largo viaje a las instalaciones de la empresa en la ciudad de Quito para su verificación y registro.

Esta situación favorece que en ocasiones el personal no asista a su jornada laboral o simplemente se retire antes de tiempo, aprovechándose de otros compañeros de trabajo para que realicen los correspondientes chequeos de ingreso y salida, lo que provoca pérdidas para la empresa, ya que la misma cancela jornadas de trabajo completas a personas que no cumplen todo el horario diario o simplemente no se presentan.

De esta manera, el desarrollo de la presente propuesta de implementación orientada en la digitalización del proceso de chequeo del personal a través de un sistema de reconocimiento facial que realice dicha tarea e organice la información para ser enviada al personal de

recursos humanos ubicado en otra ciudad, de una manera más rápida y efectiva usando un servidor en la nube, resuelve el problema en los procesos de chequeo, la pérdida de dinero y efectivizaría los procesos operativos que debe ejecutar el departamento de recurso humanos de la empresa.

En base a lo anteriormente expuesto, se tiene que las principales contribuciones aportadas con la presente propuesta de implementación son las siguientes:

1. Identificación de los requerimientos técnicos de software y hardware requeridos por la empresa para la implementación de un sistema digitalizado de reconocimiento facial para el registro del personal
2. Creación de scripts y algoritmos de reconocimiento facial que se requieren para esta implementación
3. Descripción del diseño de las interfaces que requerirá el sistema
4. Generación de una guía que permita la implementación del sistema de reconocimiento facial propuesto.

Considerando los anteriores resultados o alcances logrados, el autor del presente manuscrito, considera que fueron alcanzados los objetivos planteados en este proyecto ya que:

- Objetivo específico 1.: “Evaluar las capacidades y necesidades de la empresa en cuanto a requerimientos técnicos para la implementación de un SRF basado en un dispositivo móvil conectado a una plataforma IoT”. Este objetivo se completó al lograr determinar las carencias y limitaciones técnicas y ambientales de la empresa, que son esenciales subsanar o abordar para implementar el sistema de reconocimiento facial para el chequeo del personal.

La visualización de estas carencias o necesidades permitió establecer que hardware y las condiciones que son requeridos para esta implementación. Esto quedo expuesto en la sección 3 de este manuscrito, en los que, se listaron los requerimientos de hardware y software necesario en base a las tecnologías disponibles y a la observación in situ del entorno donde dicho sistema de reconocimiento facial se emplazaría, además de las especificaciones particulares que se describieron en el capítulo 4 sobre cómo abordar por etapas estas situaciones

- Objetivo específico 2.: “Diseñar un plan de acción para el desarrollo por parte de la empresa, de un SRF para el chequeo del personal de trabajo”. este objetivo se completó y se encuentra detallado en el capítulo 4 de manera específica, en este, se presenta de manera sistematizada y en orden los aspectos a concretar para la instalación del sistema, en los aspectos que van más allá del software requerido, si listan en ese capítulo las condiciones ambientales que deben asegurarse, la forma en cómo se deben programar y emplear los scripts y algoritmos de inteligencia artificial (mismo que se incluye en el anexo 1 de este manuscrito), las características y formas en que se deben presentar las interfaces del sistema y la manera de como configurar la plataforma IOT para la carga de los datos cuando el sistema se encuentre en funcionamiento.

Por lo antes expuesto, es posible concluir que el objetivo principal “Crear una propuesta de implementación de un SRF usado en el registro de asistencia de personal en campo basado en la inteligencia artificial”, ha sido alcanzado tras la realización de este TFM.

## Referencias bibliográficas

- Abate, A. F., Nappi, M., Riccio, D., & Sabatino, G. (2007). 2D and 3D face recognition: A survey. *Pattern Recognition Letters*, 28(14), 1885-1906. <https://doi.org/10.1016/j.patrec.2006.12.018>
- Agrawal, A. K., & Singh, Y. N. (2015). Evaluation of Face Recognition Methods in Unconstrained Environments. *Procedia Computer Science*, 48, 644-651. <https://doi.org/10.1016/j.procs.2015.04.147>
- Akbar, A., Kousiouris, G., Pervaiz, H., Sancho, J., Ta-Shma, P., Carrez, F., & Moessner, K. (2018). Real-Time Probabilistic Data Fusion for Large-Scale IoT Applications. *IEEE Access*, 6, 10015-10027. <https://doi.org/10.1109/ACCESS.2018.2804623>
- AL-Allaf, O. N. A. (2014). Review of Face Detection Systems Based Artificial Neural Networks Algorithms. *The International journal of Multimedia & Its Applications*, 6(1), 1-16. <https://doi.org/10.5121/ijma.2013.6101>
- Aldana Porras, J. M. (2018). *Diseño de un sistema de reconocimiento facial como medio de control de acceso biométrico mediado por técnicas de inteligencia artificial como herramienta base de seguridad del Cead Ibagué*. <http://repository.unad.edu.co/handle/10596/22930>
- Alfalou, A., & Brosseau, C. (2011). Robust and discriminating method for face recognition based on correlation technique and independent component analysis model. *Optics Letters*, 36(5), 645-647. <https://doi.org/10.1364/OL.36.000645>
- Almeida, N. (2017). *Facial Recognition System applied to Multipurpose Assistance robot for Social Human-robot Interaction (MASHI)* [Tesis]. Escuela Técnica Superior de Ingeniería Industrial de Barcelona.
- Amazon. (2022). *Capa gratuita de AWS | Cloud computing gratis | AWS*. Amazon Web Services, Inc. <https://aws.amazon.com/es/free/>
- APD. (2019, marzo 6). Big data qué es y para qué sirve en el desarrollo de las empresas [Blog especializado]. *APD España*. <https://www.apd.es/big-data-que-es-y-para-que-sirve/>

- Awoamin, J., & Brand, D. (2020). Impact of Artificial Neural Networks Training Algorithms on Accurate Prediction of Property Values. *Journal of Real Estate Research*, 40(3), 375-418. <https://doi.org/10.1080/10835547.2018.12091505>
- Bader, S., Maleshkova, M., & Lohmann, S. (2019). Structuring Reference Architectures for the Industrial Internet of Things. *Future Internet*, 11, 151. <https://doi.org/10.3390/fi11070151>
- Bastidas Gavilanes, J. R. (2018). *REGISTRO DE ASISTENCIA DE ALUMNOS POR MEDIO DE RECONOCIMIENTO FACIAL UTILIZANDO VISIÓN ARTIFICIAL* [Universidad Técnica de Ambato]. [https://repositorio.uta.edu.ec/bitstream/123456789/29179/1/Tesis\\_t1532masc.pdf](https://repositorio.uta.edu.ec/bitstream/123456789/29179/1/Tesis_t1532masc.pdf)
- Benammar, M., Abdaoui, A., Ahmad, S. H. M., Touati, F., & Kadri, A. (2018). A Modular IoT Platform for Real-Time Indoor Air Quality Monitoring. *Sensors*, 18(2), 581. <https://doi.org/10.3390/s18020581>
- Bisong, E. (2019). *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Apress Berkeley. <https://link.springer.com/book/10.1007/978-1-4842-4470-8>
- Bondarouk, T., & Brewster, C. (2016). Conceptualising the future of HRM and technology research. *The International Journal of Human Resource Management*, 27(21), 2652-2671. <https://doi.org/10.1080/09585192.2016.1232296>
- Bravo, C. J., Ramírez, P. E., Arenas, J., Bravo, C. J., Ramírez, P. E., & Arenas, J. (2018). Aceptación del Reconocimiento Facial Como Medida de Vigilancia y Seguridad: Un Estudio Empírico en Chile. *Información tecnológica*, 29(2), 115-122. <https://doi.org/10.4067/S0718-07642018000200115>
- Brunelli, R., & Poggio, T. (1993). Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10), 1042-1052. <https://doi.org/10.1109/34.254061>
- Cabas-Torres, D. A. (2019). *Diseño y construcción de un prototipo USV para el reconocimiento y geoposicionamiento de residuos sólidos flotantes*. <https://reunir.unir.net/handle/123456789/9846>
- Carlos Suárez. (2019). Aspectos Básicos de la Industria 4.0. *Ministerio de Tecnologías de la Información y las comunicaciones*, 43.



Carrión, R., & Yandry, D. (2022). *Desarrollo de una aplicación para registro de asistencia mediante técnicas inteligentes como reconocimiento facial, reconocimiento del hablante y geolocalización.* 52.

Cevikalp, H., Neamtu, M., Wilkes, M., & Barkana, A. (2005). Discriminative common vectors for face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1), 4-13. <https://doi.org/10.1109/TPAMI.2005.9>

Chacua Criollo, B. E. (2019). *Diseño de un sistema prototipo de reconocimiento facial para la identificación de personas en la Facultad de Ingeniería en Ciencias Aplicadas (FICA) de la Universidad Técnica del Norte utilizando técnicas de Inteligencia Artificial.* <http://repositorio.utn.edu.ec/handle/123456789/9572>

Chang, W.-J., Schmelzer, M., Kopp, F., Hsu, C.-H., Su, J.-P., Chen, L.-B., & Chen, M.-C. (2019). A Deep Learning Facial Expression Recognition based Scoring System for Restaurants. *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*, 251-254. <https://doi.org/10.1109/ICAIIIC.2019.8668998>

Chen, J., & Ran, X. (2019). Deep Learning With Edge Computing: A Review. *Proceedings of the IEEE*, 107(8), 1655-1674. <https://doi.org/10.1109/JPROC.2019.2921977>

Dagher, I., & Nachar, R. (2006). Face recognition using IPCA-ICA algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(6), 996-1000. <https://doi.org/10.1109/TPAMI.2006.118>

Derechos Digitales. (2021, noviembre 8). *Reconocimiento facial en América Latina.* <https://estudio.reconocimientofacial.info/>

Draper, B., Baek, K., Bartlett, M., & Beveridge, J. (2003). Recognizing faces with PCA and ICA. *Comput. Vis. Image Underst.* [https://doi.org/10.1016/S1077-3142\(03\)00077-8](https://doi.org/10.1016/S1077-3142(03)00077-8)

Duocom Europe. (2022). *Reloj checador para calcular horas laboradas y trabajo en sobretiempo.* [Web Institucional]. Reloj checador. [https://www.relojlaboral.com/reloj-checador-digital.php?gclid=Cj0KCQjw4PKTBhD8ARIsAHChzRKj107ZJiv1aGj3VRVYeK3fMeynWJ1q9WcVExGVkqTjQhvSwSOGg1oaAmneEALw\\_wcB](https://www.relojlaboral.com/reloj-checador-digital.php?gclid=Cj0KCQjw4PKTBhD8ARIsAHChzRKj107ZJiv1aGj3VRVYeK3fMeynWJ1q9WcVExGVkqTjQhvSwSOGg1oaAmneEALw_wcB)

Efremov, S., Pilipenko, N., & Voskov, L. (2015). An Integrated Approach to Common Problems in the Internet of Things. *Procedia Engineering*, 100. <https://doi.org/10.1016/j.proeng.2015.01.486>

EPICOR. (2022). *¿Qué es la Industria 4.0: La Internet Industrial de las Cosas (IIoT)?* [Blog especializado]. <https://www.epicor.com/es-co/resources/articles/what-is-industry-4-0/>

Er, M. J., Wu, S., Lu, J., & Toh, H. L. (2002). Face recognition with radial basis function (RBF) neural networks. *IEEE Transactions on Neural Networks*, 13(3), 697-710. <https://doi.org/10.1109/TNN.2002.1000134>

Espinoza Olgún, D. E., & Jorquera Guillén, P. I. (2015). *Reconocimiento facial* [Pontificia Universidad Católica de Valparaíso]. [http://opac.pucv.cl/pucv\\_txt/txt-1000/UCD1453\\_01.pdf](http://opac.pucv.cl/pucv_txt/txt-1000/UCD1453_01.pdf)

Evans, D. (2011). *Cómo la próxima evolución de Internet lo cambia todo*. 12.

Fernández, Y. (2021, enero 8). *Big Data: Qué es y para que sirve*. Xataka. <https://www.xataka.com/basics/big-data-que-sirve>

Fortino, G., Savaglio, C., Palau, C. E., de Puga, J. S., Ganzha, M., Paprzycki, M., Montesinos, M., Liotta, A., & Llop, M. (2018). Towards Multi-layer Interoperability of Heterogeneous IoT Platforms: The INTER-IoT Approach. En R. Gravina, C. E. Palau, M. Manso, A. Liotta, & G. Fortino (Eds.), *Integration, Interconnection, and Interoperability of IoT Systems* (pp. 199-232). Springer International Publishing. [https://doi.org/10.1007/978-3-319-61300-0\\_10](https://doi.org/10.1007/978-3-319-61300-0_10)

Gaber, M. M., Aneiba, A., Basurra, S., Batty, O., Elmisery, A. M., Kovalchuk, Y., & Rehman, M. H. U. (2019). Internet of Things and data mining: From applications to techniques and systems. *WIREs Data Mining and Knowledge Discovery*, 9(3), e1292. <https://doi.org/10.1002/widm.1292>

Garcés Nuñez, A. F. (2017). *"SISTEMA DE RECONOCIMIENTO FACIAL CON VISIÓN ARTIFICIAL PARA APOYAR AL ECU-911 CON LA IDENTIFICACIÓN DE PERSONAS EN LA LISTA DE LOS MÁS BUSCADOS"*. Universidad Técnica de Ambato.

Genius. (2020). *ECam-8000* [Web de empresa]. ECAM 8000. <https://pe.geniusnet.com/product/ecam-8000/>

- Goncalves, R., Romero, D., & Grilo, A. (2017). Factories of the Future: Challenges and Leading Innovations in Intelligent Manufacturing. *International Journal of Computer Integrated Manufacturing*, 30, 4-13. <https://doi.org/10.1080/0951192X.2016.1258120>
- Gonzalez, H., & Velásquez, S. (2019). RECONOCIMIENTO FACIAL UTILIZANDO VIOLA-JONES Y PATRONES BINARIOS. *UNIVERSIDAD, CIENCIA y TECNOLOGÍA*, 23(92), 57-63.
- González-Muñiz, A. (2018). *Aplicaciones de técnicas de inteligencia artificial basadas en aprendizaje profundo (deep learning) al análisis y mejora de la eficiencia de procesos industriales.* <https://doi.org/10.13140/RG.2.2.15144.72967>
- Gualdrón, O. E., & Suárez, O. M. D. (2014). DISEÑO DE UN SISTEMA DE RECONOCIMIENTO DE ROSTROS APLICANDO INTELIGENCIA Y VISIÓN ARTIFICIAL. *REVISTA COLOMBIANA DE TECNOLOGIAS DE AVANZADA (RCTA)*, 2(24), 117-126.
- Hamrioui, S., & Lorenz, P. (2017). Bio Inspired Routing Algorithm and Efficient Communications within IoT. *IEEE Network*, 31, 74-79. <https://doi.org/10.1109/MNET.2017.1600282>
- Hejazi, H., Rajab, H., Cinkler, T., & Lengyel, L. (2018). *Survey of platforms for massive IoT.* 1-8. <https://doi.org/10.1109/FIOT.2018.8325598>
- Hernández, R. G. (2010). *ESTUDIO DE TÉCNICAS DE RECONOCIMIENTO FACIAL.* 86.
- Hussain, F., Hussain, R., Hassan, S., & Hossain, E. (2020). Machine Learning in IoT Security: Current Solutions and Future Challenges. *IEEE Communications Surveys & Tutorials*, PP. <https://doi.org/10.1109/COMST.2020.2986444>
- Intrator, N., Reifeld, D., & Yeshurun, Y. (1996). Face recognition using a hybrid supervised/unsupervised neural network. *Pattern Recognition Letters*, 17(1), 67-76. [https://doi.org/10.1016/0167-8655\(95\)00089-5](https://doi.org/10.1016/0167-8655(95)00089-5)
- IoT Hardware Development Platforms: Past, Present, and Future. (2017). En *Internet of Things* (pp. 107-139). Chapman and Hall/CRC. <https://doi.org/10.1201/9781315155005-6>
- Iqtait, M., Mohamad, F. S., & Mamat, M. (2018). Feature extraction for face recognition via Active Shape Model (ASM) and Active Appearance Model (AAM). *IOP Conference Series: Materials Science and Engineering*, 332, 012032. <https://doi.org/10.1088/1757-899X/332/1/012032>

- Jafri, R., & Arabnia, H. R. (2009). A Survey of Face Recognition Techniques. *Journal of Information Processing Systems*, 5(2), 41-68. <https://doi.org/10.3745/JIPS.2009.5.2.041>
- Johnston, B., & Chazal, P. de. (2018). A review of image-based automatic facial landmark identification techniques. *EURASIP Journal on Image and Video Processing*, 2018(1), 86. <https://doi.org/10.1186/s13640-018-0324-4>
- Kakade, S. D. (2016). *A Review Paper on Face Recognition Techniques*. 02(02), 4.
- Kar, A., Bhattacharjee, D., Basu, D. K., Nasipuri, M., & Kundu, M. (2011). *High Performance Human Face Recognition using Independent High Intensity Gabor Wavelet Responses: A Statistical Approach* (arXiv:1106.3467). arXiv. <https://doi.org/10.48550/arXiv.1106.3467>
- Kumar, A., Kaur, A., & Kumar, M. (2019). Face detection techniques: A review. *Artificial Intelligence Review*, 52(2), 927-948. <https://doi.org/10.1007/s10462-018-9650-2>
- Lee, J. (2017). *US Patent Application for Internet of things based Deadbolt lock latch strike location smart sensor Patent Application (Application #20170024993 issued January 26, 2017)—Justia Patents Search (Patent N.º 20170024993)*. <https://patents.justia.com/patent/20170024993>
- Li, X., & Da, F. (2012). Efficient 3D face recognition handling facial expression and hair occlusion. *Image and Vision Computing*, 30, 668-679. <https://doi.org/10.1016/j.imavis.2012.07.011>
- Lin, S.-H., Kung, S.-Y., & Lin, L.-J. (1997). Face recognition/detection by probabilistic decision-based neural network. *IEEE Transactions on Neural Networks*, 8(1), 114-132. <https://doi.org/10.1109/72.554196>
- Liu, X., Wang, Y., Ji, J., Cheng, H., Zhu, X., Awa, E., He, P., Chen, W., Poon, H., Cao, G., & Gao, J. (2020). *The Microsoft Toolkit of Multi-Task Deep Neural Networks for Natural Language Understanding* (arXiv:2002.07972). arXiv. <https://doi.org/10.48550/arXiv.2002.07972>
- López-Sánchez, M. (2022). *Introducción a los sistemas de reconocimiento facial utilizando Deep learning*.
- Mankar, V., & Bhele, S. (2012). A Review Paper on Face Recognition Techniques. *International Journal of Advanced Research in Computer Engineering & Technology*, 1, 339-346.

- Marjani, M., Nasaruddin, F., Gani, A., Karim, A., Hashem, I. A. T., Siddiqa, A., & Yaqoob, I. (2017). Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges. *IEEE Access*, 5, 5247-5261. <https://doi.org/10.1109/ACCESS.2017.2689040>
- Marqués, I. (2010). *Face Recognition Algorithms* [Tesis de maestría, Universidad del País Vasco]. <https://www.ehu.es/ccwintco/uploads/e/eb/PFC-IonMarques.pdf>
- Matias, G., Martinez, B., & Stegmayer, D. (2010). *Sistema de reconocimiento facial*. 74.
- Morcillo, F. (2014). *Estudio de las variables a considerar para el desarrollo de un modelo viable de aplicación de un sistema de identificación facial segura en un entorno de organizaciones de servicios* [Tesis, Universitat Politècnica de València]. <http://polipapers.upv.es/index.php/IA/article/view/3293>
- Morrar, R., Arman, H., & Mousa, S. (2017). The Fourth Industrial Revolution (Industry 4.0): A Social Innovation Perspective. *Technology Innovation Management Review*, 7(11), 12-20. <https://doi.org/10.22215/timreview/1117>
- Muskopf-Stone, N., Votta, G., Van Essen, J., Peregrino, A., & Khan Mohd, T. (2022). Facial Recognition Technologies: A Survey and Comparison of Systems and Practical Applications. En J.-H. Kim, M. Singh, J. Khan, U. S. Tiwary, M. Sur, & D. Singh (Eds.), *Intelligent Human Computer Interaction* (pp. 50-61). Springer International Publishing. [https://doi.org/10.1007/978-3-030-98404-5\\_5](https://doi.org/10.1007/978-3-030-98404-5_5)
- Nasution, A., Bayu, D. B. S., & Miura, J. (2014). Person identification by face recognition on portable device for teaching-aid system: Preliminary report. *2014 International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA)*. <https://doi.org/10.1109/ICAICTA.2014.7005935>
- Niola Quito, C. X., & Sanango Zhinin, W. A. (2019). *Desarrollo de un software de seguridad para detección y reconocimiento facial basado en los algoritmos de Viola-Jones y PCA Eigenface* [Tesis, Universidad Politécnica Salesiana]. <http://dspace.ups.edu.ec/handle/123456789/17171>
- Orange. (2021). *Aplicaciones de reconocimiento facial curiosas e inesperadas* [Blog especializado]. Innovación. <https://blog.orange.es/innovacion/aplicaciones-reconocimiento-facial/>

Paraskevoudis, K., Karayannis, P., & Koumoulos, E. P. (2020). Real-Time 3D Printing Remote Defect Detection (Stringing) with Computer Vision and Artificial Intelligence. *Processes*, 8(11), 1464. <https://doi.org/10.3390/pr8111464>

Park, J., Naumov, M., Basu, P., Deng, S., Kalaiah, A., Khudia, D., Law, J., Malani, P., Malevich, A., Nadathur, S., Pino, J., Schatz, M., Sidorov, A., Sivakumar, V., Tulloch, A., Wang, X., Wu, Y., Yuen, H., Diril, U., ... Smelyanskiy, M. (2018). *Deep Learning Inference in Facebook Data Centers: Characterization, Performance Optimizations and Hardware Implications* (arXiv:1811.09886). arXiv. <https://doi.org/10.48550/arXiv.1811.09886>

Perasso, V. (2016, octubre 12). Qué es la cuarta revolución industrial (y por qué debería preocuparnos). *BBC News Mundo*. <https://www.bbc.com/mundo/noticias-37631834>

Petrov, V., Samuylov, A. K., Begishev, V., Moltchanov, D., Andreev, S., Samouylov, K., & Koucheryavy, Y. (2018). Vehicle-Based Relay Assistance for Opportunistic Crowdsensing Over Narrowband IoT (NB-IoT). *IEEE Internet of Things Journal*. <https://doi.org/10.1109/JIOT.2017.2670363>

Platero, D. (2015). *RECONOCIMIENTO DE IMÁGENES FACIALES ORIENTADO A CONTROLES DE ACCESO Y SISTEMAS DE SEGURIDAD*. [Tesis, UNIVERSIDAD DISTRITAL “FRANCISCO JOSÉ DE CALDAS”].

<https://repository.udistrital.edu.co/bitstream/handle/11349/7359/PlateroPlazasDonovanCamilo2015.pdf;jsessionid=028EB55A0F6D59A374AAE0AC2D8F9070?sequence=1>

Rahman, L. F., Ozcelebi, T., & Lukkien, J. (2018). Understanding IoT Systems: A Life Cycle Approach. *Procedia Computer Science*, 130, 1057-1062. <https://doi.org/10.1016/j.procs.2018.04.148>

Ramírez Madrid, D. A., & Rodríguez Hernández, E. D. (2016). *El Internet de las cosas*. <https://repository.udistrital.edu.co/bitstream/handle/11349/5343/RamirezMadridDavidAndres2017.pdf;jsessionid=7BDC9E5E7408B81048F451E7306C2108?sequence=1>

Rivas, M. (2022, abril 7). Reconocimiento facial: Cómo controlar la asistencia en empresas. *Sesame HR*. <https://www.sesamehr.mx/blog/reconocimiento-facial-como-herramienta-para-el-control-de-asistencia-en-las-empresas/>

Rivas Perea, P., Velarde Anaya, O., & Valencia Olvera, L. (2016). *Reconocimiento facial en ambientes no cooperativos* [Instituto Tecnológico de Chiguagua].

- [https://www.academia.edu/967012/Reconocimiento\\_Facial\\_en\\_Ambientes\\_No\\_Cooperativos\\_Mediante\\_Hough\\_KLT\\_en\\_Tiempo\\_Real](https://www.academia.edu/967012/Reconocimiento_Facial_en_Ambientes_No_Cooperativos_Mediante_Hough_KLT_en_Tiempo_Real)
- Roman, V. (2019, abril 15). Introduccion al Machine Learning: Una Guía Desde Cero. *Ciencia y Datos*. <https://medium.com/datos-y-ciencia/introduccion-al-machine-learning-una-gu%C3%ADa-desde-cero-b696a2ead359>
- Rosas, J., Brito, V., Palma, L. B., & Barata, J. (2017). Approach to Adapt a Legacy Manufacturing System Into the IoT Paradigm. *International Journal of Interactive Mobile Technologies (IJIM)*, 11(5), 91-104. <https://doi.org/10.3991/ijim.v11i5.7073>
- Rowley, H. A., Baluja, S., & Kanade, T. (1998). Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1), 23-38. <https://doi.org/10.1109/34.655647>
- Sahni, Y., Cao, J., Zhang, S., & Yang, L. (2017). Edge Mesh: A New Paradigm to Enable Distributed Intelligence in Internet of Things. *IEEE Access*, 5, 16441-16458. <https://doi.org/10.1109/ACCESS.2017.2739804>
- Samadiani, N., Huang, G., Cai, B., Luo, W., Chi, C.-H., Xiang, Y., & He, J. (2019). A Review on Automatic Facial Expression Recognition Systems Assisted by Multimodal Sensor Data. *Sensors*, 19(8), 1863. <https://doi.org/10.3390/s19081863>
- Schwab, K. (2016). *The Fourth Industrial Revolution*, by. World Economic Forum. [https://law.unimelb.edu.au/\\_\\_data/assets/pdf\\_file/0005/3385454/Schwab-The\\_Fourth\\_Industrial\\_Revolution\\_Klaus\\_S.pdf](https://law.unimelb.edu.au/__data/assets/pdf_file/0005/3385454/Schwab-The_Fourth_Industrial_Revolution_Klaus_S.pdf)
- Sharma, M., Prakash, S., & Gupta, P. (2013). An efficient partial occluded face recognition system. *Neurocomputing*, 116, 231-241. <https://doi.org/10.1016/j.neucom.2011.12.063>
- Shaw, P., & Varghese, R. M. (2018). INDUSTRY 4.0 AND FUTURE OF HR. *Journal of Management*, 5(6), 8.
- Singh, C. B., Sarkar, B., & Yadav, P. (2021). *Facial Expression Recognition* (SSRN Scholarly Paper N.º 3852666). Social Science Research Network. <https://doi.org/10.2139/ssrn.3852666>
- Singh, M., & Arora, A. S. (2016). Varying Illumination and Pose Conditions in Face Recognition. *Procedia Computer Science*, 85, 691-695. <https://doi.org/10.1016/j.procs.2016.05.255>

Sivathanu, B., & Pillai, R. (2018). Smart HR 4.0 – how industry 4.0 is disrupting HR. *Human Resource Management International Digest*, 26(4), 7-11. <https://doi.org/10.1108/HRMID-04-2018-0059>

Tivive, F. H. C., & Bouzerdoum, A. (2003). A new class of convolutional neural networks (SiCoNNets) and their application of face detection. *Proceedings of the International Joint Conference on Neural Networks, 2003.*, 3, 2157-2162 vol.3. <https://doi.org/10.1109/IJCNN.2003.1223742>

Torres, I. A. G., Borja, X. T., Torre, J. D. D. L., & Espín, W. N. (2021). API para control de asistencia con reconocimiento facial usando OpenCv.JS. *Revista Tecnológica Ciencia y Educación Edwards Deming*, 5(1), Article 1. <http://www.revista-edwardsdeming.com/index.php/es/article/view/67>

Unal, C., & Tecim, V. (2018). The Use of Biometric Technology for Effective Personnel Management System in Organization. *KnE Social Sciences*, 221-232. <https://doi.org/10.18502/kss.v3i10.3540>

Universitat Politècnica de Catalunya. (2013). *Interactive Image Processing Demos for the Web* [Presentacion de PPT]. <https://www.slideshare.net/xavigiro/interactive-image-processing-demos-for-the-web>

Veiga García, J. (2021). *Reconocimiento facial en América Latina: Tendencias en la implementación de una tecnología perversa | LET*. <http://let.iiec.unam.mx/node/3951>

Wanasinghe, T. R., Gosine, R. G., James, L. A., Mann, G. K. I., de Silva, O., & Warriar, P. J. (2020). The Internet of Things in the Oil and Gas Industry: A Systematic Review. *IEEE Internet of Things Journal*, 7(9), 8654-8673. <https://doi.org/10.1109/JIOT.2020.2995617>

Wang, Y., Li, Z., Zeng, C., Xia, G.-S., & Shen, H. (2020). An Urban Water Extraction Method Combining Deep Learning and Google Earth Engine. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 769-782. <https://doi.org/10.1109/JSTARS.2020.2971783>

xmart. (2019, agosto 10). ¿Cómo usar la biometría facial para controlar el personal remoto? *XmartClock*. <https://www.xmartclock.com/blog/la-biometria-facial-para-control-del-personal-remoto/>



Zhang, X., & Gao, Y. (2009). Face recognition across pose: A review. *Pattern Recognition*, 42(11), 2876-2896. <https://doi.org/10.1016/j.patcog.2009.04.017>

Zhao, W., & Chellappa, R. (2002). *Image-based Face Recognition: Issues and Methods 1*. <https://www.semanticscholar.org/paper/Image-based-Face-Recognition%3A-Issues-and-Methods-1-Zhao-Chellappa/dbd5e9691cab2c515b50dda3d0832bea6eef79f2>

Zhou, W., Jia, Y., Yao, Y., Zhu, L., Guan, L., Mao, Y., Liu, P., & Zhang, Y. (2019). *Discovering and Understanding the Security Hazards in the Interactions between IoT Devices, Mobile Apps, and Clouds on Smart Home Platforms | USENIX*. 1133-1150. <https://www.usenix.org/conference/usenixsecurity19/presentation/zhou>

Zhu, Q., Wang, R., Chen, Q., Liu, Y., & Qin, W. (2010). IOT Gateway: Bridging Wireless Sensor Networks into Internet of Things. *2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*, 347-352. <https://doi.org/10.1109/EUC.2010.58>

## Anexo A. Código implementado

### Código a Implementarse para el SRF

#### Guide Principal

```
function varargout = Guide(varargin)
    clc

    % Begin initialization code - DO NOT EDIT

    gui_Singleton = 1;
    gui_State = struct('gui_Name',
        mfilename, ...

        'gui_Singleton', gui_Singleton,
        ...

        'gui_OpeningFcn',
        @Guide_OpeningFcn, ...

        'gui_OutputFcn',
        @Guide_OutputFcn, ...

        'gui_LayoutFcn', [] , ...

        'gui_Callback',
        []);

    if nargin && ischar(varargin{1})
        gui_State.gui_Callback = str2func(varargin{1});
    end

    if nargin
        [varargout{1:nargout}] =
            gui_mainfcn(gui_State, varargin{:});
    else
        gui_mainfcn(gui_State,
            varargin{:});
    end

    % End initialization code - DO NOT EDIT

    % --- Executes just before Guide is
    % made visible.
    function Guide_OpeningFcn(hObject,
        ~, handles, varargin)

        % This function has no outputargs,
        % see OutputFcn.

        % hObject handle to figure

        % eventdata reserved - to be defined
        % in a future version of MATLAB

        % handles structure with handles
        % and user data (see GUIDATA)

        % varargin command line
        % arguments to Guide (see VARARGIN)
        axes(handles.axes8);

        Fondo3 =
            imread(strcat('C:\ProgramaTesis\Fondo\','Fondo1.jpg')); imshow(Fondo3)
```

**Villafuerte Carrillo, Ricardo Patricio. Sistema de reconocimiento facial mediante inteligencia artificial para registro de asistencia de personal en campo.**

```

axes(handles.axes10); % eventdata reserved - to be defined
                        % in a future version of MATLAB

Fondo4 = % handles structure withhandles
         imread(strcat('C:\ProgramaTesis\Fondo4.jpg')); and user data (seeGUIDATA)
         imshow(Fondo4)
axes(handles.axes1); % Get default command line output
                        % from handles structure
Fondo1 = varargout{1} = handles.output;
         imread(strcat('C:\ProgramaTesis\Fondo5.jpg'));
         imshow(Fondo1); % --- Ejecuta al presionar el botón
                        % pushbutton1 (Botón Abrir). function
                        % pushbutton1_Callback(hObject,
                        % eventdata, handles)
                        % set(handles.text1,'string','');
Fondo2 = %Borra contenido de text.
         imread(strcat('C:\ProgramaTesis\Fondo6.jpg'));
         imshow(Fondo2);
                        % set(handles.text12,'string','');
                        % set(handles.text14,'string','');
                        % set(handles.text18,'string','');
                        % set(handles.text19,'string','');
                        % set(handles.text20,'string','');
         title('Ingresar usuario');
axes(handles.axes2);

Fondo3 =
         imread(strcat('C:\ProgramaTesis\Fondo3.jpg'));
         imshow(Fondo3);
         title('Buscar usuario');

% Choose default command line
output for Guide
handles.output = hObject;

% Update handles
structureguidata(hObject,
handles);

% UIWAIT makes Guide wait for user
response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function
are returned to the command line.
function varargout =
Guide_OutputFcn(hObject,
eventdata, handles)

% varargout cell array for
returning output args (see
VARARGOUT);

% hObject handle to figure

```

```
set(handles.text21,'string','');
global vid % Define una variable
global
global I

[filename path] =
uigetfile(['*.jpg'],'Seleccionar
Imágen'); % Abre cuadro de diálogo
de selección de archivos.

vid =
imread(strcat(path,filename)); %Lee
la imagen de la carpeta
seleccionada.

axes(handles.axes1); % Selecciona
el eje carteciano para visualizar
la imagen.

imshow(vid); title('Imágen de
entrada');

RGB=vid;

I=RGB(50:694, 500:1400,:);%Recorta
imagen de alta resolución 900..

% --- Ejecuta al presionar elbotón
pushbutton4 (Botón Encender
Cámara).

function
pushbutton4_Callback(hObject,
eventdata, handles)
set(handles.text1,'string','');

%Borra contenido de text.
set(handles.text12,'string','');

set(handles.text14,'string','');

set(handles.text18,'string','');

set(handles.text19,'string','');

set(handles.text20,'string','');
set(handles.text21,'string','');
global vid % Define una variable
global.

axes(handles.axes1); % Selecciona
el eje carteciano para visualizar
la imagen.

vid=videoinput('winvideo',
1,'MJPG_1280x720' ); %Crea un
objeto de entrada de video y
seleccionamos el formato.
vidRes=get(vid,'VideoResolution');

% Muestra las propiedades del
adaptador gráfico 'Resolución'.
nBands=get(vid,'NumberOfBands'); %
Muestra las propiedades del
adaptador gráfico 'Número de
bandas'.

himage=
image(zeros(vidRes(2),vidRes(1),nB
ands)); % Muestra los datos en la
matriz como una imagen.
rectangle('Position',[300 100 680
300],'EdgeColor','b'); % Dibuja un
rectángulo en la pantalla. izq,
arrib, der,abaj preview(vid,himage);
% Visualizar

% --- Ejecuta al presionar elbotón
pushbutton6 (Botón Salir). function
pushbutton6_Callback(hObject,
eventdata, handles)

opc = questdlg('¿Desea salir del
programa?','SALIR','Si','No','No')
```

```

; % Crea un cuadro de diálogo de pregunta
if
    strcmp(opc, 'No
        ')return;
end

delete(handles.figure1); % Cierra
la figura 1 (ventana guide)

% --- Ejecuta al presionar elbotón
pushbutton7 (BotónCapturar).

function
pushbutton7_Callback(hObject,
eventdata, handles)

global
vidglobal
I

vid = getsnapshot(vid); % Devuelve
inmediatamente el cuadro de una
sola imagen.

axes(handles.axes1);

imshow(vid);    title('Imagen
                deentrada');

closepreview; % Cierra la ventana
de vista previa de vídeo.

RGB=vi
d;

I=RGB;

% --- Ejecuta al presionar elbotón
pushbutton9 (BotónProcesar).

function
pushbutton9_Callback(hObject,
eventdata, handles)

tic                % Toma el
tiempo de ejecución.

global I           % Contiene
la imagen del usuario a identificar.
global IFaces     % Contiene
la imagen con la detección facial-
global work_area  % Contiene la
imagen facial recortada.

global img        % Contiene la
imagen de la media.

global ReshapedImage % Contiene la
imagen reconstruida.

global M
tic;

a ='C:\ProgramaTesis\BaseDeDatos';

% Dirección de la carpeta Base de
Datos

A =dir( fullfile(a, '*.jpg')); %
Muestra lista de contenidos de la
carpeta.

M=size(A,1);           %
Número de imágenes en la base de
datos.

% Para normalizar las imágenes,
definimos ustd (desviación estándar)
y um (media) de tal manera que
% su valor se encuentre cerca de la
media y la desviación estándar de la
mayoría de las imágenes.
ustd=80;um=100;

% Leer imágenes de la base de datos
(.jpg).

```

```

S=[];
for i=1:M
    img =
    imread(strcat('C:\ProgramaTesis\Ba
seDeDatos\',int2str(i),'.jpg')); %
Concatena la dirección de las
imágenes y dos cadenas que forman
el nombre de la imagen.

    [ifil icol]=size(img);
% Obtenemos el número de filas
(N1) y columnas (N2).

    temp=reshape(img',ifil*icol,1); %
Reordena la matriz, crea una matriz
(N1*N2)x1.

    S=[S temp];
% Matriz con todas las imágenes
reordenadas (N1*N2)xM, en una sola
columna los valores de cada foto.
end

    for i=1:2

        axes(handles.axes2); Fondo2
        =

        imread(strcat('C:\ProgramaTesis\Fo
ndo\', 'Fondo3.jpg'));

        imshow(Fondo2);
title('Iniciando Búsqueda');

        end

% Normalizar imágenes de la base
de datos para reducir el error
debido a las condiciones de
iluminación.

for i=1:size(S,2) % Valordel
número de columnas de la matriz S.

    temp=double(S(:,i)); %Almacena
las variables numéricas como
valores de punto flotante.

```

```

img=reshape(mn,icol,ifil); % Toma
el vector N1*N2x1 y crea una
matriz N2xN1

img=img'; % crea
una matriz N1xN2 mediante la
transposición de la imagen.

% Cambio de imagen
por manipulación.
X=[]; % X
matrixfor i=1:M
    temp=double(S(:,i))-m;
    X=[X temp]; %Almacena todos
los temp, todas las restas en una
matriz
end

% Matriz de
CovarianzaA=X';
L=A*A';%MxM, si fuera alreves
seria Ncuadrado x Ncuadrado

% vv vector propio para y dd valor
propio para L=X'*X y C=X*X'
[vv dd]=eig(L); %vectores
evalores , igual dimencion que
covarianza L MxM, dd evalores son
0 excepto diagonal.

% Ordenar y eliminar aquellos cuyo
valor propio es cero
v=[];
d=[];

for i=1:size(vv,2)
    if(dd(i,i)>1e-4) % Sólo
sila diagonal de los dd evalores
es
>1e-4.
        v=[v vv(:,i)]; % Guarda env
columnas de vv cuando las columnas
de dd son >1e-4.
        d=[d dd(i,i)]; % Guarda end
columnas de dd solo >1e-4, ademas
mantiene en una sola fila la
diagonal.
        end
    end

% Vectores propios (Eigenvectors)de
la matriz de covarianza L
u=[];
for i=1:size(v,2)
    temp=sqrt(d(i)); % Raizcuadrada
de cada valor de d
    u=[u (X*v(:,i))./temp]; % X
resta de cada imagen menos media,
v eigen vectors, temp raizcuadrada
de d eigen valores.
end

% Cálculo del peso de cada rostro
de la base de datos
omega = [];
xx1=10;

for h=1:size(X,2)
    W=[];
    for i=1:size(u,2)

```

```

        t = u(:,i)';
        PesoImagen = t*X(:,h); %
Evector normalizado por resta de
fotos con media.
        W = [W; PesoImagen];
    end
    omega = [omega W]; % Filas u,
columnas
    axes(handles.axes2);
    base =
imread(strcat('C:\ProgramaTesis\Ba
seDeDatos\',int2str(h),'.jpg'));
    imshow(base);
title('Buscando...');
    tocc=ceil(toc);
    texto = sprintf('Tiempo
transcurrido en segundos:
%d',tocc);
set(handles.text12,'String',texto)
;
end
axes(handles.axes2);
fin =
imread(strcat('C:\ProgramaTesis\Fo
ndo\', 'Fondo3.jpg'));
imshow(fin); title('Finalizando
busqueda');
pause(1)
% Aplicamos filtro espacial a
imagen de entrada
R=I(:,:,1);
G=I(:,:,2);
B=I(:,:,3);
w=[0 -1 0;-1 5 -1;0 -1 0];
Rf=imfilter(R,w);
Gf=imfilter(G,w);
Bf=imfilter(B,w);
I(:,:,1)=Rf;
I(:,:,2)=Gf;
I(:,:,3)=Bf;
%Detección facial aplicando Viola-
Jones.
faceDetector =
vision.CascadeObjectDetector; bboxes
= step(faceDetector, I); IFaces =
insertObjectAnnotation(I,
'rectangle', bboxes, 'Rostro');
[hi,ni] = size(bboxes);
if(hi>1)
    opcl = questdlg('Se detectomás
de un rostro. Por favor vuelva a
intentarlo','DETECCIÓN
FALLIDA','Ok','Si'); % Crea un
cuadro de diálogo de pregunta
if strcmp(opcl,'Si')
        return;
    end
end
if hi==1
recorte=imcrop(I,bboxes(1,:)); %
Recorta la imagen.

```



```

imwrite(recorte,
['C:\ProgramaTesis\RostraNuevoUsua
rio\','1.jpg']);
work_area=imresize(recorte,[210
210]); % Cambia el tamaño de la
imagen a 210x210 pixeles.
recorte2=work_area(28:187,
45:164,:); % Recorta la imagen.

% Area de trabajo
work_area2=imresize(recorte2,[120
120]); % Cambia el tamaño de la
imagen a 120x120 pixeles.
img=rgb2gray(work_area2); %
Transforma la imagen a escala de
grises.

% Imagen de entrada
120x120 pixeles (Usuario a
identificar).

InImage=reshape(double(img)',ifil*
icol,1); % Reordena la matriz, crea
una matriz (N1*N2)x1.
me=mean(InImage); % Cálculo de la
media.

st=std(InImage); % Cálculo de la
desviación estandar.

NormImage=(InImage-me)*ustd/st+um;
Diferencia = NormImage-m;

% Reconstrucción de la imagen
p = [];
aa=size(u,2)
; for i =
1:aa

pare = dot(NormImage,u(:,i));
% Producto punto.
p = [p;
pare];
end

```

```

ReshapedImage = m + u(:,1:aa)*p; %m
es la media de la imagen, u isel
eigenvector

ReshapedImage =
reshape(ReshapedImage,icol,ifil);
ReshapedImage = ReshapedImage';

% Cálculo del peso de la imagen de
entrada.

PesoInImage = [];
for i=1:size(u,2)

t = u(:,i)'; % Transpuesto de
la matriz eigenfaces.

PesoInputImage = t*Diferencia;
PesoInImage = [PesoInImage;
PesoInputImage]; % Peso de la
imagen de entrada.

end

% Cálculo de la distancia
Euclidiana.

e=[];
for i=1:size(omega,2)
q = omega(:,i); % Matriz de
pesos de las imágenes de la base
de datos.

DifPesos = PesoInImage-q;

mag = norm(DifPesos); %
Distanci euclidiana.

e = [e mag]; % Vector de
distancias euclidianas.

```

```

end
valor = e/1000;
format long % Muestra más decimales.
[c d]=min(valor) % Indica la
distncia euclidiana mínima y su
posición en el vector.
g=num2str(d); % Converte números
a una matriz de caracteres

if c<=4.75 %Umbral para el
reconocimiento

% Usuario 1
if d<19
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','1.jpg'));
imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprint
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
('Nombre: Álvaro
Vazques'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprint
f('Edad: 24 años'));

set(handles.text20,'String',sprint
f('Domicilio: Azogues'));

set(handles.text14,'String',sprint
f('NO TIENE ANTECEDENTES
PENALES'),'ForegroundColor'
,'blue');

else
% Usuario 2
if d>=19 & d<37
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','2.jpg'));
imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprint
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
('Nombre: Christian
Valdez'),'ForegroundColor',
'black');

set(handles.text19,'String',sprint
f('Edad: 25 años'));

set(handles.text20,'String',sprint
f('Domicilio: Cañar'));

set(handles.text14,'String',sprint
f('TIENE ANTECEDENTES
PENALES'),'ForegroundColor'
,'red');

```

```

else
    set(handles.text18,'String',sprintf('INFORMACIÓN DEL USUARIO'));

% Usuario 3

    if d>=37 & d<55
        axes(handles.axes2);
        UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','3.jpg'));

        imshow(UsuarioFinal);
        title('Usuario Final');

set(handles.text18,'String',sprintf('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf('Nombre:
Jose
ph
Abarca'),'ForegroundColor','black');

set(handles.text19,'String',sprintf('Edad: 24 años'));

set(handles.text20,'String',sprintf('Domicilio: Cuenca'));

set(handles.text14,'String',sprintf('NO
TIENE
ANTECEDENTES
PENALES'),'ForegroundColor','blue');
    else
% Usuario 5

        if d>=73 & d<91
            axes(handles.axes2);
            UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','5.jpg'));

            imshow(UsuarioFinal);
            title('Usuario Final');

set(handles.text18,'String',sprintf('INFORMACIÓN DEL USUARIO'));

    else
% Usuario 4

        if d>=55 & d<73
            axes(handles.axes2);
            UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','4.jpg'));

            imshow(UsuarioFinal);
            title('Usuario Final');

```

```

set(handles.text1,'String',sprintf
('Nombre: José
Vintimilla'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprint
f('Edad: 35 años'));

set(handles.text20,'String',sprint
f('Domicilio: Cuenca'));

set(handles.text14,'String',sprint
f('TIENE ANTECEDENTES
PENALES'),'ForegroundColor'
,'red');

else
% Usuario 6

if d>=91 & d<109
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','6.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprint
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
('Nombre: Eduardo
Ávila'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprint
f('Edad: 28 años'));

set(handles.text20,'String',sprint
f('Domicilio: Azogues'));

set(handles.text14,'String',sprint
f('TIENE ANTECEDENTES
PENALES'),'ForegroundColor'
,'red');

else
% Usuario 7

if d>=109 & d<127
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','7.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprint
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
('Nombre: Raúl
Deleg'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprint
f('Edad: 27 años'));

```

```

set(handles.text20,'String',sprintf
f('Domicilio: Cañar'));

set(handles.text14,'String',sprin
tf('NO TIENE
ANTECEDENTES
PENALES'),'ForegroundColor'
,'blue');

else
% Usuario 8
if d>=127 & d<145
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','8.jpg'));
imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprin
f('Nombre: Miguel
el
Abad'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprin
f('Edad: 23 años'));

set(handles.text20,'String',sprin
f('Domicilio: Manabí'));

set(handles.text14,'String',sprin
tf('NO TIENE
ANTECEDENTES
PENALES'),'ForegroundColor'
,'blue');

else
% Usuario 9
if d>=145 & d<163
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','9.jpg'));
imshow(UsuarioFinal);
title('Usuario Final')

set(handles.text18,'String',sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprin
f('Nombre: Rafael
Illares'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprin
f('Edad: 24 años'));

set(handles.text20,'String',sprin
f('Domicilio: Cuenca'));

set(handles.text14,'String',sprin
f('NO TIENE ANTECEDENTES
PENALES'),'ForegroundColor'
,'blue');

else
% Usuario 10
if d>=163 & d<181
axes(handles.axes2);

```

```

        UsuarioFinal = Yúbi'), 'ForegroundColor'
        ,
        'black');
imread(strcat('C:\ProgramaTesis\Ro
stros\','10.jpg'));

        imshow(UsuarioFinal);
        title('Usuario Final');

set(handles.text18, 'String', sprintf
('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprintf
('Nombre: Klever
Vazquez'), 'ForegroundColor'
,
'black');

set(handles.text19, 'String', sprintf
('Edad: 28 años'));

set(handles.text20, 'String', sprintf
('Domicilio: Gualaceo'));

set(handles.text14, 'String', sprintf
('TIENE ANTECEDENTES
PENALES'), 'ForegroundColor'
,
'red');

        else
        % Usuario 12
        if d>=199 & d<217
            axes(handles.axes2);
            UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','12.jpg'));
            imshow(UsuarioFinal);
            title('Usuario Final');

set(handles.text18, 'String', sprintf
('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprintf
('Nombre: Jheferson

        else
        % Usuario 11
        if d>=181 & d<199
            axes(handles.axes2);
            UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','11.jpg'));

            imshow(UsuarioFinal);
            title('Usuario Final');

set(handles.text18, 'String', sprintf
('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprintf
('Nombre: Agustin

```

```

Gualán','ForegroundColor'      ,
'black'));

set(handles.text19,'String',sprintf('Edad: 27 años'));

set(handles.text20,'String',sprintf('Domicilio: Tambo'));

set(handles.text14,'String',sprintf('TIENE ANTECEDENTES PENALES'),'ForegroundColor'      ,
'red');

else

% Usuario
13

if      d>=217      &      d<235
axes(handles.axes2);
UsuarioFinal      =
imread(strcat('C:\ProgramaTesis\Ro
stros\','13.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprintf('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf('Nombre: Dani lo
Vanegas'),'ForegroundColor'      ,
'black');

set(handles.text19,'String',sprintf('Edad: 26 años'));

set(handles.text20,'String',sprintf('Domicilio: Deleg'));

set(handles.text14,'String',sprintf('TIENE ANTECEDENTES PENALES'),'ForegroundColor'      ,
'red');

else

% Usuario 14

if      d>=235      &      d<253
axes(handles.axes2);
UsuarioFinal      =
imread(strcat('C:\ProgramaTesis\Ro
stros\','14.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprintf('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf('Nombre: Jhanet
Lema'),'ForegroundColor'      ,
'black');

set(handles.text19,'String',sprintf('Edad: 22 años'));

set(handles.text20,'String',sprintf('Domicilio: Biblian'));

set(handles.text14,'String',sprintf('NO TIENE ANTECEDENTES PENALES'),'ForegroundColor'      ,
'red');

```

```

PENALES'), 'ForegroundColor'      ,      set(handles.text18, 'String', sprin
'blue');                               f('INFORMACIÓN DEL USUARIO'));

    else

% Usuario 15                            set(handles.text1, 'String', sprinf
    if      d>=253      &      d<271    ('Nombre:                               María
        axes(handles.axes2);          Pérez'), 'ForegroundColor'      ,
        UsuarioFinal      =          'black');

imread(strcat('C:\ProgramaTesis\Ro      set(handles.text19, 'String', sprin
stros\', '15.jpg'));                    f('Edad: 22 años'));

        imshow(UsuarioFinal);

title('Usuario Final');

set(handles.text18, 'String', sprin    set(handles.text20, 'String', sprin
f('INFORMACIÓN DEL USUARIO'));        f('Domicilio: Loja'));

set(handles.text1, 'String', sprinf    set(handles.text14, 'String', sprin
('Nombre:                               Bruno
Benavidez'), 'ForegroundColor'      ,    f('NO      TIENE      ANTECEDENTES
'black');                               PENALES'), 'ForegroundColor'      ,
                                          'blue');

set(handles.text19, 'String', sprin    else
f('Edad: 27 años'));                    % Usuario 17

set(handles.text20, 'String', sprin    if      d>=289      &      d<307
f('Domicilio: Cuenca'));                axes(handles.axes2);
                                          UsuarioFinal      =

set(handles.text14, 'String', sprin    imread(strcat('C:\ProgramaTesis\Ro
f('NO      TIENE      ANTECEDENTES      stros\', '17.jpg'));

PENALES'), 'ForegroundColor'      ,    imshow(UsuarioFinal);
'blue');                                  title('Usuario Final');

    else

% Usuario 16                            if      d>=271      &      d<289
    if      d>=271      &      d<289    axes(handles.axes2);
        axes(handles.axes2);          UsuarioFinal      =

imread(strcat('C:\ProgramaTesis\Ro      =
stros\', '16.jpg'));                    imread(strcat('C:\ProgramaTesis\Ro
                                          stros\', '16.jpg'));

        imshow(UsuarioFinal);

title('Usuario Final');

```



```

set(handles.text18,'String',sprintf
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
f('Nombre:
Carl
os
Padilla'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprintf
f('Edad: 28 años'));

set(handles.text20,'String',sprintf
f('Domicilio: Cuenca'));

set(handles.text14,'String',sprin
t f('TIENE
ANTECEDENTE
S
PENALES'),'ForegroundColor'
,'red');

else

% Usuario
18

if d>=307 & d<325
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','18.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprintf
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
f('Nombre:
Santia
go
Bermeo'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprintf
f('Edad: 26 años'));

set(handles.text20,'String',sprintf
f('Domicilio: Cuenca'));

set(handles.text14,'String',sprintf
f('TIENE
ANTECEDENTES
PENALES'),'ForegroundColor'
,'red');

else

% Usuario 19

if d>=325 & d<343
axes(handles.axes2);
UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\','19.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprintf
f('INFORMACIÓN DEL
USUARIO'),'ForegroundColor'
,'black');

set(handles.text1,'String',sprintf
f('Nombre: Jhon Lima'));

set(handles.text19,'String',sprintf
f('Edad: 26 años'));

```

```

if d>=361 & d<379
    axes(handles.axes2);
    UsuarioFinal =
        imread(strcat('C:\ProgramaTesis\Ro
stros\', '21.jpg'));
        imshow(UsuarioFinal);
        title('Usuario Final');

set(handles.text18, 'String', sprintf
('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprintf
('Nombre: Medardo
Arévalo'), 'ForegroundColor',
'black');

set(handles.text19, 'String', sprintf
('Edad: 26 años'));

set(handles.text20, 'String', sprintf
('Domicilio: Guayaquil'));

set(handles.text14, 'String', sprintf
('TIENE ANTECEDENTES
PENALES'), 'ForegroundColor',
'red');

else
    % Usuario 22

    if d>=379 & d<397

set(handles.text20, 'String', sprintf
('Domicilio: Cuenca'));

set(handles.text14, 'String', sprintf
('NO TIENE ANTECEDENTES
PENALES'), 'ForegroundColor',
'blue');

else
    % Usuario 21

set(handles.text20, 'String', sprintf
('Domicilio: Cuenca'));

set(handles.text14, 'String', sprintf
('TIENE ANTECEDENTES
PENALES'), 'ForegroundColor',
'red');

else
    % Usuario 20

    if d>=343 & d<361
        axes(handles.axes2);
        UsuarioFinal =
            imread(strcat('C:\ProgramaTesis\Ro
stros\', '20.jpg'));

            imshow(UsuarioFinal);
            title('Usuario Final');

set(handles.text18, 'String', sprintf
('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprintf
('Nombre: Paola
Paguay'), 'ForegroundColor',
'black');

set(handles.text19, 'String', sprintf
('Edad: 24 años'));

set(handles.text20, 'String', sprintf
('Domicilio: Cuenca'));

set(handles.text14, 'String', sprintf
('NO TIENE ANTECEDENTES
PENALES'), 'ForegroundColor',
'blue');

else
    % Usuario 21

```

```

        axes(handles.axes2);
        UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\', '22.jpg'));
        imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18, 'String', sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprin
f('Nombre:
Manu
el
Iglesias'), 'ForegroundColor'
,
'black');

set(handles.text19, 'String', sprin
f('Edad: 25 años'));

set(handles.text20, 'String', sprin
f('Domicilio: Sigsig'));

set(handles.text14, 'String', sprin
f('TIENE
ANTECEDENTE
S
PENALES'), 'ForegroundColor'
,
'red');
        else
        % Usuario
        23
        if d>=397 & d<415
            axes(handles.axes2);
            UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\', '23.jpg'));
            imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18, 'String', sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprin
f('Nombre:
Abraham
set(handles.text18, 'String', sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprin
f('Nombre:
Pedro
Siguenca'), 'ForegroundColor'
,
'black');

set(handles.text19, 'String', sprin
f('Edad: 25 años'));

set(handles.text20, 'String', sprin
f('Domicilio: Matchala'));

set(handles.text14, 'String', sprin
f('NO
TIENE
ANTECEDENTES
PENALES'), 'ForegroundColor'
,
'blue');
        else
        % Usuario 24
        if d>=415 & d<433
            axes(handles.axes2);
            UsuarioFinal =
imread(strcat('C:\ProgramaTesis\Ro
stros\', '24.jpg'));
            imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18, 'String', sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprin
f('Nombre:
Abraham

```

```

Renteria'),'ForegroundColor'      ,      PENALES'),'ForegroundColor'      ,
'black');                               'blue');

                                     else

set(handles.text19,'String',sprintf   % Usuario 26
f('Edad: 26 años'));

                                     if      d>=451      &      d<469
set(handles.text20,'String',sprintf   axes(handles.axes2);
f('Domicilio: Matchala'));           UsuarioFinal      =
                                     imread(strcat('C:\ProgramaTesis\Ro
set(handles.text14,'String',sprintf   stros\','26.jpg'));
f('NO      TIENE      ANTECEDENTES   imshow(UsuarioFinal);
PENALES'),'ForegroundColor'      ,   title('Usuario Final');
'blue');

                                     set(handles.text18,'String',sprintf
                                     f('INFORMACIÓN DEL USUARIO'));

                                     % Usuario 25

                                     if      d>=433      &      d<451
                                     axes(handles.axes2);
                                     UsuarioFinal      =
imread(strcat('C:\ProgramaTesis\Ro   set(handles.text1,'String',sprintf
stros\','25.jpg'));                 ('Nombre:      Isauro
                                     imshow(UsuarioFinal);           Jaramillo'),'ForegroundColor'      ,
                                     title('Usuario Final');           'black');

                                     set(handles.text19,'String',sprintf
                                     f('Edad: 25 años'));

                                     set(handles.text20,'String',sprintf
                                     f('Domicilio: Matchala'));

                                     set(handles.text14,'String',sprintf
                                     f('NO      TIENE      ANTECEDENTES
                                     f('Nombre:      Elvis
Landi'),'ForegroundColor'      ,   set(handles.text14,'String',sprintf
'black');                           f('NO      TIENE      ANTECEDENTES

                                     set(handles.text19,'String',sprintf
                                     f('Edad: 25 años'));

                                     set(handles.text20,'String',sprintf
                                     f('Domicilio: Yantzaza'));

                                     set(handles.text14,'String',sprintf
                                     f('NO      TIENE      ANTECEDENTES

```

```

PENALES'), 'ForegroundColor'      ,
'blue');

    else

% Usuario
    27

    if      d>=469      &      d<487
        axes(handles.axes2);
        UsuarioFinal      =

imread(strcat('C:\ProgramaTesis\Ro
stros\','27.jpg'));

        imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18, 'String', sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprin
f('Nombre:
        Alejand
roAbad'), 'ForegroundColor'      ,
'black');

set(handles.text19, 'String', sprin
f('Edad: 28 años'));

set(handles.text20, 'String', sprin
f('Domicilio: Cañar'));

set(handles.text14, 'String', sprin
t
        f('TIENE
        ANTECEDENT
ES
PENALES'), 'ForegroundColor'      ,
'red');

    else

% Usuario
    28

    if      d>=487      &      d<505
        axes(handles.axes2);
        UsuarioFinal      =

imread(strcat('C:\ProgramaTesis\Ro
stros\','28.jpg'));
    
```

```

        imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18, 'String', sprin
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1, 'String', sprin
f('Nombre:
        Adrián
Aguilar'), 'ForegroundColor'      ,
'black');

set(handles.text19, 'String', sprin
f('Edad: 25 años'));

set(handles.text20, 'String', sprin
f('Domicilio: Girón'));

set(handles.text14, 'String', sprin
f('NO
        TIENE
        ANTECEDENTES
PENALES'), 'ForegroundColor'      ,
'blue');

    else

% Usuario 29

    if      d>=505      &      d<523
        axes(handles.axes2);
        UsuarioFinal      =

imread(strcat('C:\ProgramaTesis\Ro
stros\','29.jpg'));

        imshow(UsuarioFinal);
title('Usuario Final');
    
```

```

set(handles.text18,'String',sprintf
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
('Nombre:          Christian
Ortiz'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprintf
f('Edad: 28 años'));

set(handles.text20,'String',sprintf
f('Domicilio: Sevilla de Oro'));

set(handles.text14,'String',sprintf
f('TIENE          ANTECEDENTES
PENALES'),'ForegroundColor'
,'red');

else
% Usuario 30

if      d>=523      &      d<541
axes(handles.axes2);
UsuarioFinal      =

imread(strcat('C:\ProgramaTesis\Ro
stros\','30.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprintf
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
('Nombre:          Klever
Tepán'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprintf
f('Edad: 28 años'));

set(handles.text20,'String',sprintf
f('Domicilio: Cuenca'));

set(handles.text14,'String',sprintf
f('TIENE          ANTECEDENTES
PENALES'),'ForegroundColor'
,'red');

else
% Usuario 31

if      d>=541      &      d<559
axes(handles.axes2);
UsuarioFinal      =

imread(strcat('C:\ProgramaTesis\Ro
stros\','31.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprintf
f('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprintf
('Nombre:          Xavier
Niola'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprintf
f('Edad: 25 años'));

```

```

set(handles.text20,'String',sprintf
f('Domicilio: Cuenca'));

set(handles.text14,'String',sprin
tf('NO TIENE ANTECEDENTES
PENALES'),'ForegroundColor'
,'blue');

else
% Usuario
32

if d>=559 & d<577
axes(handles.axes2);
UsuarioFinal =

imread(strcat('C:\ProgramaTesis\Ro
stros\','32.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprin
tf('INFORMACIÓN DEL USUARIO'));

set(handles.text1,'String',sprint
f('Nombre: Wili
am
Sanango'),'ForegroundColor'
,'black');

set(handles.text19,'String',sprint
f('Edad: 26 años'));

set(handles.text20,'String',sprint
f('Domicilio: Biblian'));

set(handles.text14,'String',sprin
tf('NO TIENE ANTECEDENTES
PENALES'),'ForegroundColor'
,'blue');

else
% Usuario Nuevo1

if d>=577 & d<595
axes(handles.axes2);
UsuarioFinal =

imread(strcat('C:\ProgramaTesis\Ba
seDeDatos\','g','.jpg'));

imshow(UsuarioFinal);
title('Usuario Final');

set(handles.text18,'String',sprint
f('INFORMACIÓN DEL USUARIO'));

load('Nuevo1.mat')

set(handles.text1,'String',sprintf
('Nombre:
%s',x1),'ForegroundColor'
,'black');

set(handles.text19,'String',sprint
f('Edad: %s',x2));

set(handles.text20,'String',sprint
f('Domicilio: %s',x3));

set(handles.text14,'String',x4,'Fo
regroundColor' ,'magenta');

else
% Usuario Nuevo2

if d>=595 & d<613
axes(handles.axes2);

```

```

        UsuarioFinal = end
        imread(strcat('C:\ProgramaTesis\Ro
stroNuevoUsuario\','1.jpg')); end
        imshow(UsuarioFinal); end
        title('Usuario Final'); end
        end
        set(handles.text18,'String',sprint
f('INFORMACIÓN DEL USUARIO')); end
        load('Nuevo2.mat') end
        end
        set(handles.text1,'String',sprintf
('Nombre:
%s',x1),'ForegroundColor' , end
        'black'); end
        end
        set(handles.text19,'String',sprint
f('Edad: %s',x2)); end
        end

        else

        axes(handles.axes2);
        UsuarioFinal =

        imread(strcat('C:\ProgramaTesis\Fon
do\','Fondo2.jpg'));

        imshow(UsuarioFinal);
        title('Usuario Final');

        set(handles.text1,'String',sprintf
('EL USUARIO NO
EXISTE'),'ForegroundColor' ,
        'red');

        set(handles.text19,'String',sprint
f('No se encuentra en la Base de
Datos'));

        end
        else
            opc2 = questdlg('No se detectó
ningun rostro. Por favor vuelva a
intentarlo','DETECCIÓN
FALLIDA','Ok','Si'); % Crea un
cuadro de diálogo de pregunta
            if strcmp(opc2,'Si')

                return;

            end

        end
    end
end

```



**Villafuerte Carrillo, Ricardo Patricio. Sistema de reconocimiento facial mediante inteligencia artificial para registro de asistencia de personal en campo.**

```

set(handles.text20,'String',sprintf      a2=toc;
f('Domicilio: %s',x3));                  tocc=ceil(toc);

set(handles.text14,'String',x4,'Fo      texto = sprintf('Tiempo de
regroundColor' , 'magenta');           operación total en segundos:
                                        %d',tocc);
                                        set(handles.text12,'String',texto)

                                        ;

                                        % --- Se ejecuta al cambiar de
                                        selección en popupmenu1.

                                        function
                                        popupmenu1_Callback(hObject,
                                        eventdata, handles)

                                        global vid
                                        global IFaces
                                        global work_area
                                        global img

                                        global ReshapedImage
                                        value =
                                        get(handles.popupmenu1,'Value');
                                        switch value

                                        case 1
                                        axes(handles.axes1);

```

```
        imshow(vid),                                get(0,'defaultUicontrolBackgroundC
title('Imagen de entrada');                        olor'))

    case 2

        axes(handles.axes1);
        imshow(IFaces),

title('Detección facial');
    case 3

        axes(handles.axes1);
        imshow(work_area);
title('Área de trabajo');
    case 4

        axes(handles.axes1);
        imshow(img);

title('Escala de grises');
    case 5

        axes(handles.axes1);
        imagesc(ReshapedImage);

colormap('gray');    title('Imagen
Reconstruida');

end

% --- Executes during object
creation, after setting all
properties.

function
popupmenu1_CreateFcn(hObject,
eventdata, handles)

% hObject    handle to popupmenu1
(see GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB

% handles empty - handles not
created until after all CreateFcns
called

% Hint: popupmenu controls usually
have a white background onWindows.

%     See ISPC and COMPUTER.

if    ispc    &&
isequal(get(hObject,'BackgroundCol
or'),
```

```
% --- Se ejecuta durante
    la creación del objeto,
después de configurar todas las
propiedades. function
    text1_CreateFcn(hObject,
eventdata, handles)

% hObject    handle to text1
(seeGCBO)

% eventdata reserved - to be
defined in a future version of
MATLAB

% handles empty - handles not
created until after all CreateFcns
called

% Hint: edit controls usually have
a white background on Windows.

%       See ISPC and COMPUTER.

if         ispc
            &&
isequal(get(hObject,'BackgroundCo
l
            or'),
get(0,'defaultUicontrolBackground
Color'))

set(hObject,'BackgroundColor','whi
te');

end

function    edit2_Callback(hObject,
eventdata, handles)

% hObject    handle to text1 (see
GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB

% handles    structure withhandles
and user data (seeGUIDATA)

% Hints:    get(hObject,'String')
returns contents of text1 as text

%

set(hObject,'BackgroundColor','whi
te');

end

% --- Se ejecuta cuando el usuario
intenta cerrar figure1.

function
figure1_CloseRequestFcn(hObject,
eventdata, handles)

% hObject    handle to figure1
(see GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB

% handles    structure withhandles
and user data (seeGUIDATA)

opc = questdlg('¿Desea salir del
programa?','SALIR','Si','No','No')
;
if strcmp(opc,'No')
    return;

str2double(get(hObject,'String'))
returns contents of text1 as a
double
```

**Villafuerte Carrillo, Ricardo Patricio.** *Sistema de reconocimiento facial mediante inteligencia artificial para registro de asistencia de personal en campo.*

```
end
delete(hObject)
;

% --- Ejecuta al presionar el botón
pushbutton11 (Dase De Datos).

function
pushbutton11_Callback(hObject,
eventdata, handles)

% hObject          handle
%   topushbutton11 (see GCBO)

% eventdata reserved - to be
defined in a future version of
MATLAB

% handles structure withhandles
and user data (seeGUIDATA)

close(Guide)
;Guide2
```

```

% --- Ejecuta al presionar el botón
pushbutton13 (Guardar Usuario).

function
pushbutton13_Callback(hObject,
eventdata, handles)

% hObject handle to
pushbutton13 (see GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB

% handles structure with handles
and user data (see GUIDATA)

% guardar en formatos
global I
p=1;

ab
='C:\ProgramaTesis\BaseDeDatos'; %
Dirección de la carpeta Base de
Datos

B =dir( fullfile(ab, '*.jpg'));

%Muestra lista de contenidos de la
carpeta.

t=size(B,1);

%Número de imágenes en la base de
datos.

% Se aplica un filtro espacial a
cada imagen.

R=I(:,:,1);

G=I(:,:,2);

B=I(:,:,3);

w=[0 -1 0;-1 5 -1;0 -1 0];
Rf=imfilter(R,w);
Gf=imfilter(G,w);
Bf=imfilter(B,w);
I(:,:,1)=Rf;

I(:,:,2)=Gf;

I(:,:,3)=Bf;

% Detección facial utilizando
Viola-Jones

faceDetector =
vision.CascadeObjectDetector; %
Encuentra rostros en imágenes.
bboxes = step(faceDetector, I); %
Detecta rostros en imágenes.

IFaces = insertObjectAnnotation(I,
'rectangle', bboxes, 'Rostro'); %
Comenta rostros detectados. [hi,ni]
= size(bboxes);

if(hi>1)

    opcl = questdlg('Se detectó más
de un rostro. Por favor vuelva a
intentarlo','DETECCIÓN
FALLIDA','Ok','Si'); % Crea un
cuadro de diálogo de pregunta
    if strcmp(opcl,'Si')

        return;
    end

end

if hi==1
load('pul.mat','p1') p1
= p1 + 1

save('pul.mat','p1')

```

```

texto = sprintf('Foto: %d',p1);
set(handles.text21,'String',texto)

;

% Area de trabajo
recorte=imcrop(I,bboxes(1,:)); %
Recorta la imagen.
work_area=imresize(recorte,[210
210]); % Cambia el tamaño de la
imagen a 210x210 pixeles.
recorte2=work_area(28:187,
45:164,:); % Recorta la imagen.
img=rgb2gray(recorte2); %
Transforma la imagen a escala de
grises.
work_area2=imresize(img,[120
120]); % Cambia el tamaño de la
imagen a 120x120 pixeles.

t = t+1
b = num2str(t);
imwrite(work_area2,
['C:\ProgramaTesis\BaseDeDatos\',
b
, '.jpg']);

msj = questdlg('Se ha agregado a
la Base de Datos correctamente',
'Guardado','Ok','Ok');

if p1>17

p1 = 0

msj = questdlg('Límite de fotos
permitido por usuario', 'Nuevo
Usuario Creado','Ok','Ok');

save('pul.mat','p1')

end

else

opc2 = questdlg('No se detectó
ningun rostro. Por favor vuelva a
intentarlo','DETECCIÓN
FALLIDA','Ok','Si'); % Crea
un

```

```

cuadro de diálogo de pregunta
if strcmp(opc2,'Si')

return;

end

end

% --- Ejecuta al presionar el botón
pushbutton14 (Agregar Datos).

function
pushbutton14_Callback(hObject,
eventdata, handles)

% hObject handle to
pushbutton14 (see GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB

% handles structure with handles
and user data (see GUIDATA)

close(Guide);
Guide3

% --- Executes on button press in
pushbutton15.

function
pushbutton15_Callback(hObject,
eventdata, handles)

```

```

% hObject      handle to      [varargout{1:nargout}]      =
pushbutton15 (see GCBO)      gui_mainfcn(gui_State, varargin{:});

% eventdata reserved - to be defined
in a future version of MATLAB      else
% handles      structure withhandles      gui_mainfcn(gui_State,
and user data (seeGUIDATA)      varargin{:});
end
close(Guide);
Guide4      % End initialization code - DO NOT
      EDIT

```

### **Segunda Interfaz**

```

function      varargout      =      % --- Executes just before Guide2is
Guide2(varargin)      made visible.

function
Guide2_OpeningFcn(hObject,
eventdata, handles, varargin)
% This function has no outputargs,
see OutputFcn.
% hObject      handle to figure
% eventdata reserved - to be defined
in a future version of MATLAB
% handles      structure withhandles
and user data (seeGUIDATA)
% varargin      command line
arguments to Guide2 (see VARARGIN)
axes(handles.axes4);

% Begin initialization code - DO
NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',
mfilename, ...

'gui_Singleton',      gui_Singleton,
...

'gui_OpeningFcn',
@Guide2_OpeningFcn, ...

'gui_OutputFcn',
@Guide2_OutputFcn, ...

'gui_LayoutFcn', [] , ...

      'gui_Callback',
[]);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback      =
str2func(varargin{1});
end

if nargin

```

```

Fondo3 = % hObject handle to pushbutton2
                                                (see GCBO)
imread(strcat('C:\ProgramaTesis\Fondo\', 'Fondo1.jpg')); % eventdata reserved - to be defined
imshow(Fondo3) % eventdata reserved - to be defined
                                                in a future version of MATLAB
                                                % handles structure withhandles
                                                and user data (seeGUIDATA)
% Choose default command line output for Guide2
close(Guide2);
handles.output = hObject;
% Update handles
structureguidata(hObject, handles);
                                                Guide
                                                % --- Se ejecuta cuando el usuario
                                                intenta cerrar figura1.
function
figure1_CloseRequestFcn(hObject, eventdata, handles)
% hObject handle to figure1
(see GCBO)
% eventdata reserved - to be defined
in a future version of MATLAB
% handles structure withhandles
and user data (seeGUIDATA)
close(Guide2);
Guide
function
figure1_OutputFcn(hObject, eventdata, handles) =
% hObject handle to figure
% eventdata reserved - to be defined
in a future version of
MATLAB
% handles structure withhandles
and user data (seeGUIDATA)
% varargout cell array for
returning output args (see
VARARGOUT);
% Get default command line output
from handles structure
varargout{1} = handles.output;
% --- Ejecuta al presionar elbotón
pushbutton2 (BotónTerminar).
function
pushbutton2_Callback(hObject, eventdata, handles)

```



```

% --- Executes during object
creation, after setting all
properties.

function edit1_CreateFcn(hObject,
 eventdata, handles)

% hObject handle to edit1 (see GCBO)
% eventdata reserved - to be defined
in a future version of MATLAB
% handles empty - handles not
created until after all CreateFcns
called

% Hint: edit controls usually have a
white background on Windows.

% See ISPC and COMPUTER.
if ispc &&
isequal(get(hObject,'BackgroundCol
or'),
get(0,'defaultUicontrolBackgroundC
olor'))

set(hObject,'BackgroundColor','whi
te');

end

% --- Executes during object
creation, after setting all
properties.

function text1_CreateFcn(hObject,
 eventdata, handles)

% hObject handle to text1 (see GCBO)
% eventdata reserved - to be defined
in a future version of MATLAB
% handles empty - handles not
created until after all CreateFcns
called

% --- Ejecuta en cambio de selección
en popupmenu1 (Botón
Características).

function
popupmenu1_Callback(hObject,
 eventdata, handles)

% hObject handle to popupmenu1
(see GCBO)
% eventdata reserved - to be defined
in a future version of MATLAB
% handles structure with handles
and user data (see GUIDATA)

% Hints: contents =
cellstr(get(hObject,'String'))
returns popupmenu1contents as
cell array

%
contents{get(hObject,'Value')}
returns selected item from
popupmenu1

clc

a ='C:\ProgramaTesis\BaseDeDatos';A
=dir( fullfile(a, '*.jpg') );

```

```

M=size(A,1); % Obtenemos el número de filas
value = (N1) y columnas (N2).

get(handles.popupmenu1,'Value');
switch value
    case 1
        tic;
        for i=1:M
            img =
imread(strcat('C:\ProgramaTesis\Ba
seDeDatos\',int2str(i),'.jpg'));
            if
i<=25
axes(handles.uipanel1);
subplot(ceil(sqrt(25)),ceil(sqrt(2
5)),i);
                imshow(img)
                ;end
            en
            dtoc;
            tocc=ceil(toc);
            texto = sprintf('Imágenes
faciales: %d',M);
set(handles.text4,'String',texto);
            texto = sprintf('Tiempo de
carga en segundos: %d',tocc);
set(handles.text5,'String',texto);

        case 2
            tic
            uestd=80;
            um=100;S=[];
            for i=1:M
                img =
imread(strcat('C:\ProgramaTesis\Ba
seDeDatos\',int2str(i),'.jpg'));
                [ifil icol]=size(img);

```

```

temp=reshape(img',ifil*icol,1); %
Reordena la matriz, crea una matriz
(N1*N2)x1.
        S=[S temp];
        end
% Normalizar imágenes de la base
de datos para reducir el error
debido a las condiciones de
iluminación.
        for i=1:size(S,2) %Valordel
número de columnas de la matriz S
            temp=double(S(:,i));
%almacena todas las variables
numéricas como valores de punto
flotante de doble precisión que
son 8 bytes (64 bits)
            m=mean(temp); %saca lamedia
de toda cada (1) columna un solo
valor
            st=std(temp);
            S(:,i)=(temp-
m)*ustd/st+um; % (foto-media de
foto)*ustd/desviacion estandar de
foto +um

```

```

end
img=img';
end

% muestra imágenes normalizadas

for i=1:M
    %Cálculo de la media de las
    imagenes;
    m=mean(S,2); % obtienela
    media de cada fila en lugar de cada
    columna
    mn=uint8(m); % se
    convierte en un entero de 8 bits sin
    signo. Los valores van de 0 a 255.

    axes(handles.uipanel1);

    subplot(ceil(sqrt(25)),ceil(sqrt(2
5)),i);
    imshow(img);
end
end
toc;
tocc=ceil(toc);
texto = sprintf('Imágenes
Normalizadas: %d',M);

set(handles.text4,'String',texto);
texto = sprintf('Tiempo de
operación en segundos: %d',tocc);
set(handles.text5,'String',texto);

case 3
    tic
    S=[];
    ustd=80; um=100;
    for i=1:M
        img =
imread(strcat('C:\ProgramaTesis\Ba
seDeDatos\',int2str(i),'.jpg'));
        [ifil
icol]=size(img);

```

**Villafuerte Carrillo, Ricardo Patricio. Sistema de reconocimiento facial mediante inteligencia artificial para registro de asistencia de personal en campo.**

```

img=reshape(mn,icol,ifil);      operación en segundos: %d',tocc);
% toma el vector N1*N2x1 y crea  set(handles.text5,'String',texto);
una matriz N2xN1

    img=img'; % crea una
matriz N1xN2 mediante la
transposición de la imagen.

    axes(handles.uipanel1);
    imshow(img);

    toc;
    tocc=ceil(toc);

    texto = sprintf('Grafica
la media de la base de datos');

set(handles.text4,'String',texto);
    texto = sprintf('Tiempo de

end

    for i=1:size(S,2) %Valordel
número de columnas de la matriz S

        temp=double(S(:,i));

%almacena todas las variables
numéricas como valores de punto
flotante de doble precisión que
son 8 bytes (64 bits)

        m=mean(temp); %saca la
media de toda cada (1) columna un
solo valor

        st=std(temp);
        S(:,i)=(temp-

m)*ustd/st+um; % (foto-media de
foto)*ustd/desviacion estandar de
foto +um

end

% muestra imágenes normalizadas
    for i=1:M

str=strcat(int2str(i),'.jpg');
img=reshape(S(:,i),icol,ifil);

case 4
tic
ustd=80; um=100;
S=[];
for i=1:M

img =
imread(strcat('C:\ProgramaTesis\Ba
seDeDatos\',int2str(i),'.jpg'));

[ifil icol]=size(img);

%Obtenemos el número de filas (N1)
y columnas (N2)

temp=reshape(img',ifil*icol,1);

%Reordena la matriz, crea unamatriz
(N1*N2)x1

end

for i=1:size(S,2) %Valordel
número de columnas de la matriz S

temp=double(S(:,i));

%almacena todas las variables
numéricas como valores de punto
flotante de doble precisión que
son 8 bytes (64 bits)

m=mean(temp); %saca la
media de toda cada (1) columna un
solo valor

st=std(temp);
S(:,i)=(temp-

m)*ustd/st+um; % (foto-media de
foto)*ustd/desviacion estandar de
foto +um

end

% muestra imágenes normalizadas
for i=1:M

```

```

str=strcat(int2str(i),'.jpg');
img=reshape(S(:,i),icol,ifil);
    img=img';
end

%Cálculo de la media de las
imagenes;
    m=mean(S,2); % obtienela
media de cada fila en lugar de cada
columna
    mn=uint8(m); % se
convierte en un entero de 8 bits sin
signo. Los valores van de 0 a 255.
    img=reshape(mn,icol,ifil);
    img=img';

% Cambio de imagen por
manipulación
    X=[]; % A matrix
    for i=1:M
        temp=double(S(:,i))-m;
        X=[X temp]; %Almacena
todos los temp, todas las restas
en una matriz
    end

% Cambio de imagen por
manipulación.
    A=X';
    L=A*A';%MxM, si fuera
alreves seria Ncuadrado xNcuadrado
    [vv dd]=eig(L); %vectores
evalores , igual dimencion que
covarianza L MxM, dd es decir
evalores son 0 excepto diagonal

% Ordenar y eliminar aquellos cuyo
valor propio es cero
    v=[];
    d=[];
    for i=1:size(vv,2)
        if(dd(i,i)>1e-4) %solo
si la diagonal de los dd evaloreses
>1e-4
            v=[v
vv(:,i)];%guarda en v columnas devv
en donde las columas de dd sean
>1e-4
            d=[d
dd(i,i)];%guarda en d columnas dedd
solo >1e-4, ademas mantiene enuna
sola fila la diagonal
        end
    end

% Vectores propios (Eigenvectors)de
la matriz de covarianza.
    u=[];
    for i=1:size(v,2)
        temp=sqrt(d(i));%raiz
cuadrada de cada valor de d.
        u=[u
(X*v(:,i))./temp]; %X es la restade
cada imagen menos media.

```

```

                                % v
es eigen vectors.
                                % --- Executes during object
                                % creation, after setting all
                                % properties.
                                %
                                % function
                                % popupmenu1_CreateFcn(hObject,
                                % eventdata, handles)
                                % hObject handle to popupmenu1
                                % (see GCBO)
                                % eventdata reserved - to be defined
                                % in a future version of MATLAB
                                % handles empty - handles not
                                % created until after all CreateFcns
                                % called
                                % Hint: popupmenu controls usually
                                % have a white background onWindows.
                                % See ISPC and COMPUTER.
                                %
                                % if ispc &&
                                % isequal(get(hObject,'BackgroundCol
                                % or'),
                                % get(0,'defaultUicontrolBackgroundC
                                % olor'))
                                %
                                % set(hObject,'BackgroundColor','whi
                                % te');
                                %
                                % end
                                %
                                % end
                                % toc
                                % ;
                                % tocc=ceil(toc);
                                %
                                % texto =
                                % sprintf('Eigenfaces obtenidas:
                                %d',wil);
                                %
                                % set(handles.text4,'String',texto);
                                % texto = sprintf('Tiempo de
                                % operación en segundos; %d',tocc);
                                %
                                % set(handles.text5,'String',texto);
                                %
                                % end
```

```
% --- Executes during object
creation, after setting all
properties.
```

```
function text6_CreateFcn(hObject,
 eventdata, handles)
```

```
% hObject handle to text6 (see GCBO)
```

```
% eventdata reserved - to be defined
in a future version of MATLAB
```

```
% handles empty - handles not
created until after all CreateFcns
called
```

### **Tercera Interfaz**

```
function varargout =
Guide3(varargin)
```

```
% Begin initialization code - DO NOT
EDIT
```

```
gui_Singleton = 1;
```

```
gui_State = struct('gui_Name',
 mfilename, ...
```

```
'gui_Singleton', gui_Singleton,
```

```
...
```

```
'gui_OpeningFcn',
 @Guide3_OpeningFcn, ...
```

```
'gui_OutputFcn',
 @Guide3_OutputFcn, ...
```

```
'gui_LayoutFcn', [] , ...
```

```
 'gui_Callback',
```

```
[]);
```

```
if nargin && ischar(varargin{1})
    gui_State.gui_Callback =
```

```
str2func(varargin{1});
```

```
end
```

```
if nargin
```

```
    [varargout{1:nargout}] =
```

```
gui_mainfcn(gui_State, varargin{:});
```

```
else
```

```
    gui_mainfcn(gui_State,
```

```
    varargin{:});
```

```
end
```

```
% End initialization code - DO NOT
EDIT
```

```
% --- Executes just before Guide3is
made visible.
```

```
function
Guide3_OpeningFcn(hObject,
 eventdata, handles, varargin)
```

```
% This function has no outputargs,
see OutputFcn.
```

```
% hObject handle to figure
```

```
% eventdata reserved - to be defined
in a future version of MATLAB
```

```
% handles structure withhandles
and user data (seeGUIDATA)
```

```

% varargin      command      % handles      structure withhandles
      linearguments to Guide3 (see      and user data      (seeGUIDATA)
VARARGIN)

      axes(handles.axes2);      % Get default command line output
      Fondo3      =      from      handles      structure
imread(strcat('C:\ProgramaTesis\Fondo\','Fondo1.jpg'));      varargout{1} = handles.output;

      imshow(Fondo3)      % --- Ejecuta al presionar elbotón1
      (Guardar Datos).

      handles.output      =      hObject;      function
      guidata(hObject, handles);      pushbutton1_Callback(hObject,
eventdata, handles)

function      varargout      =      % hObject      handle to pushbutton1
click_veces_OutputFcn(hObject,      (see GCBO)
eventdata, handles)      % eventdata reserved - to be defined
varargout{1} = handles.output;      in a future version of MATLAB
% Choose default command line      % handles      structure withhandles
output for Guide3      and user data      (seeGUIDATA)
handles.output = hObject;      load('count.mat','y1')
count = y1

%      Update      handles      y1      =      count      +      1
structureguidata(hObject,      save('count.mat','y1')
handles);      if y1==1

% UIWAIT makes Guide3 wait for      x1=get(handles.edit1,'String');
user response (see UIRESUME)      x2=get(handles.edit2,'String');

% uiwait(handles.figure1);

% --- Outputs from this function
are returned to the command line.
function      varargout      =
Guide3_OutputFcn(hObject,
eventdata, handles)

%      varargout      cell      array      for
returning      output      args      (see
VARARGOUT);

% hObject      handle to figure
% eventdata reserved - to be
defined in a future version of
MATLAB

```



```
x3=get(handles.edit3,'String');
x4=get(handles.edit4,'String');

save('Nuevo1.mat','x1','x2','x3','
x4')

    msj = questdlg('Los Datos se han
guardado                correctamente',
'Guardado','Ok','Ok');

else

    if                                y1==2
x1=get(handles.edit1,'String');
x2=get(handles.edit2,'String');
x3=get(handles.edit3,'String');
x4=get(handles.edit4,'String');

save('Nuevo2.mat','x1','x2','x3','
x4')

    msj = questdlg('Los Datos se han
guardado                correctamente',
'Guardado','Ok','Ok');

else

    if                                y1==3
x1=get(handles.edit1,'String');
x2=get(handles.edit2,'String');
x3=get(handles.edit3,'String');
x4=get(handles.edit4,'String');

save('Nuevo3.mat','x1','x2','x3','
x4')

    msj = questdlg('Los Datos se han
guardado                correctamente',
'Guardado','Ok','Ok');

end
end
end

function    edit1_Callback(hObject,
eventdata, handles)
```

```
% hObject    handle to edit1 (see
GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB

% handles    structure withhandles
and user data (seeGUIDATA)

% Hints:    get(hObject,'String')
returns contents of edit1 as text

%
str2double(get(hObject,'String'))
returns contents of edit1 as a
double
```

```
% --- Executes during object
creation, after setting all
properties.

function edit1_CreateFcn(hObject,
eventdata, handles)

% hObject handle to edit1 (see
GCBO)

% eventdata reserved - to be
defined in a future version of
MATLAB

% handles empty - handles not
created until after all CreateFcns
called

% Hint: edit controls usually have
a white background on Windows.

% See ISPC and COMPUTER.
if ispc
    %%
isequal(get(hObject,'BackgroundCo
lor','white'),
get(0,'defaultUicontrolBackground
Color'))

set(hObject,'BackgroundColor','whi
te');

end

% --- Ejecuta al presionar elbotón
pushbutton2 (BotónTerminar).

function
pushbutton2_Callback(hObject,
eventdata, handles)

% hObject handle to
pushbutton2(see GCBO)

% eventdata reserved - to be
defined in a future version of
MATLAB

% handles structure withhandles
and user data (seeGUIDATA)

close(Guide3)
;Guide

% --- Executes when user attemptsto
close figure1.

function
figure1_CloseRequestFcn(hObject,
eventdata, handles)

% hObject handle to figure1
(see GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB

% handles structure withhandles
and user data (seeGUIDATA)

close(Guide3);

Guide

function edit2_Callback(hObject,
eventdata, handles)

% hObject handle to edit2 (see
GCBO)

% eventdata reserved - to be defined
in a future version of MATLAB
```

```

% handles structure withhandles and user data (seeGUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
%
str2double(get(hObject,'String')) returns contents of edit2 as adouble

% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject,eventdata, handles)
% hObject handle to edit2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcnscalled

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject,eventdata, handles)
% hObject handle to edit3 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure withhandles and user data (seeGUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%
str2double(get(hObject,'String')) returns contents of edit3 as adouble

% --- Executes during object creation, after setting all properties.
function edit3_CreateFcn(hObject,eventdata, handles)
% hObject handle to edit3 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcnscalled

```

```

% Hint: edit controls usually have a white background on Windows.

%       See ISPC and COMPUTER.

if      ispc      && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit4_Callback(hObject,eventdata, handles)

% hObject      handle to edit4 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints:  get(hObject,'String') returns contents of edit4 as text
%
str2double(get(hObject,'String')) returns contents of edit4 as adouble

% --- Executes during object creation, after setting all properties.
function edit4_CreateFcn(hObject,eventdata, handles)

% hObject handle to edit4 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcnscalled

% Hint: edit controls usually have a white background on Windows.

%       See ISPC and COMPUTER.

if      ispc      && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

```

## Índice de acrónimos

- Adaboost: Adaptive Boosting
- API: Application Programming Interface
- ASM: Máquina de estados algorítmica
- CNN: Red Neuronal Convolutacional
- DAS: Sistemas de Adquisición de Datos
- EBGM: Coincidencia de Gráfico de Grupo Elástico
- HD: High Definition
- HMM: Modelos Ocultos de Markov
- IA: Inteligencia artificial
- ICA: Análisis de Componentes Independientes
- IoT: Internet de las Cosas
- LDA: análisis discriminante lineal
- MLP: Perceptrones Multicapa Difusos
- PCA: Análisis de Componentes Principales
- RRHH: Recursos Humanos
- SRF: sistema de reconocimiento facial
- SVM: Máquinas de vectores soporte
- TAM: modelo de aceptación de tecnologías
- TRI: índice de predisposición tecnológica