

**UNIVERSIDAD VALLE DEL MOMBOY  
VICERECTORADO ACADÉMICO  
FACULTAD DE INGENIERÍA  
INGENIERÍA DE COMPUTACIÓN**



**SISTEMA DE RECONOCIMIENTO DE IMÁGENES PARA EMITIR ALERTAS DE  
SEGURIDAD**

Presentado por:

Br. Jesus David Jerez Moreno

TRUJILLO, 2022

**UNIVERSIDAD VALLE DEL MOMBOY  
VICERECTORADO ACADÉMICO  
FACULTAD DE INGENIERÍA  
INGENIERÍA DE COMPUTACIÓN**



**SISTEMA DE RECONOCIMIENTO DE IMÁGENES PARA EMITIR ALERTAS DE  
SEGURIDAD**

Trabajo presentado como requisito para optar al título de Ingeniero de Computación

Presentado por:

Br. Jesus David Jerez Moreno

TRUJILLO, 2022



REPÚBLICA BOLIVARIANA DE VENEZUELA  
UNIVERSIDAD VALLE DEL MOMBOY  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA INDUSTRIAL  
CARVAJAL ESTADO TRUJILLO

ACEPTACIÓN DEL TUTOR

Carvajal, 15 de septiembre de 2021

Ciudadano: Ing. Wilmer Mendez.  
Director del CIDIFI  
Presente.-

Por medio de la presente, hago de su conocimiento, que ante la solicitud realizada por el ciudadano: **JESUS DAVID JEREZ MORENO**, portador de la **C.I.V.- 17.598.687**, acepto el compromiso de Tutorar el desarrollo de su trabajo de investigación titulado: **SISTEMA DE RECONOCIMIENTO DE IMÁGENES PARA EMITIR ALERTAS DE SEGURIDAD** para optar al título universitario en **INGENIERIA DE COMPUTACIÓN**; hasta su presentación y evaluación.

Atentamente,

---

Prof. Roberto Di Michele  
C.I.Nº 19.794.455



Av. Independencia con calle La Paz, Sede Mirabel, Urbanización Mirabel, Plata I,  
Diagonal al Parque SAPNNAET. Municipio Valera Estado Trujillo.

**VICERRECTORADO  
FACULTAD DE INGENIERÍA**

**VEREDICTO**

Nosotros, Prof. Roberto Di Michele, Prof. Edgardo Paolini y Prof. Luis Paredes, designados como miembros del Jurado Examinador del Trabajo Especial de Grado titulado: **“SISTEMA DE RECONOCIMIENTO DE IMÁGENES PARA EMITIR ALERTAS DE SEGURIDAD”**, que presenta el Bachiller **JESÚS DAVID JEREZ MORENO**, portador de la Cédula de Identidad N° **17.598.687**, nos hemos reunido para revisar dicho Trabajo y después de la presentación, defensa e interrogatorio correspondiente lo hemos calificado con: **VEINTE (20)** puntos, de acuerdo con las normas vigentes dictadas por el Consejo Universitario de la Universidad Valle del Momboy, referente a la evaluación de los Trabajos Especiales de Grado para optar al título de Ingeniero de Computación.

En fe de lo cual firmamos, en Valera a los doce (12) días del mes de julio de dos mil veintidós (2022).

Prof. Edgardo Paolini  
C.I. 13.897.564  
JURADO

Prof. Roberto Di Michele  
C.I. 19.794.455  
TUTOR

Prof. Luis Paredes  
C.I. 12.500.339  
PRESIDENTE DEL JURADO

Profa. Marilyn Briceño  
C.I.- N° 13.205.436  
DECANA



Profa. Ana Linares  
C.I.- N° 9.013.217  
VICERRECTORA

## DEDICATORIA

*Este logro se lo dedico a mis padres.*

*En especial a mi padre, a quien le puedo atribuir mi entrada al mundo de la tecnología, quien soportó que mi curiosidad resultara en muchos viajes al taller de reparaciones, y el que con mucho esfuerzo, intentó complacerme con cada libro que le pedía.*

*Gracias a ambos...*

*Jesus David Jerez Moreno*

## AGRADECIMIENTO

*A mi compañera de vida y mejor amiga Karina...*

*E igualmente a mi amiga Sandra...*

*Sin la ayuda de ambas, llegar acá no habría sido posible...*

*Jesus David Jerez Moreno*

# ÍNDICE DE CONTENIDO

ACEPTACIÓN .....	2
VEREDICTO .....	3
DEDICATORIA .....	4
AGRADECIMIENTO.....	5
ÍNDICE DE CONTENIDO.....	6
ÍNDICE DE TABLAS.....	9
ÍNDICE DE FIGURAS .....	10
RESUMEN .....	12
INTRODUCCIÓN .....	13
CAPÍTULO I: EL PROBLEMA .....	17
Planteamiento del Problema.....	17
Objetivos de la Investigación .....	24
<i>Objetivo General</i> .....	24
Objetivos Específicos.....	24
Justificación .....	24
Delimitación de la investigación.....	26
CAPÍTULO II: MARCO TEÓRICO.....	27
Antecedentes de la Investigación .....	27
Bases teóricas .....	32
Historia de la Inteligencia Artificial (IA).....	33
Inteligencia artificial (IA).....	38
Aplicación de la Inteligencia Artificial.....	40
Robótica.....	45
Redes neuronales .....	47
Redes Bayesianas.....	50
Machine Learning.....	54

Deep Learning .....	56
Procesamiento del Lenguaje Natural (PLN).....	58
Sistemas Expertos .....	60
Datasets.....	62
Visión Computacional .....	65
Clasificación.....	69
Detección y Reconocimiento.....	72
Segmentación.....	75
Aplicaciones de la Visión Computacional.....	80
Sistemas de Seguridad y Vigilancia con IA, Basado en el Modelo de Visión Computacional .....	84
Python.....	90
OpenCV .....	94
Glosario de términos.....	96
Operacionalización y Conceptualización de las Variables .....	98
CAPÍTULO III: MARCO METODOLÓGICO .....	99
Tipo de Investigación.....	99
Diseño de la Investigación.....	100
Técnicas de Recolección de Datos.....	102
Análisis y Presentación de Resultados.....	103
Metodología Holística: Investigación Proyectiva.....	104
CAPÍTULO IV: APLICACIÓN DE LA METODOLOGÍA.....	109
Etapa I: Análisis .....	109
Situación Actual .....	110
Factibilidad.....	111
Requerimientos del Sistema .....	112
Hardware .....	113
<i>Microcomputador</i> .....	113
<i>Cámara Digital</i> .....	115
<i>Sensor de Movimiento</i> .....	118
Software.....	120

<i>Lenguajes de Programación</i> .....	120
<i>Herramientas y Bibliotecas de Programación</i> .....	121
<i>Base de Datos</i> .....	123
Etapa II: Diseño .....	124
Diseño y construcción del hardware .....	124
<i>Paso 1. Configuración y habilitación del módulo de placa de cámara Raspberry Pi</i> .....	125
<i>Paso 2. Configuración de sensores</i> .....	125
<i>Paso 3. Módulos de iluminación</i> .....	128
Diseño y construcción del software.....	129
<i>Paso 1. Instalación del sistema operativo para la placa Raspberry PI</i> .....	129
<i>Paso 2. Habilidad de los componentes de hardware a través del software</i> ..	130
<i>Paso 3. Instalación de picamera y prueba de captura de imagen</i> .....	131
<i>Paso 4. Captura y reconocimiento de imágenes</i> .....	132
<i>Paso 5. Envío de alertas</i> .....	144
<i>Paso 6. Diseño de interfaz gráfica</i> .....	148
Etapa III: Prueba.....	149
Detección y Reconocimiento.....	149
Envío de Alertas .....	152
CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES .....	154
Conclusiones .....	154
Recomendaciones .....	159
REFERENCIAS.....	160
Referencias Digitales.....	162

## ÍNDICE DE TABLAS

1. Principales ámbitos de aplicación de los sistemas de inteligencia artificial .....	42
2. Tabla de variables .....	98
3. Características de microcomputador Raspberry Pi 3 B+ .....	114
4. Características de la Cámara Raspberry Ov5647.....	116
5. Módulos de luz infrarroja Para Cámara Ov5647 .....	118
6. Características del Módulo sensor de movimiento infrarrojo Pir Hc Sr501 .....	119

## ÍNDICE DE FIGURAS

1. Una red neuronal artificial con múltiples capas de neuronas.....	48
2. Datasets (rectángulo exterior) es una colección de puntos de datos (círculos). .....	62
3. Caso práctico de visión computacional general.....	65
4. Caso práctico de clasificación y etiquetado. ....	70
5. Imagen de entrada.....	71
6. Detección y localización de objetos. ....	73
7. Ejemplo de imagen con las regiones significativas.....	76
8. Segmentación.....	77
9. Conjunto de datos de Pascal VOC. ....	78
10. Visión por computadora en la industria automotriz. ....	81
11. Paradoja de los falsos positivos en una prueba diagnóstica con el 99% de precisión. ....	87
12. Cómo funcionan los conectores.....	93
13. Esquema para seleccionar los diseños de investigación.....	101
14. Ciclo Holístico de la Investigación. Investigación proyectiva. ....	105
15. Kit Raspberry Pi 3 B+. ....	114
16. Cámara Raspberry Ov5647.....	116
17. Módulos de luz infrarroja para cámara Ov5647.....	117
18. Módulo sensor de movimiento infrarrojo PIR Hc Sr501.....	119
19. Kit Raspberry Pi 3B+. ....	125
20. Diagrama de la GPIO.....	126
21. Cableado de un sensor PIR.....	127
22. Control de sensibilidad (izquierda) y tiempo de espera (derecha) de un PIR. ....	128
23. Módulos de iluminación. ....	129
24. Prototipo funcional ensamblado.....	129
25. Imagen capturada usando picamera.....	132

26. Estructura general de la base de datos. Modelo de ER.....	136
27. Arquitectura de MTCNN.....	138
28. Arquitectura de MobileFaceNet. ....	142
29. Vista principal de la interfaz web.....	149
30. Detección del rostro. ....	150
31. Asignación de identidad.....	151
32. Reconocimiento del rostro. ....	151
33. Nuevo reconocimiento del rostro en ambiente nocturno.....	152
34. Mensaje de alerta recibido en el correo electrónico.....	153
35. Detalle del acceso al reporte enviado por correo.....	153

**UNIVERSIDAD VALLE DEL MOMBOY  
VICERECTORADO ACADÉMICO  
FACULTAD DE INGENIERÍA  
INGENIERÍA DE COMPUTACIÓN**

**SISTEMA DE RECONOCIMIENTO DE IMÁGENES PARA EMITIR ALERTAS DE  
SEGURIDAD**

Autor: Br. Jesus David Jerez Moreno  
Tutor: Ing. Roberto Di Michele  
Año: 2022

**RESUMEN**

La presente investigación tiene como objetivo desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad, el cual tiene como características ser una alternativa de fácil uso y bajo costo, que responda de forma efectiva a las carencias de muchos sistemas de seguridad actuales que no poseen mecanismo en tiempo real sobre la detección, reconocimiento y envío de alerta de una intrusión dentro de un área controlada. En este marco, la investigación se realizó bajo el paradigma de la investigación holística, con un tipo de investigación proyectiva y diseño metodológico de fuente mixta, que integra el diseño documental y de laboratorio. Se utilizó como técnica de recolección de datos, la recopilación y revisión de fuentes de información a través de los distintos medios disponibles (físicos, digitales), categorizando la información basada en el objeto de estudio, mediante un enfoque descriptivo y analítico, que permitió interpretar cualitativamente los resultados, concluyendo con una investigación factible a nivel técnico, bibliográfico y económico, logrando obtener un sistema accesible en cuanto a costos y entendimiento, de fácil instalación y configuración, con una interfaz web simple, pero concisa para la exploración de las capturas de imágenes y ver el resultado de los análisis, finalmente en materia de visión computacional el sistema cumple el objetivo de hacer detecciones y reconocimientos en diversos escenarios gracias a la utilización de redes neuronales convolucionales.

**Palabras clave:** Inteligencia artificial, visión computacional, redes neuronales, data sets, clasificación, segmentación, reconocimiento, Python, OpenCV.

## INTRODUCCIÓN

Desde 1940 con la creación del primer computador a la actualidad, el desarrollo tecnológico ha revolucionado todo lo que conocemos; orientado inicialmente a la automatización de procesos, permitió agilizar y perfeccionar operaciones hasta ese entonces realizado solo por el hombre, por lo que hoy en día resulta difícil pensar en alguna labor donde no intervenga la aplicación de alguna tecnología.

Dicha evolución no solamente ha cambiado procesos e infraestructura, sino nuestra forma de pensar rompiendo paradigmas donde se creía imposible traspasar la barrera humano-máquina a través de la aplicación de razonamiento llevada a cabo por máquinas conocidas como inteligencia artificial (IA), por lo que resulta difícil pensar que existen tecnologías capaces de imitar cualidades imprescindibles del ser humano como percibir, razonar, aprender y resolver problemas, considerando todo a su alrededor, por medio de una alimentación constante de datos y análisis de ejemplos para identificar patrones, generando soluciones a partir de estos.

Hoy en día la aplicación de la IA se extiende en numerosos campos como la economía, medicina, ingeniería, transporte, comunicación, milicia entre otros; actualmente aplicaciones como Facebook, Twitter e Instagram utilizan algoritmos de programación que se retroalimentan constantemente del comportamiento de los usuarios con dichas aplicaciones, sugieren productos y contenido de acuerdo al análisis de los gustos de los mismos, así como servicios de atención al cliente que utilizan mecanismo de reconocimiento de escritura y habla que por medio de asistentes virtuales (chatbot) responden a las preguntas de los usuarios simulando ser una

persona, además de un creciente número de aplicaciones imperceptibles que ya forman parte de la cotidianidad del ser humano.

En este sentido, existen varios modelos de aplicación de IA, entre los cuales se destaca por su amplia aplicación y utilidad en la sociedad actual el Modelo de Visión computacional, el cual tiene la capacidad de extraer información de una imagen, clasificándose en categorías de entrenamiento que facilitan la identificación y detección de cada uno de los elementos encontrados dentro de la misma, ampliando además su uso a generación de imágenes e identificación de acciones.

En función de lo planteado, se presenta el marco referencial donde se ubica el propósito de esta investigación orientada a *Desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad*, usando el modelo de Visión computacional, el cual a través de la detección de imágenes permitirá entender e interpretar diferentes características de una imagen, así como la segmentación, localización y reconocimiento de ciertos objetos dentro de la misma.

Todo esto con el propósito de ofrecer una alternativa efectiva en cuanto a facilidad de uso, aplicación y costos, que responda de forma eficiente a la limitante de muchos sistemas de seguridad actuales que solo se basan en servicios de video pasivos o no poseen mecanismos de alerta en vivo sobre la detección de personas extrañas dentro de una propiedad, lo que disminuye las probabilidades de una respuesta rápida y efectiva ante un problema como el irrumpimiento de morada.

En este sentido, la investigación se fundamenta en los postulados teóricos de autores en el campo de IA, machine learning y visión computacional, haciendo un recorrido de la literatura sobre los trabajos que anteceden al presente con el fin de

indicar su relación y sustentación con el estudio en proceso. Del mismo modo se fundamenta en el paradigma de la investigación holística, más específicamente el tipo de investigación proyectiva, con un diseño de fuente mixta, relacionado a la integración del diseño documental y de laboratorio, utilizando como técnica de recolección de datos, la recopilación y revisión de fuentes de información a través de los distintos medios disponibles (físicos, digitales), categorizando la información basada en el objeto de estudio, mediante un enfoque descriptivo y analítico.

Así mismo, el presente estudio se estructura en cinco (5) capítulos con el propósito de reflejar el proceso investigativo. En el Capítulo I se plantea el enunciado del problema, donde se exponen las razones, factores que intervienen y que dan vida a la investigación, seguida de los objetivos, justificación y delimitación.

El Capítulo II se enfoca en el Marco Teórico, donde se concentran las investigaciones relacionadas con el estudio. De igual forma, se presentan los fundamentos teóricos de diferentes autores empleados como base para la operacionalización de las variables, así como para el análisis posterior de los resultados.

En cuanto al Capítulo III, se desarrolla el Marco Metodológico, conformado por el tipo y diseño de investigación, técnica de recolección de datos e instrumentos seleccionados, proceso de validación y confiabilidad de los resultados.

Por último, la aplicación de las técnicas para la preparación de datos, desarrollo, análisis e interpretación de los resultados obtenidos a partir de los capítulos IV y V, donde se describe el procedimiento llevado a cabo para la realización de la

investigación, seguido de las conclusiones y recomendaciones, finalizando con la presentación de referencias bibliográficas.

# CAPÍTULO I

## EL PROBLEMA

### Planteamiento del Problema

Desde el desarrollo del pensamiento humano, la necesidad de soluciones concretas a problemas cotidianos llevó a la elaboración de herramientas y mecanismos que se fueron transformando de elementos rudimentarios como técnicas para generar fuego a complejos mecanismos de producción de artículos de alcance masivo, construcción de alumbrados y medios de transporte como ferrocarriles y automóviles, dando inicio a la era de la tecnología como la aplicación de la ciencia de forma ordenada sobre problemas concretos, transformándose en simplificación y mejora en tiempo y recursos de numerosas tareas, que con el pasar de los años y la acumulación de conocimiento permitieron que la humanidad experimentará un gran crecimiento económico y social.

En este sentido, Karl Theodor Jaspers (1960) filósofo alemán, dijo “La tecnología es solo un medio, ni bueno ni malo en sí. Todo depende de lo que el hombre haga con ella, para qué propósito le sirva, bajo en qué condiciones la utilice” (Definición de la tecnología, parr. 1).

Por lo que de acuerdo con lo planteado, resulta paradójico que en tiempos tan hostiles como la segunda guerra mundial, la tecnología haya evolucionado de manera exponencial, comenzando con el desarrollo en 1940 del primer calculador electrónico (ENIAC), diseñado por el ingeniero eléctrico y pionero de la informática Presper J.

Eckert y el físico John W. Mauchly, cuyo propósito fue calcular las trayectorias de proyectiles y bombas, dando paso al primer programa y ordenador digital electrónico con control de almacenamiento de memoria, lo que suscitó en los años posteriores en un asentamiento comercial de gran demanda, con modelos más accesibles en forma y costo que con la ayuda del desarrollo de las redes de comunicación electrónica y digital conectan hoy en día a personas en todo el mundo al instante.

Y es así, que de la mano del internet la globalización del mundo se aceleró y con ello el desarrollo de tecnología mucho más ambiciosa, la cual no solo se enfoca en resolver problemas específicos, sino que se prepara y adelanta a ellos, interviniendo de manera activa en el desarrollo del conocimiento científico en áreas como la ingeniería, milicia, con gran impacto en la medicina con la implementación de equipos avanzados de análisis biológicos y de imágenes que permiten ofrecer diagnósticos rápidos y predictivos que mejoran la calidad de vida, poniendo a la orden de la educación y enseñanza diversos medios digitales y de comunicación que estimulan el crecimiento intelectual.

Por lo que rompiendo paradigmas donde se creía imposible superar la barrera humano-máquina, la tecnología dio un paso gigante con la aplicación de razonamiento llevado a cabo por máquinas, conocido como inteligencia artificial (IA), la cual de acuerdo a Andreas Kaplan y Michael Haenlein (2018) se define como “la capacidad de un sistema para interpretar correctamente datos externos, para aprender de dichos datos y emplear esos conocimientos para lograr tareas y metas concretas a través de la adaptación flexible” (p. 15).

Considerando lo anterior, puede resultar complejo pensar que una IA es capaz de percibir, razonar, aprender y resolver problemas tal como lo hace un ser humano, contemplando las condiciones y variables del entorno que lo rodea, todo esto gracias a algoritmos que se alimentan constante de datos y del análisis de ejemplos que le permiten reconocer patrones, basando el aprendizaje en repetición y recursividad de los datos para comprenderlos, ordenarlos y clasificarlos.

La IA no es algo reciente, se considera que el primer trabajo en IA se debe a Warren McCulloch y Walter Pitts (1943) sobre un modelo de red neuronal artificial, que inspirado en la conducta del cerebro humano, intentaba modelar el comportamiento de una neurona a través de unidades de cálculo (modelos matemáticos) que por medio de funciones lógicas y digitales pueda entender la manera en que se responde a estímulos del mundo exterior y aprender a reaccionar en función de ellos, lo que sirvió como inspiración en el diseño de otros modelos, convirtiéndose así en un estudio de gran influencia en el desarrollo de la ciencia de la computación e IA actual.

En este sentido, actualmente existen varios modelos de IA basados en el aprendizaje, la visión y la robótica, entre los que se destacan el modelo de procesamiento de lenguaje natural (LPM), el cual permite interpretar, procesar y generar resultados sobre textos basados en el lenguaje natural de comunicación del ser humano por medio de la clasificación del texto (NERD), el cual toma el cuerpo del texto y es capaz de predecir la connotación del mismo si es negativo, positivo o neutral, así como el reconocimiento de entidades nombradas la cual es capaz de identificar ciertos campos o variables dentro de un texto, como lo es reconocer nombres de personas, países y empresas.

Aunado a lo anterior, cuando se trata de decisiones estratégicas, el modelo de toma de decisiones que propone la IA permite evaluar múltiples escenarios, puntos de vista y resultados potenciales, que disminuyen el margen de error al no estar alterados por factores emocionales inherentes al ser humano, actuando como consejeros estratégicos muy útiles a nivel empresarial, donde el recurso más valioso es la cantidad de datos y los resultados dados luego del análisis de estos. Otro modelo como la generación de datasets sugiere a través de grandes colecciones de datos de cualquier tipo, interpretar y generar resultados de forma rápida y efectiva, la gran cantidad de datos acelera la curva de aprendizaje y de análisis de este modelo de IA volviéndose más preciso.

Así mismo, en la robótica hay modelos de IA que han evolucionado para entender y responder al lenguaje natural del ser humano, integrando el razonamiento lógico y la actividad física. Actualmente, se desarrolla robótica con apreciación basada en el tacto por medio del uso de sensores, brindándole al robot un sentido de percepción y representación visual del objeto a través del tacto. En este sentido Rainer Bischoff (2020), director de KUKA principal fabricante de robot industriales de mundo, plantea “El robot con inteligencia artificial aprende de modo en que lo haría un niño, con tareas simples de reconocimiento y clasificación de piezas, pero de manera automática, para así logra dar un paso más con la capacidad de autoaprendizaje, reconociendo piezas y ensamblando las mismas, adaptándose al entorno de forma independiente” (Cómo cambia nuestra sociedad la inteligencia artificial. DW, Documental. 2020, 1:57).

Resulta asombroso todos los alcances que tiene la implementación de los modelos de IA mencionados anteriormente, de acuerdo con esto, uno de los modelos

más usados en la actualidad, de gran interés y utilidad para la presente investigación es el modelo de visión computacional, el cual tiene la capacidad de extraer información de una imagen para luego clasificarla en categorías que faciliten la identificación y detección de cada uno de los elementos encontrados dentro de la misma. Basado principalmente en la visión humana, trabaja en el modelado y réplica de esa visión empleando software y hardware avanzado, con el propósito de desarrollar teorías y métodos hacia la extracción automática de información útil contenida en imágenes.

El modelo de visión computacional encontró inicialmente su apogeo en la milicia con la detección de unidades enemigas y envío de misiles hacia el objeto identificado. Este modelo ha sabido combinarse de forma extraordinaria con la robótica, intervenido exitosamente en el campo de la medicina en operaciones de implantes de cadera mediante asistentes robot que utilizan técnicas de visión computacional para crear un modelo tridimensional de la anatomía interna del paciente que después utiliza un control robotizado para guiar el implante, este es un pequeño ejemplo de cómo el modelo parte de una imagen 2D, para reconstruir una visión en tres dimensiones (3D), lo que le permite generar y describir la escena, eliminando todo el ruido posible en la imagen, facilitando la interpretación y entendimiento de la misma.

De esta manera, en materia de defensa y seguridad ciudadana, siguiendo el modelo de visión computacional, se han implementado prototipos de IA en sistemas de vigilancia y seguridad que permiten reconocer la identidad de los ciudadanos en tiempo real logrando identificar y ubicar aquellos que puedan estar dentro de los patrones reconocidos como “inseguros” o “peligrosos” como terroristas o evasores de la justicia y en este sentido responder de forma anticipada a eventos delictivos.

Si bien resulta una solución efectiva para un problema tan grande no se encuentra a disposición de todos, debido a los altos niveles de inversión económica y preparación en cuanto a conocimiento que implica la implementación de estos sistemas; sin embargo, países con gran poder económico como China, Estados Unidos y Alemania, han demostrado la efectividad de estos sistemas inteligentes mitigando los efectos de grupos criminales y acciones delictivas individuales, a través del uso efectivo de estos sistemas, por lo que Latinoamérica no se queda atrás dando pequeños pasos en países como Ecuador, Chile y Argentina por medio alianzas con China, dando resultados positivos en reconocimiento temprano de situaciones delictivas e intervención rápida de la fuerza policial.

En este sentido, la presente investigación propone desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad aplicando inteligencia artificial (IA), con un alcance orientado al empleo particular, donde su aplicación se puede extender a cualquier espacio físico donde se requiera controlar situaciones que se reconozcan como perjudiciales o peligrosas para sus ocupantes, por medio de un mecanismo de seguridad que a través del aprendizaje de las características en cuanto a formas y movimientos de los objetos detectados en la imagen, pueda hacer un análisis para decidir si el objeto observado es una persona y si su apariencia es desconocida.

Por lo que a los fines de la investigación, tal como se anticipó en puntos anteriores, se hará una búsqueda y análisis de modelos de IA en visión computacional así como materiales asequibles que en conjunto permitan el desarrollo de un sistema el cual a través de la detección de imágenes puede interpretar diferentes características

como segmentación, localización y reconocimiento de todos los elementos dentro de una imagen, permitiendo a través de un entrenamiento recursivo de datos poder generar patrones que le permita responder de manera rápida y efectiva mediante el envío de alertas a los usuarios cuando se detecte la presencia de personas desconocidas; si bien no es una solución definitiva la presente investigación pretende ofrecer una alternativa mucho más accesible en cuanto a costos, entendimiento y aplicación de un sistema de seguridad con IA.

#### Formulación del Problema:

Con base al planteamiento anterior, surge la siguiente interrogante: En virtud de las limitaciones tanto en oferta como en costo de sistemas de seguridad, que en el mejor de los casos se limita a agregar detecciones sin aplicación de reconocimiento, ¿Es posible desarrollar a bajo costo y con recursos limitados un sistema de reconocimiento de imágenes para emitir alertas de seguridad de forma rápida y precisa usando inteligencia artificial (IA)?.

## **Objetivos de la Investigación**

### ***Objetivo General***

Desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad.

### **Objetivos Específicos**

1. Identificar las herramientas de hardware y software a requerir para el desarrollo de la IA.
2. Aplicar algoritmos inteligentes de reconocimiento de imágenes, bajo el modelo de visión computacional.
3. Entrenar la IA en el reconocimiento de imágenes basado en patrones establecidos.
4. Estudiar métricas que permitan comprobar el funcionamiento de la IA desarrollada.
5. Realizar las pruebas de la IA en el envío de alertas de seguridad en un ambiente real.

### **Justificación**

Desde el punto de vista teórico, la investigación propuesta busca mediante la aplicación de teorías, conceptos, metodologías y demás referencia bibliográficas sustentar de forma clara y efectiva el tema en estudio el cual está orientado a desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad aplicando inteligencia artificial (IA), con la finalidad de contar con un marco de conocimiento actualizado que sustente de forma oportuna la investigación y que

además sirva como antecedente para futuras investigaciones, sustentado en autores como Bostrom, N. (2016), Russell, S., Norvig P (2004), Kurzweil R (2012), entre otros.

Desde la perspectiva metodológica este trabajo propicia la investigación científica a través de la aplicación de parámetros y teorías orientadas hacia la consecución de los objetivos planteados, siguiendo un tipo de investigación proyectiva, con un diseño de fuente mixto (laboratorio y documental), la cual busca a través del desarrollo y puesta en marcha del objetivo general, sustentar de manera efectiva la investigación, brindando información confiable y precisa sobre el reconocimiento de imágenes a través de la aplicación de inteligencia artificial (IA) y como este puede responder de modo exitoso en un entorno específico, sirviendo así como base metodológica y experimental para futuros estudios relacionados con el tema.

Considerando la problemática, desde un enfoque práctico, en cuanto a los objetivos de la investigación, dichos resultados permitirán encontrar soluciones concretas sobre la aparente ausencia de sistemas de seguridad efectivos en tiempo real que sean de bajo costo con una fácil aplicación, que al ser implementados en una vivienda puedan responder de forma rápida y efectiva mediante el envío de alertas a sus usuarios sobre la detección de personas desconocidas.

Así mismo, su relevancia social se fundamenta dentro del ámbito de la cotidianidad del hogar, donde para muchos resulta necesario estar al tanto de estado de sus viviendas y en especial de los seres queridos que se encuentran en ellas, siendo necesario estar informados de manera inmediata ante cualquier situación que atente con la integridad física de estos por medio de un sistema de seguridad con tecnología inteligente que resulte entendible y accesible en cuanto a costos y aplicación.

## **Delimitación de la investigación**

De acuerdo al objetivo de la investigación orientado a desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad, se plantea el problema sobre si es posible o no desarrollar un sistema de seguridad con reconocimiento de rostros con recursos limitados y de bajo costo, con una fácil aplicación que responda de forma precisa y efectiva a la demanda actual de sistemas de seguridad, sustentándose en teorías de autores influyentes en temas de IA y visión computacional, así como demás fuentes físicas y electrónicas que servirán de soporte en el desarrollo de la investigación.

En este sentido, el área geográfica donde se desarrollará el estudio se limitará a la residencia del investigador, ubicada en la ciudad de Buenos Aires, Argentina. El proyecto de investigación tendrá un periodo aproximado de realización desde mayo del 2020 hasta octubre del 2021.

Dicha investigación se enmarca bajo los lineamientos propuestos por la Universidad Valle del Momboy en la línea de investigación de tecnología e innovación para el desarrollo local.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

Este capítulo comprende la fundamentación teórica del estudio, el mismo incluye los antecedentes relacionados con la investigación, bases teóricas, el contraste entre diversos autores para la definición de conceptos, métodos y aplicaciones, así como el glosario de términos básicos.

Según Baena (2014), “las teorías, los sistemas de pensamiento y las ideas de algún autor son fundamentales para entender la realidad. Cada una de ellas es una forma de interpretarla, lo que nos ayuda en su comprensión y con ello podemos mejorar, resolver o encontrar caminos para que un problema se suprima, se minimice, se rectifique” (p. 112). En este sentido, el marco teórico es la exposición y análisis de la teoría o teorías que sirven como fundamento para explicar los antecedentes e interpretar los resultados de la investigación.

#### **Antecedentes de la Investigación**

Según Arias (2006), los antecedentes “se refieren a todos los trabajos de investigación que anteceden al nuestro, es decir, aquellos trabajos donde se hayan manejado las mismas variables o se hallan propuestos objetivos similares; además sirven de guía al investigador y le permiten hacer comparaciones y tener ideas sobre cómo se trató el problema en esa oportunidad” (p. 05).

Considerando lo planteado, a continuación, se presentan los aportes tomados de diferentes investigadores relacionados con el tema objeto de estudio. Es importante señalar que los aportes servirán de apoyo a la presente investigación; dada su aplicación práctica, relevancia social, estrategias, así como los fundamentos teóricos y de metodología que lo sustentan, permitiendo visualizar aspectos relevantes para la construcción tanto del marco teórico como metodológico, en función de los aportes reportados.

Ibarra (2017), en su artículo científico titulado Generación de dataset para problema de visión computarizada, plantea el problema de la conversión de imágenes del lenguaje de señas mexicano a sus respectivas letras del alfabeto; orientada principalmente en resolver el problema al momento de brindar otras alternativas que permitan a personas con problemas auditivos o de habla interactuar con las computadoras, así como concientizar e incluso crear métodos de enseñanza para mejorar la comunicación a través de los medios digitales de las personas con dificultad de habla y escucha.

En este sentido, para iniciar con el proceso de la obtención del dataset, el primer paso fue adquirir la mayor cantidad de imágenes posibles relacionadas a la problemática, se buscaron imágenes de las señas donde la mano sea fácilmente distinguible en la imagen y la forma de la seña que se está interpretando sea clara. Esto con el propósito de eliminar ruido en las imágenes y poder obtener la posición de la mano con mayor facilidad, permitiendo normalizar los datos.

Las pruebas fueron ejecutadas bajo un ambiente controlado donde la luz no tuviera variación y el fondo de la imagen sea distinguible del color de la mano,

obteniendo una lectura por lotes de 300 imágenes similares en un periodo de dos minutos, generando como resultado una imagen que muestra por pantalla la forma de la letra que se quiere representar a partir de cada imagen de la seña de mano captada, obteniendo así con las sesiones realizadas para la adquisición de los datos un total de 900 imágenes por letra las cuales suman un total de 21600 imágenes para clasificar 24 clases distintas, 10800 ejemplos para aprender la forma general de cada letra representada.

Concluyendo con una tecnología elaborada sin muchos recursos, que responder de manera aceptable a una necesidad de comunicación muy presente en la sociedad, como es la ausencia o pocas alternativas que existen para que personas con problemas auditivos o de habla se pueda comunicar a través de un computador y quizás concientizar a la sociedad sobre cómo estos temas se puede convertir en un hilo conductor que permita que más proyectos como estos se puedan seguir desarrollando y cuyos resultados sean accesibles para aquel que lo necesite.

En relación con la investigación planteada, los aportes de la misma servirán como apoyo para el presente estudio, debido a su enfoque social y práctico, haciendo uso del modelo de visión computacional y cómo a través del entrenamiento adecuado por medio de datasets se pueden obtener resultados medibles y satisfactorios; sin dejar a un lado su incidencia social, que al igual que esta investigación busca proporcionar una alternativa mucho más accesible en cuanto a costos y recursos ante una problemática que plantea la presente investigación, como lo es el punto de seguridad y vigilancia.

Seguidamente, García (2018) en su investigación científica para la Universidad Complutense de Madrid, titulada Métodos de visión por computador para la detección automática de líneas de cultivo de curvas/rectas y malas hierbas en campos de maíz, tuvo como objetivo desarrollar métodos de visión por computador concretos que en consecuencia permitan responder a las dificultades que hay al momento de discriminar entre cultivo y malas hierbas en sistemas de visión por computador a bordo de vehículos autónomos para agricultura.

Las condiciones no controladas dadas por los entornos agrícolas de exterior, las diferentes alturas y volúmenes de plantas manifestadas en sus distintos estados de crecimiento y la presencia de discontinuidades a lo largo de las líneas de cultivo debido a una mala germinación o defectos durante la siembra, dificulta la calidad de las imágenes y por defecto la detección de líneas de cultivo y discriminación de malas hierbas por parte de los vehículos autónomos.

Por lo que a través de la proposición de dos nuevos métodos, se pudo concluir con la implementación de: (a) guiado preciso en vehículos autónomos, (b) tratamientos específicos, incluyendo la eliminación de malas hierbas, situadas entre las líneas; diseñados con la robustez requerida para abordar el problema de las condiciones adversas, la cual consta de tres fases consecutivas: (i) segmentación de la imagen, (ii) identificación de los puntos de comienzo de las líneas de cultivo y (iii) detección de las propias líneas. Obteniendo resultados satisfactorios para su uso en vehículos autónomos equipados con sistemas de visión computacional en tiempo real para agricultura de precisión, comparando el rendimiento de los métodos propuestos, se

consiguió una precisión entre el 86.3% y el 92.8%, con tiempos de procesamiento menores que 0.64 segundos por imagen.

De acuerdo con lo planteado, dicha investigación científica resulta de gran interés para la presente gracias a los resultados dados, donde refleja como la implementación de métodos específicos sobre una tecnología de visión computacional existente puede mejorar su rendimiento y precisión, brindando pautas y estrategias muy importantes sobre los métodos a aplicar para la selección y detección de imágenes precisas y cómo dicha elección va de la mano del contexto de la aplicación, bondades y limitaciones de la misma, prestando especial atención a su aplicación en tiempo real sobre entornos no controlados.

Por su parte, López (2016), en su trabajo especial de grado, titulado Sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos en áreas de almacenamiento de las Pymes, para la universidad Técnica de Ambato de Ecuador; tiene como objetivo principal el reconocimiento de patrones de una persona en las áreas de almacenamiento de las PYMES.

Para el desarrollo del objetivo propuesto, se hizo hincapié en la detección de bordes y desarrollo de algoritmos de procesamiento en tiempo real, consiguiendo así un sistema robusto con eficiencia en la detección de individuos con un 98% de precisión con 3 segundos de respuesta, condición eficaz para la menor sensación posible de ruido, tratándose de una secuencia de fotogramas procesados con técnicas de visión artificial a una distancia máxima aproximada de entre 5,5 m a 10,5 m.

Concluyendo con algoritmos de extracción de rasgos óptimos en el 99% a la visualización del patrón humano en forma binaria, que contribuye con un diseño e

implementación de un sistema de reconocimiento de patrones para la alerta automática de intrusos, con una interfaz gráfica de usuario en un 98% útil para la adaptación al dominio de la propuesta y de fácil manipulación para el usuario.

En este sentido, dicha investigación ofrece información y resultados relevantes para la presente investigación, debido a los algoritmos de procesamiento que sirven de referencia, ya que al ser una implementación cuya aplicación práctica se asemeja a la presente investigación brinda pautas importantes sobre diseño e implementación de los algoritmos, así como recomendaciones efectuadas en el desarrollo y la implementación del sistema, haciendo especial enfoque sobre el modelo de visión computacional debido a su facilidad de aplicación y recursos disponibles.

### **Bases teóricas**

Para Arias (2006), las bases teóricas están formadas por: “un conjunto de conceptos y proposiciones que constituyen un punto de vista o enfoque determinado, dirigido a explicar el fenómeno o problema planteado” (p. 36). Las mismas permiten desarrollar aspectos conceptuales del tema objeto de estudio, resultando evidente la revisión y documentación necesaria de teorías y demás estudios vinculados con la investigación.

De acuerdo con lo planteado, resulta necesario conocer por medio de una verificación y documentación exhaustiva de fuentes bibliográficas, los elementos teóricos que sirven como base de referencia y sustentación de la investigación, brindando una visión contextual de la misma, por lo que a continuación se citarán

conceptos, descripciones y puntos de vista muy particulares de grandes autores que resultan apropiados y de gran importancia para la investigación.

## **Historia de la Inteligencia Artificial (IA)**

De acuerdo con Bostrom (2016), La IA vio sus inicios durante el verano de 1956 en medio de tiempos de desesperanza y desesperación, en el Dartmouth College (universidad privada, Nuevo Hampshire, Estados Unidos); diez científicos se reunieron para celebrar un taller de seis semanas de duración para compartir su interés por las redes neuronales, la teoría de autómatas y el estudio de la inteligencia; considerado como el punto de partida de la inteligencia artificial como campo de investigación, donde muchos de sus participantes serían reconocidos posteriormente como pioneros.

Seis décadas transcurridas desde este audaz comienzo, el campo de la inteligencia artificial ha pasado por periodos de entusiasmo generalizado y altas expectativas, alternando con períodos de retroceso y decepción. Para contrarrestar afirmaciones de investigadores de la época donde se refuta la posibilidad de que las máquinas pudieran realizar tareas por sí solas, llevó a los investigadores de la IA a crear pequeños sistemas dentro de un ámbito limitado, con lo que aportaron pruebas conceptuales y demostraron que en principio dicha autonomía podría ser realizado por un hardware. A mediados de los años 70 y con la comprensión de que muchos proyectos en IA no lograban cumplir sus promesas iniciales, lo que condujo a la aparición del periodo de “invierno de la IA”, que llevó a la disminución de fondos y aumento del escepticismo.

Así mismo, el autor plantea que una nueva primavera llegó a principios de la década de los 80, cuando Japón lanzó su proyecto de Sistemas computacionales de quinta generación, la cual nació de la necesidad de descubrir la fórmula para el éxito económico en tiempos de posguerra, periodo en que líderes gubernamentales y empresariales occidentales invirtieron grandes cantidades en IA, por lo que otros países como Estados Unidos siguieron su ejemplo, temporada de crecimiento que se extinguió en la década de los 80 con el fracaso de grandes proyectos que eclipsaron los pequeños avances realizados.

El optimismo sobre la IA se reavivó en los 90 gracias a la introducción de nuevas técnicas que parecían ofrecer alternativas al paradigma logicista tradicional; estas incluían avances en las redes neuronales ya existentes desde los años 50, modelos que aprenden de la experiencia, buscar patrones y generalizar ejemplos. Los métodos basados en la evolución, como los algoritmos genéticos, contribuyen a otro enfoque con un impacto menor que las redes neuronales, el cual mantiene una población de soluciones candidatas (estructuras de datos o programas) generando nuevas soluciones aleatoriamente mediante mutaciones o variantes que combinan la población existente. Lo que llevo en años posteriores al desarrollo de IA alrededor del aprendizaje artificial y de la resolución creativa de problemas, donde cada avance significaba un reto y sigue progresando (pp. 5-9).

En este sentido, Benítez, Escudero y Kanaan (2014), exponen el origen y evolución de la IA desde un contexto científico matemático con el Teorema de incompletitud de Gödel de 1931, un conjunto de teoremas de lógica matemática que establecen las limitaciones inherentes a un sistema basado en reglas y procedimientos

lógicos (como lo son todos los sistemas de IA). Tras los primeros trabajos en IA de los años cincuenta, en la década de los sesenta se produjo un gran esfuerzo de formalización matemática de los métodos utilizados por los sistemas de IA.

En los años setenta, en parte como respuesta al test de Turing, se produjo el nacimiento de un área conocida como procesado del lenguaje natural (NLP, Natural Language Processing), una disciplina dedicada a sistemas artificiales capaces de generar frases inteligentes y de mantener conversaciones con humanos. El NLP ha dado lugar a diversas áreas de investigación en el campo de la lingüística computacional, incluyendo aspectos como la desambiguación semántica o la comunicación con datos incompletos o erróneos. A pesar de los grandes avances en este ámbito, sigue sin existir una máquina que pueda pasar el test de Turing.

Dichos autores plantean que esto no es tanto debido a un fracaso de la IA como a que los intereses del área se han ido redefiniendo a lo largo de la historia. En 1990, el controvertido empresario Hugh Loebner y el Cambridge Center for Behavioral Studies instauraron el premio Loebner, un concurso anual en el que se premia al sistema artificial que mantenga una conversación más indistinguible de la de un humano. Sin embargo, hoy en día, la comunidad científica considera que la inteligencia artificial debe enfocarse desde una perspectiva diferente a la que se tenía en los años cincuenta, pero iniciativas como la de Loebner expresan el impacto sociológico que sigue teniendo la IA en la sociedad actual.

Así mismo, en los años ochenta empezaron a desarrollarse las primeras aplicaciones comerciales de la IA, fundamentalmente dirigidas a problemas de producción, control de procesos o contabilidad. Con estas aplicaciones aparecieron los

primeros sistemas expertos, que permitían realizar tareas de diagnóstico y toma de decisiones a partir de información aportada por profesionales expertos. En torno a 1990, IBM construyó el ordenador ajedrecista Deep Blue, capaz de hacerle frente a un gran maestro de ajedrez utilizando algoritmos de búsqueda y análisis que le permitían valorar cientos de miles de posiciones por segundo.

Más allá del intento de diseñar robots humanoides y sistemas que rivalicen con el cerebro humano en funcionalidad y rendimiento, el interés hoy en día es diseñar e implementar sistemas que permitan analizar grandes cantidades de datos de forma rápida y eficiente. En la actualidad, cada persona genera y recibe a diario una gran cantidad de información no solo a través de los canales clásicos (conversación, carta, televisión) sino mediante nuevos medios que nos permiten contactar con más personas y transmitir un mayor número de datos en las comunicaciones (Internet, fotografía digital, telefonía móvil).

Aunque el cerebro humano es capaz de reconocer patrones y establecer relaciones útiles entre ellos de manera excepcionalmente eficaz, es ciertamente limitado cuando la cantidad de datos resulta excesiva. Un fenómeno similar ocurre en el ámbito empresarial, donde cada día es más necesario barajar cantidades enormes de información para poder tomar decisiones. La aplicación de técnicas de IA a los negocios ha dado lugar a ámbitos de reciente implantación como la inteligencia empresarial, business intelligence o a la minería de datos, data mining. En efecto y de acuerdo a los autores, hoy más que nunca la información está codificada en masas ingentes de datos, de modo que en muchos ámbitos se hace necesario extraer la

información relevante de grandes conjuntos de datos antes de proceder a un análisis detallado (pp. 13-76).

Sin embargo, Berzal (2018), hace una apertura a la historia de la IA desde las redes neuronales artificiales, más conocidas actualmente bajo la denominación anglosajona de deep learning, el autor plantea que es la tercera vez en la historia de la Inteligencia Artificial en la que algo así sucede. Tras experimentar varios ciclos expansivos y sufrir severas correcciones que las pusieron al borde de la desaparición, actualmente disfrutan de un auge espectacular que, en cierto modo, se asemeja al experimentado por los sistemas expertos en los años 80 o la minería de datos en el cambio de siglo. El boom de la industria de los sistemas expertos ya pasó a la historia. Sin embargo, la minería de datos ha ido evolucionando hasta transformarse en “ciencia de datos”, el trabajo más interesante del siglo XXI según Hal Varian, economista jefe de Google.

Del mismo modo y tal como lo plantea el autor, todo parece indicar que los grandes logros de las técnicas basadas en deep learning a la hora de resolver problemas que se habían mostrado intratables hasta la fecha harán gozar a las redes neuronales artificiales de una nueva época dorada. Según la consultora Gartner, el 80% de los científicos de datos usarán herramientas de deep learning. Aunque los orígenes de las redes neuronales artificiales se remontan a los comienzos de la Inteligencia Artificial, en los años 40 del siglo XX y el deep learning en sí despegará en 2006, el éxito real del deep learning se debe a su aplicación práctica en problemas de interés para la industria tecnológica, como el reconocimiento de voz o la visión artificial (p. 19).

De acuerdo con lo planteado, resulta muy interesante el enfoque que le da cada autor a la evolución histórica de la IA y lo que sus aportes brindan a la investigación gracias a una perspectiva variada, pero respetando los acontecimientos históricos y sus principales protagonistas, en especial Berzal (2018), quien plantea el surgimiento de la IA a partir de la deep learning, que en comparación a los algoritmos más tradicionales propone grandes aportes al desarrollo de la visión artificial en análisis y diagnóstico de imágenes, evolucionando en redes especializadas en detectar diferentes aspectos de la visión, bordes, inclinación de las líneas, la simetría e incluso áreas íntimamente relacionadas con el reconocimiento de rostros y la expresión emocional, premisas que resultan de gran interés para la presente investigación.

### **Inteligencia artificial (IA)**

Gollapudi (2019), plantea la IA como el campo de la informática que implica permitir que las computadoras se comporten como humanos o realicen tareas que generalmente requieren inteligencia humana, definición que ha evolucionado desde su primera referencia en 1956 en una conferencia de Dartmouth, desde emular cómo funciona el cerebro humano hasta resolver problemas complejos y enfocados a hacer todo lo que un humano puede hacer, como ver, oír, comunicarse, actuar, aprender, percibir, pensar, decidir, demostrar emoción y compasión, interactuar con el entorno y más. Los avances de la IA en 2012 con visión, reconocimiento de idioma y vehículos autónomos cambiaron la forma en que se ve la IA en la actualidad (p. 3).

Por su parte, Benítez, Escudero y Kanaan (2014), definen la IA como una disciplina académica relacionada con la teoría de la computación cuyo objetivo es

emular algunas de las facultades intelectuales humanas en sistemas artificiales. Con inteligencia humana nos referimos típicamente a procesos de percepción sensorial (visión, audición, etc.) y a sus consiguientes procesos de reconocimiento de patrones, por lo que las aplicaciones más habituales de la IA son el tratamiento de datos y la identificación de sistemas, no excluyendo sus inicios en la década del 1960, orientada a la resolución de problemas de carácter más abstracto como la demostración de teoremas matemáticos, la adquisición del lenguaje, el jugar a ajedrez o la traducción automática de textos (p. 10).

Sin embargo, Berzal (2018), dice que, a la hora de definir la IA, algunos autores priorizan centrarse en sistemas que actúan, como los robots, mientras que otros prefieren focalizar su atención en el desarrollo de sistemas capaces de razonar o pensar. Tanto en un bando como en otro, hay quien define la IA con respecto al ser humano (asumiendo que es inteligente, algo de lo que uno puede tener sus dudas más que justificadas). Otros prefieren definir la inteligencia como un fenómeno asociado a la racionalidad, asumiendo que esa sea su característica definitoria.

Por lo que el autor expone que es prácticamente imposible alcanzar una definición de IA con la que todo el mundo esté de acuerdo, quedándose con la definición clásica de Elaine Rich (2012), de la Universidad de Texas en Austin donde expone la IA como “el estudio de cómo hacer que los ordenadores hagan cosas que, por ahora, los humanos hacemos mejor” (pp. 9-11).

De acuerdo con lo expuesto, resulta muy interesante para la investigación el planteamiento de Berzal (2018), el cual abre su definición más allá de lo académico y lo interpreta desde la evolución conceptual dada por autores pioneros en temas de

Inteligencia artificial, sintetizando la definición de IA, permitiendo así conocer un concepto a partir de las características propias del ser humano y sus sentidos, en especial el de la visión en la cual se enfoca la presente investigación.

## **Aplicación de la Inteligencia Artificial**

A continuación, se muestran las distintas aplicaciones en IA, haciendo especial énfasis en el modelo de visión computacional y campos relacionados.

Benítez, Escudero y Kanaan (2014), plantean que en la actualidad la IA se ha consolidado como una disciplina que permite diseñar aplicaciones de gran utilidad práctica en numerosos campos. Actualmente, existe una enorme lista de ámbitos de conocimiento en los que se utilizan sistemas de IA, entre los que son de especial relevancia la minería de datos, el diagnóstico médico, la robótica, la visión artificial, el análisis de datos bursátiles o la planificación y logística.

En la robótica, uno de los campos de investigación actual con mayor proyección es el aprendizaje adaptativo, en el que un sistema robotizado explora diferentes configuraciones con el objetivo de realizar movimientos complejos (caminar, agarrar un objeto, realizar una trayectoria), siguiendo un proceso de exploración y aprendizaje similar al que realiza un recién nacido durante los primeros meses de vida, explorando diferentes movimientos de forma aleatoria, midiendo las variables de cada articulación y comprobando en cada momento el grado de éxito alcanzado por cada secuencia de acciones (altura del centro de masas, desplazamiento horizontal, fluctuaciones en la posición vertical, velocidad de desplazamiento) a través de sensores ópticos y de tacto que responden a toda la información dada por el entorno que lo rodea.

Dentro del campo de la medicina, los autores plantean que un sistema de diagnóstico clínico, por ejemplo, puede partir de imágenes médicas como ecos, radiografías, serie temporal de la frecuencia del ritmo cardíaco de un paciente o de un conjunto de datos con valores extraídos de análisis de sangre o de orina, extraer información clínica útil, que permita clasificar a los pacientes según diferentes patologías y en especial brindar un tratamiento y diagnóstico preciso y en el mejor de los casos predictivo.

En cuanto al reconocimiento de rostros, las imágenes tomadas en diferentes condiciones (diferente expresión facial, ropa, iluminación, exposición, fondo, etc.), de forma que sean representativas de las características faciales de cada persona, hoy en día existen sistemas que identifican las características principales de cada una de las fotos y las clasifica en un determinado número de grupos. Entre las características utilizadas pueden aparecer, entre otras, el color de los ojos, el grosor de los labios o el perímetro de la cabeza; donde la fase de predicción parte de una imagen de uno de los individuos que no haya sido incluida en el conjunto de datos de entrenamiento, donde el sistema de IA calcula las características faciales del nuevo individuo, siendo capaz de identificar esa imagen como perteneciente a uno de los grupos definidos durante el proceso de aprendizaje (pp. 10-16).

En este orden de ideas, Benítez, Escudero y Kanaan (2014), plantean las siguientes como las principales aplicaciones en inteligencia artificial con un especial enfoque en la visión computacional (ver tabla 1), idóneo para la presente investigación (p. 18).

**Tabla 1:**

*Principales ámbitos de aplicación de los sistemas de inteligencia artificial.*

Áreas	Aplicaciones
Medicina	<ul style="list-style-type: none"><li>– Ayuda al diagnóstico</li><li>– Análisis de imágenes biomédicas</li><li>– Procesamiento de señales fisiológicas</li></ul>
Ingeniería	<ul style="list-style-type: none"><li>– Organización de la producción</li><li>– Optimización de procesos</li><li>– Cálculo de estructuras</li><li>– Planificación y logística</li><li>– Diagnóstico de fallos</li><li>– Toma de decisiones</li></ul>
Biología	<ul style="list-style-type: none"><li>– Análisis de estructuras biológicas</li><li>– Genética médica y molecular</li></ul>
Robótica y automática	<ul style="list-style-type: none"><li>– Sistemas adaptativos de rehabilitación</li><li>– Interfaces cerebro-computadora</li><li>– Sistemas de visión artificial</li><li>– Sistemas de navegación automática</li></ul>

*Nota.* Adaptado de Inteligencia artificial avanzada (p.18), por R. Benítez, G. Escudero y S. Kanaan, 2014, Universitat Oberta de Catalunya (UOC).

De forma particular, Bostrom (2016) expone diferentes aplicaciones que se extienden del campo de visión computacional, como es el caso de robots de limpieza, rescate, quirúrgicos y más de un millón de robots industriales que ayudan en labores específicas de predicción y toma de decisiones en función a las fuentes visuales que

reciben, así como el reconocimiento óptico de caracteres de texto manuscritos y mecanografiados, utilizados rutinariamente en aplicaciones como la clasificación de correo y la digitalización de documentos antiguos.

En este sentido, Boston (2016) plantea que el reconocimiento facial ha mejorado tanto en los últimos años que ahora se emplea en los pasos fronterizos automatizados de Europa y Australia como sistema de seguridad y vigilancia. El departamento de los Estados Unidos cuenta con un sistema de reconocimiento facial con más de 75 millones de fotografías para la tramitación de visados. Los sistemas de vigilancia que usan IA son cada vez más sofisticados, además de tecnologías de análisis de datos para analizar voz, videos y textos, un buen número de los cuales se obtienen de los medios de comunicación electrónicos de todo el mundo y se almacenan en enormes bases de datos.

Así mismo, el autor plantea que los organismos militares y de inteligencia de los Estados Unidos han liderado la senda que ha llevado al despliegue a gran escala de robots desactivadores de bombas, drones de vigilancia y ataque y de otros vehículos no tripulados. Estos todavía dependen principalmente del control remoto por parte de operadores humanos, pero se está trabajando para superar sus capacidades autónomas (p. 15).

En este orden de ideas, García (2017), expresa que en la actualidad son muchas las universidades que investigan el campo de la IA, pero cada vez más empresas como Google, gobiernos y otras instituciones invierten grandes cantidades de dinero. Mucha culpa de todo esto la tiene internet, cuya gran cantidad de

información facilita el acceso a grandes cantidades de datos analizables y a su vez demanda de nuevas técnicas que permitan manejar tales cantidades de información.

De igual forma, el uso de clasificadores bayesianos (clasificador probabilístico fundamentado en el teorema de Bayes) y otras técnicas probabilísticas se muestran muy eficaces contra el problema de spam o correo basura. El mercado de consumo tampoco deja escapar el tren y actualmente los teléfonos inteligentes comienzan a hacer uso del reconocimiento facial como es el caso de los teléfonos Android o el reconocimiento del habla del que hace uso Apple en su iPhone con su sistema Siri; en especial la industria automovilística trabaja actualmente en coches autónomos que se conducen solos o que son capaces de prever situaciones complicadas y actuar en consecuencia. La aplicación militar, aunque la mayoría de las veces permanecen en secreto, son visibles, sobre todo en robótica, donde aviones no tripulados y robots que desactivan minas son nada más la punta del iceberg.

Prácticamente, todos los sistemas actuales en uso son del primer tipo: de nivel reducido, no obstante, muchos de ellos contienen componentes que también podrían desempeñar un papel en la inteligencia artificial general del futuro, o estar al servicio de su desarrollo, como los clasificadores bayesianos, sistemas expertos, de planificación y programación de logística automatizada, sistemas de reconocimiento de texto y habla, por mencionar algunos, los cuales de acuerdo al nivel de capacidad cognitivo o su capacidad de resolución de problemas de aplicación más generalizada, proponen soluciones efectivas a actividades cada vez más complejas gracias al gran volumen de información que cada día se hace más grande (pp. 7-9).

En este sentido, los aportes sugeridos por los autores sugieren desde su punto de vista las principales aplicaciones que tiene la IA, con especial énfasis en el desarrollo de la visión computacional, siendo el campo de la IA en el que se basa la presente investigación, resultando interesante el enfoque e incidencia indiscutible en áreas tan importantes como la medicina y la milicia en cuanto a estrategias para la toma de decisiones, seguridad y vigilancia; por medio de sistemas de monitoreo y análisis de imagen, capaces de describir e interpretar esa información y aprende de ella, con el propósito de brindar la mejor alternativa de solución.

Avances que apuntan hacia la mejora en la calidad de vida del ser humano, siendo este un aspecto fundamental que se busca obtener como resultado de la implementación del objetivo general de esta investigación, la cual propone desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad.

## **Robótica**

Gollapudi (2019) plantea que la robótica es una disciplina informática que se ocupa del diseño, la programación, la ingeniería y el desarrollo de robots o máquinas físicas que se construyen para ejecutar tareas que generalmente realizan los humanos. La adopción de la robótica originalmente estaba dirigida a trabajos que son peligrosos para los humanos, como soldar, remachar, minar, limpiar desechos tóxicos o desactivar bombas, entre otros, o aquellos que necesitan alta precisión o poca tolerancia a errores humanos, como cirugías largas en el campo de la medicina (p. 10).

En este orden de ideas, Wilkins (2019) considera la robótica como una rama exhaustiva de la ciencia y la ingeniería, la cual abarca diferentes campos de la

ingeniería, como la electrónica, la mecánica, la informática y mucho más. Por lo general, está relacionado con el uso de robots, como se inventan, como funcionan y el sistema informático que se utiliza para controlar el robot. Las tecnologías robóticas se emplean para crear máquinas que ocupen el lugar de los humanos al replicar las acciones humanas, los robots se pueden usar en muchas situaciones y propósitos.

El concepto de crear robots se remonta a la época clásica, sin embargo, la investigación y el potencial de crear el robot perfecto no se pusieron en práctica hasta principios del siglo XX. A lo largo de la historia, la gente ha asumido que los robots son capaces de imitar el comportamiento humano y manejar tareas similares a las de los humanos todos los días. Ahora, la robótica es un campo en rápido crecimiento gracias a los avances tecnológicos (p. 1).

Así mismo, Benítez, Escudero y Kanaan (2014), plantean que en robótica uno de los campos de investigación actual con mayor proyección es el del aprendizaje adaptativo, en el que un sistema robotizado explora diferentes configuraciones con el objetivo de realizar un movimiento complejo (caminar, agarrar un objeto, realizar una trayectoria, jugar al golf, etc.). El objetivo podría ser, por ejemplo, que un robot cuadrúpedo se levante y ande de forma autónoma, de forma que siga un proceso de exploración y aprendizaje similar al que lleva a cabo un recién nacido durante los primeros meses de vida (p. 10).

Es así y en especial para la presente investigación que Benítez, Escudero y Kanaan (2014), revelan cómo la visión artificial acompaña al proceso de modulación y ejecución de diferentes acciones en la robótica; así como comprender el espacio y calcular dimensiones que faciliten el desplazamiento y reacción ante las condiciones

del entorno forma parte de los objetivos que plantea esta investigación, que si bien se centra en un sistema de reconocimiento de imágenes para emitir alertas de seguridad, busca comprender cómo el uso de secuencias de imágenes, vistas desde cámaras de video o de datos multidimensionales, pueden generar reacciones en dispositivos robotizados de manera controlada y precisa, interpretando la información dada por el entorno que la rodea.

## **Redes neuronales**

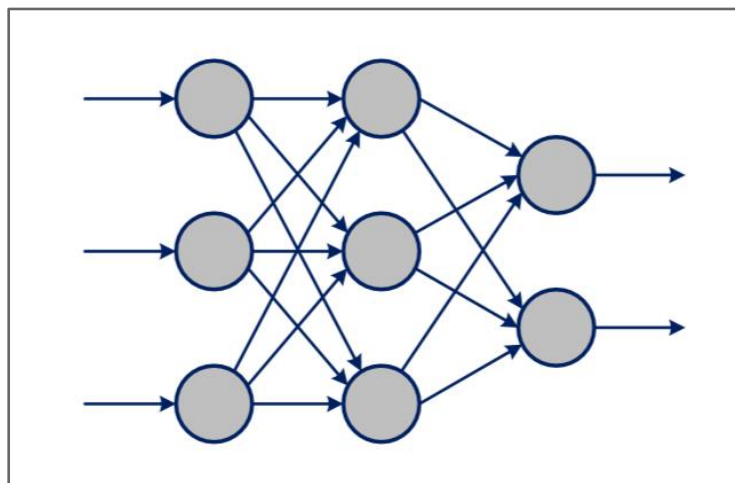
Berza (2018), expresa que, desde los albores de la inteligencia artificial, las redes neuronales se utilizaron como modelo para intentar dotar de inteligencia a las máquinas. Los conexionistas, como se les llamaba a los que empleaban el cerebro como modelo, proponían un enfoque alternativo a las técnicas basadas en modelos lógicos de los simbólicos. De ahí que las redes neuronales también recibiesen la denominación de técnicas sub simbólicas (no sin cierto desdén, según se pronuncie). Actualmente, se hace referencia a las redes neuronales artificiales con el término deep learning. Literalmente, aprendizaje profundo.

Tras distintos altibajos desde sus orígenes, las redes neuronales artificiales se han convertido en las técnicas de moda dentro del aprendizaje automático ¿Por qué? Porque hasta ahora, han sido las únicas capaces de alcanzar una eficacia comparable con la del hombre en distintos problemas complejos de reconocimiento de patrones, como el reconocimiento de voz (usado hoy por numerosos dispositivos móviles y gadgets particulares), la traducción de textos de un idioma a otro, o la identificación de objetos en imágenes, un problema típico de visión artificial.

Tal como lo plantea el autor, la idea común detrás de todas las técnicas basadas en redes neuronales artificiales es intentar expresar la solución de problemas complejos no como un algoritmo secuencial que resuelve un problema aplicando una secuencia de pasos concreta, sino como el resultado de combinar, en paralelo, las pequeñas contribuciones realizadas por un gran número de elementos simples de procesamiento que se hallan interconectados entre sí. Esos elementos simples de procesamiento son las neuronas que forman parte de una red neuronal artificial y las conexiones entre ellas son las que representan el “conocimiento” adquirido por la red (ver figura 1).

**Figura 1:**

*Una red neuronal artificial con múltiples capas de neuronas.*



Nota. Adaptado de Redes neuronales & Deep Learning (p. 127), por F. Berzal, 2018, Edición independiente.

La capa de entrada, a la izquierda, recibe un estímulo del exterior. Esta entrada se acaba traduciendo en una respuesta en la capa de salida, a la derecha. Entre las

capas de entrada y salida pueden existir capas intermedias que se denominan capas ocultas (pp. 125 - 127).

García (2017) expone que las redes neuronales artificiales o RNAs son un intento de emular la forma de trabajar del cerebro humano y aunque estamos lejos de alcanzar su misma capacidad, son un instrumento de gran potencia para gran cantidad de aplicaciones y un campo de investigación muy prometedor. Una neurona animal está compuesta por el núcleo, que no es más que una célula especializada, rodeada de millones de conexiones que la unen a otras neuronas. Estas conexiones se denominan sinapsis (p. 209).

En este sentido, Bostrom (2016) plantea que, aunque ya se conocían las redes neuronales desde finales de la década de los 50 del siglo XX, este campo tuvo un renacimiento después de la introducción del algoritmo de retropropagación, el cual posibilita entrenar redes neuronales de capas múltiples. Este tipo de redes de capas múltiples, que tenían una o más capas de neuronas "ocultas" intermedias entre las capas de entrada y salida, pueden aprender una gama mucho más amplia de funciones que sus predecesoras más simples; en comparación con los ordenadores cada vez más potentes que había disponibles, estas mejoras algorítmicas permitieron a los ingenieros construir redes neuronales lo suficientemente buenas como para ser útiles en sentido práctico en muchas aplicaciones (p. 8).

Tal como lo plantea Berza (2018), las redes neuronales artificiales son capaces de analizar con una eficiencia comparable a la del hombre problemas complejos relacionados al reconocimiento de patrones de voz, escritura e imágenes, siendo de especial interés para la presente investigación esta última, ya que la red neuronal

artificial toma el conocimiento adquirido durante su entrenamiento, estimulando su crecimiento mediante la entrada y salida de datos, por lo que orientado al objetivo de esta investigación, a través del procesamiento de un set de imágenes como base inicial, es posible el reconocimiento de imágenes por medio del diseño e implementación de una red neuronal artificial, gracias a su aprendizaje adaptativo, preciso y previsible ante los cambios y las diferentes características que pueden presentar las imágenes estudiadas, siendo esta una premisa importante al deep learning.

## **Redes Bayesianas**

Según Berzal (2018), el teorema de Bayes recibe su nombre del reverendo presbiteriano inglés Thomas Bayes, que en realidad nunca publicó en vida el trabajo por el que pasaría a la historia. Fue el predicador unitario galés Richard Price el que recopiló y editó los manuscritos inéditos de Bayes, publicados de forma póstuma en 1763, un par de años después de la muerte de Bayes. Price defendía que el teorema de Bayes ayudaba a demostrar la existencia de Dios usando un argumento teleológico: la existencia de Dios (deidad o creador inteligente) a partir de la evidencia percibida en forma de orden y regularidades en el mundo natural; esto es, Dios como causa última de un diseño deliberado. De modo aparentemente independiente, el matemático francés Pierre-Simon, marqués de Laplace, derivó los mismos resultados en 1774.

El conocido teorema de Bayes es la piedra angular de las técnicas probabilísticas, de ahí que los que trabajan en este tipo de técnicas reciben el apelativo de bayesianos:

$$p(y \vee x)p(x) = p(x \vee y)p(y) \quad (1)$$

Reordenando los términos, obtenemos la forma en la que resulta más útil en aprendizaje automático:

$$p(y \vee x) = \frac{p(x \vee y)p(y)}{p(x)} \quad (2)$$

Como lo plantea el autor con esta expresión, podemos predecir la probabilidad  $p(y|x)$  de una clase y dado un caso  $x$  disponiendo únicamente de información asociada a la probabilidad de los casos  $p(x)$ , la probabilidad de las clases  $p(y)$  y la probabilidad  $p(x|y)$  de los casos dada la clase. Todas esas probabilidades las podemos estimar con facilidad a partir de los datos de nuestro conjunto de entrenamiento (p. 111).

Complementando lo anterior, Russell y Norvig (2009), plantea lo siguiente:

El caso más general para variables multivaluadas puede escribirse con la notación  $P$ , tal como se indica en la fórmula anterior. Donde de nuevo se interpreta como la representación de un conjunto de ecuaciones y cada relación con valores específicos de las variables. También tendremos ocasión de usar una versión más general, condicionada a alguna evidencia base  $e$ .

$$P(Y \vee X, e) = \frac{P(X \vee Y, e)P(Y \vee e)}{P(X \vee e)} \quad (3)$$

A primera vista, la regla de Bayes no parece muy útil. Requiere tres términos (una probabilidad condicional y dos probabilidades incondicionales) solo para calcular una probabilidad. La regla de Bayes es útil en la práctica porque hay muchos casos donde disponemos de buenas estimaciones probabilistas para estos tres números y necesitamos calcular el cuarto. En una tarea como el diagnóstico médico, con

frecuencia tenemos probabilidades condicionales de relaciones causales y necesitamos deducir un diagnóstico.

Un médico sabe que la enfermedad de meningitis causa al paciente que el cuello se agarrote, digamos, el 50 por ciento de las veces. El médico también conoce algunos hechos incondicionados: la probabilidad a priori de que un paciente tenga meningitis es  $1/50.000$  y la probabilidad a priori de que cualquier paciente tenga un cuello agarrotado es  $1/20$ . Si tomamos  $s$  como la proposición de que el paciente tiene un cuello agarrotado y  $m$  como la proposición de que el paciente tiene meningitis, tenemos:

$$P(s \vee m) = 0,5 \tag{4}$$

$$P(m) = 1/50.000$$

$$P(s) = 1/20$$

$$P(m \vee s) = P(s \vee m)P(m)(s) = \frac{0,5 \times 1/50.000}{1/20} = 0,0002$$

Es decir, esperamos que solamente un paciente de entre 5.000 con un cuello agarrotado tenga meningitis. Nótese que, aun pensando en que una meningitis apunta a un cuello agarrotado con bastante contundencia (con probabilidad 0,5), la probabilidad de meningitis en el paciente se queda pequeña. Esto es porque la probabilidad a priori de un cuello agarrotado es mucho más alta que la de meningitis (p. 546).

Así mismo, Bostrom (2016), plantea que el trabajo realizado sobre el último par de décadas en los modelos de gráficos probabilísticos, como las redes bayesianas, proporcionan una forma concisa de representar las relaciones de independencia probabilísticas y condicionales que comparten algún ámbito particular. Un agente

bayesiano óptimo comienza con una "distribución de probabilidad a priori", una función que asigna probabilidades a cada "mundo posible", esta a priori incorpora un sesgo inductivo de tal modo que a los mundos más simples se les asigna probabilidades más altas. Y a priori también incorpora cualquier conocimiento subyacente que los programadores desean dar al agente.

A medida que el agente recibe nueva información de sus sensores, actualiza su distribución de probabilidad condicionando la distribución de la nueva información de acuerdo al teorema de Bayes, cuyo resultado es una "distribución de probabilidad posteriori" y a medida que el agente hace observaciones, su masa de probabilidad se concentra de este modo en el menguante conjunto de mundos posibles que siguen siendo consistentes con las evidencias y de entre otros mundos posibles, otros más siempre tienen más probabilidades (p. 546).

Resulta importante para la presente investigación, comprender cómo las redes bayesianas permiten cuantificar e identificar las distintas probabilidades de una solución de acuerdo a la entrada visual, en este sentido, en especial Berzal (2018), plantea el teorema de Bayes como la piedra angular de las técnicas probabilísticas, exponiendo un ejemplo simple sobre su aplicación, dicha expresión matemática permitirá estimar con mayor facilidad a partir de los datos del conjunto de entrenamiento, que a través del nivel de incertidumbre permite que el modelo maneje mejor los errores, con base en la información que conoce y que en su aplicación más común como lo es la robótica móvil puede mejorar tareas como la localización, segmentación, reconocimiento y estimación de profundidad, puntos a tratar dentro de los objetivos de esta investigación.

## Machine Learning

Según Guido y Müller (2017), machine learning se trata de extraer conocimiento de los datos. Es un campo de investigación en la intersección de estadísticas, inteligencia artificial y ciencias de la computación, conocido también como análisis predictivo o aprendizaje estadístico. La aplicación de métodos de aprendizaje automático se ha vuelto omnipresente en la vida cotidiana en los últimos años. Desde recomendaciones automáticas sobre qué películas mirar, qué alimentos pedir o qué productos comprar, hasta la radio en línea personalizada, así como reconocimiento de amigos en fotos, por lo que muchos sitios web y dispositivos modernos tienen en su núcleo algoritmos de machine learning. Cuando miras un sitio web complejo como Facebook, Amazon o Netflix, es muy probable que cada parte del sitio contenga múltiples modelos de machine learning (p. 1).

En este sentido, Gollapudi (2019), plantea que el machine learning es una forma de construir inteligencia en una máquina para que pueda aprender con el tiempo y mejorar utilizando su propia experiencia. Se trata de un mecanismo de búsqueda de patrones que filtra los detalles relevantes, cuyo objetivo del algoritmo de aprendizaje está en producir un resultado en modo de una regla que sea precisa al máximo (p. 11).

Sin embargo, Russell (2018) expresa que el machine learning es la práctica de programación de computadoras para aprender de los datos. En el machine learning, los datos son conocidos como conjuntos de capacitación o ejemplos.

¿Cuándo debemos usar machine learning?:

- Cuando se tiene un problema que requiere de largas listas de reglas para encontrar la solución.
- Problemas muy complejos para los que no hay solución con un enfoque tradicional.
- Ambientes no estables: Softwares con Machine Learning se pueden adaptar a nuevos datos.

Es así, que el machine learning, también conocido como análisis predictivo o aprendizaje estadístico, resulta uno de los modelos con más aplicaciones, de las cuales muchas son imperceptibles en la actualidad, las grandes plataformas de comunicación como YouTube, Facebook, Instagram y Twitter tienen incorporado múltiples modelos de machine learning que los ayudan a posicionar contenido, ofrecer productos y recomendaciones orientados a los gustos e intereses de sus usuarios de forma automática (p. 3).

Para la presente investigación y de acuerdo con lo planteado por Gollapudi (2019), el machine learning aprende con el tiempo, mejorando sus resultados en función de la experiencia, que de acuerdo a su capacidad de autoprogramarse, permite obtener una respuesta lógica con base en la información suministrada, siendo ampliamente ventajosa en implementaciones relacionadas con visión artificial, ya que pueden reconocer patrones, aprender de ellos y realizar distintas acciones, así como aplicar diferentes mediciones y variables para identificar un objeto dentro de una imagen, analizar su estado y decidir qué hacer con él.

## Deep Learning

Según Gollapudi (2019), deep learning es un área de aprendizaje automático que se centra en unificar el aprendizaje automático con inteligencia artificial. Para un requisito de detección de rostros, un algoritmo de deep learning registra o aprende características como la longitud de la nariz, la distancia entre los ojos, el color de los globos oculares, etc. Estos datos se utilizan para abordar una clasificación o un problema de predicción y son evidentemente diferentes del algoritmo tradicional de aprendizaje superficial (p. 12).

En este sentido y de acuerdo con Berzal (2018), las técnicas de deep learning son capaces de descubrir características a partir de los datos, como las técnicas de aprendizaje de representaciones, pero también son capaces de crear nuevas características a partir de otras características. De esa forma, son capaces no solo de identificar características que nos ayuden a discriminar mejor en función de la situación en la que nos encontremos, sino de aumentar el nivel de abstracción al que se trabaja mediante la creación de jerarquías con varios niveles de características, en las que nuevas características se descubren de forma automática a partir de las características del nivel anterior.

En otras palabras, las técnicas de deep learning realizan ingeniería de características de manera completamente automática. Las redes neuronales que se emplean están formadas por múltiples capas de neuronas, como el córtex cerebral, el cerebelo o la retina humana. Cada capa permite construir nuevas características a partir de las características identificadas por las capas anteriores, lo que resulta

especialmente útil en el análisis de datos no estructurados como señales de audio o imágenes. Es la capacidad de formar modelos a distintos niveles de abstracción lo que ha permitido construir sistemas automáticos de reconocimiento de voz o de identificación de objetos en imágenes con un rendimiento similar al conseguido por el ser humano (p. 144).

Así mismo, Eckroth (2018) plantea que el deep learning se refiere a varios métodos que pueden usarse en una aplicación particular, los cuales incluyen capas convolucionales y de agrupación. Varios factores han contribuido al crecimiento dramático del deep learning en los últimos cinco años. Grandes conjuntos de datos públicos que contienen millones de imágenes y voces etiquetadas que cubren mil categorías están disponibles. Dichos conjuntos de datos han satisfecho el requisito básico para el deep learning de muchos datos de capacitación. Las GPU (unidad de procesamiento gráfico) han hecho la transición al deep learning y a los clústeres, mientras que también se centran en los juegos, ayudando a hacer posible el aprendizaje profundo a gran escala (p. 105).

De acuerdo a lo planteado, tal como lo plantea Berzal (2018) los algoritmos de deep learning realizan ingeniería de características de forma completamente automática, lo que significa que son capaces de crear nuevas características a partir de otras características, sin necesidad de reprogramar los algoritmos principales, cualidades a explorar en la presente investigación, ya que potencia el sistema de visión tradicional, aportando mejoras en la lectura de una imagen a pesar del ruido e imperfectos que esta tenga, conceptualizando y generalizando la apariencia de la imagen basada en sus características distintivas, reconociendo correctamente los

patrones gracias a las capas de algoritmos integrados en su red neuronal que le permite sacar sus propias conclusiones.

## **Procesamiento del Lenguaje Natural (PLN)**

Según García (2017) el procesamiento del lenguaje natural o NLP (Natural Language Processing) es una rama de la Inteligencia Artificial que se ocupa de las capacidades de comunicación de los ordenadores con los humanos utilizando su propio lenguaje. Es un área cuyas aplicaciones son múltiples y variadas, como la traducción automática o el reconocimiento y comprensión del lenguaje humano entre otros (p. 3).

En este sentido, Gollapudi (2019), plantea que el procesamiento del lenguaje natural (PLN) se refiere a un área de especialización en informática que se ocupa de analizar y derivar información útil o significativa del lenguaje natural o del lenguaje humano. En un nivel alto, esto requiere el empleo de técnicas formales como la tokenización, la extracción de relaciones en el contexto de un caso de negocios específico, la clasificación de palabras y la detección de oraciones.

Para un lenguaje, la sintaxis se refiere a reglas básicas que sigue el lenguaje y la semántica se refiere a su significado. La complejidad proviene del hecho de que el significado del texto puede ser ambiguo y puede cambiar con el contexto. Por ejemplo, la saturación de palabras podría tener diferentes definiciones cuando se usa con colores o cuando se usa en el contexto de comportamientos humanos.

PLN se utiliza en una amplia variedad de disciplinas para resolver una variedad de problemas. A continuación, se incluye una breve lista de aplicaciones:

- Buscadores, se refiere a la identificación de elementos específicos de texto dentro de un contexto de contenido más amplio.
- La traducción automática se trata de traducir texto de un idioma natural a otro y resumir textos más largos en documentos, blogs, etc.
- El reconocimiento de entidad con nombre (NER), se refiere a extraer los nombres de ubicaciones, personas y cosas del texto.
- La agrupación de información consiste en clasificar el texto en función de su contenido y contexto.
- El análisis de sentimientos se emplea generalmente para percibir y proporcionar ayuda o comentarios automatizados sobre cómo un producto como un libro o una película está funcionando en el mercado.
- La respuesta a consultas o ayuda se usa en medicina o servicios minoristas, por ejemplo, en bots de chat.
- El reconocimiento de voz ayuda a analizar y comprender automáticamente el contexto en una conversación con humanos (p. 7).

Sin embargo, Bostrom (2016), expresa que el sentido común y la comprensión del lenguaje natural también han resultado ser muy difíciles. En la actualidad, se suele pensar que alcanzar un rendimiento completamente humano en estas tareas es un problema de la “IA completa”, queriendo con esto decir que solucionar estos problemas son esencialmente equivalentes a la dificultad de crear máquinas de inteligencia general de nivel humano. En otras palabras, si alguien llega a tener éxito originando una IA que pudiera entender el lenguaje natural tal como lo hace un adulto humano, también habría tenido éxito, con toda la probabilidad en la tarea de producir una IA que

podiese hacer todo lo que una inteligencia humana puede hacer o estaría muy cerca de una capacidad general de esa clase (p.14).

Debido a esto, tomando el planteamiento de Gollapudi (2019), el lenguaje natural (PLN) es el área que se ocupa de analizar y derivar información útil o significativa del lenguaje natural o del lenguaje humano, si bien es una rama de la IA que expone un extremo diferente al de la visión computacional, la cual a pesar de las limitaciones de las versiones preliminares y su reducido campo de aplicación, resulta interesante para la presente investigación comprender cómo la gramática puede definir acciones examinando imágenes de video, dando sentido a las escenas completas y similares, donde es posible ahorrar cálculos cuando no se encaje con las hipótesis.

## **Sistemas Expertos**

Benítez, Escudero y Kanaan (2014) plantean que, dentro del campo de la IA, una de las ramas con mayor proyección son los denominados sistemas expertos, en los que el objetivo es diseñar un sistema que permita analizar un conjunto de datos y realizar tareas típicamente asociadas a la figura de un profesional experto como el diagnóstico, la detección de fallos, la planificación o la toma de decisiones. Los datos con los que trabaja el sistema experto pueden ser de naturaleza muy diversa (p.10).

Según Gollapudi (2019) los sistemas expertos (ES) son uno de los dominios de investigación más importantes de IA que se mencionaron por primera vez en la Universidad de Stanford. Estos sistemas se centran en resolver problemas complejos en un dominio particular a un nivel de inteligencia o experiencia humana ejemplar. Los sistemas expertos son altamente receptivos, confiables, precisos y de alto rendimiento.

Si bien no pueden reemplazar a un ser humano en lo que respecta a la toma de decisiones, se utilizan como asesores de los humanos y pueden ayudar en el diagnóstico, explicación, predicción, justificación y razonamiento. Cualquier sistema experto incluye tres componentes principales: una base de conocimiento, un motor de inferencia y una interfaz de usuario.

Aunado a lo anterior, el autor establece que los sistemas expertos son usados en muchos dominios, algunos ejemplos de uso son la detección de fraudes (la identificación de transacciones sospechosas y el comercio de bolsa en el dominio financiero), el diagnóstico de dolencia crítica y la deducción de la causa raíz de una dolencia en el dominio médico y la predicción del comportamiento potencial de un sistema mediante el monitoreo de su estado actual contra los patrones derivados de informes de monitoreo anteriores (p. 12).

Así mismo, García (2017) expresa que desde los inicios de la Inteligencia Artificial se ha tratado de conseguir que los ordenadores razonen de forma similar a como lo hacen los humanos. Los primeros sistemas que trataban de conseguirlo se denominan Sistemas Basados en Conocimiento (SBC). Los SBC no buscan ser capaces de razonar en cualquier circunstancia, sino que se circunscriben a un problema concreto. Por ejemplo, podríamos pensar en un SBC capaz de diagnosticar enfermedades respiratorias. Es lo que se denomina dominio de aplicación del SBC (p. 172).

En este sentido y para la presente investigación resulta interesante cómo el autor Gollapudi (2019) plantea cómo los sistemas expertos se centran principalmente en resolver problemas complejos en un dominio particular a un nivel de inteligencia

humana, como rama de la IA resultan muy útiles en la toma de decisiones, sistemas altamente receptivos, confiables, precisos y de alto rendimiento, criterios de aceptación que se persiguen con la implementación del objetivo general que propone esta investigación.

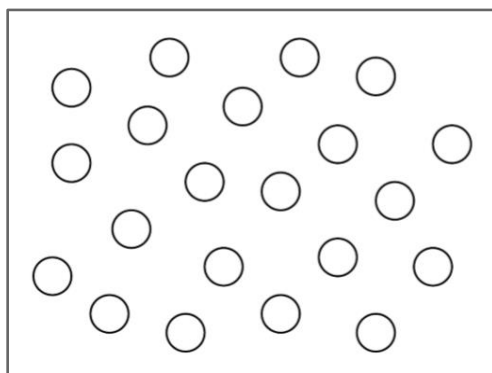
## Datasets

Rosebrock (2017), plantea que al realizar el aprendizaje automático (machine learning) y el aprendizaje profundo (deep learning), tenemos un conjunto de datos del que estamos tratando de extraer conocimiento. Cada ejemplo o elemento en el conjunto de datos (ya sean imagen, texto, audio, etc.) es un punto de datos. Por lo tanto, un conjunto de datos es una colección de puntos de datos (ver figura 2).

El objetivo de aplicar algoritmos de machine learning y deep learning es descubrir patrones subyacentes en el conjunto de datos, lo que nos permite clasificar correctamente los puntos de datos que el algoritmo aún no ha encontrado.

### Figura 2:

*Datasets (rectángulo exterior) es una colección de puntos de datos (círculos).*



*Nota.* Adaptado de Deep Learning for Computer Vision with Python (p. 41), por A. Rosebrock, 2017, PyImageSearch.

De acuerdo con lo que se observa en la figura 2, el autor expone lo siguiente:

- En el contexto de la clasificación de imágenes, el conjunto de datos es una colección de imágenes.
- Cada imagen es, por lo tanto, un punto de datos (p. 40-41).

Adicionalmente, Rosebrock (2017), establece que cuando se trabaja con conjuntos de datos de imágenes, primero debemos considerar el tamaño total del conjunto de datos en términos de bytes. ¿Es nuestro conjunto de datos lo suficientemente grande como para caber en la RAM disponible en nuestra máquina? ¿Podemos cargar el conjunto de datos como si cargara una matriz? ¿O el conjunto de datos es tan grande que excede la memoria de nuestra máquina, lo que requiere fragmentar el conjunto de datos en segmentos y solo cargar algunas partes a la vez?

Los datasets dentro de Starter Bundle son lo suficientemente pequeños como para que podamos cargarlos en la memoria principal sin tener que preocuparnos por la administración de la memoria; sin embargo, los conjuntos de datos mucho más grandes dentro del paquete PractitionerBundle y ImageNetBundle requerirán que se desarrollen algunos métodos para manejar de manera eficiente las imágenes de carga de una manera que podamos entrenar a un clasificador de imágenes (sin quedarse sin memoria).

Dicho esto, el autor plantea que siempre se debe reconocer el tamaño de datos antes de comenzar a trabajar con algoritmos de clasificación de imágenes, por lo que tomarse el tiempo para organizar, preprocesar y cargar los datasets es un aspecto crítico de la construcción de un clasificador de imágenes (p. 69).

Así mismo, Eckroth (2018) expresa que varios factores han contribuido al crecimiento dramático del aprendizaje profundo en los últimos cinco años. Grandes datasets públicos, como ImageNet, que contiene millones de imágenes etiquetadas que cubren mil categorías y el Proyecto de Voz Común de Mozilla, que contienen muestras de voz, ahora están disponibles para todos, satisfaciendo así el requisito básico para el aprendizaje profundo (p. 105).

En este sentido, según Dadhich (2018) en la visión por computadora, los datasets juegan un papel clave en el desarrollo de aplicaciones eficientes. Además, ahora, con la disponibilidad de grandes conjuntos de datos de código abierto, se ha vuelto mucho más fácil crear modelos con el mejor rendimiento para las tareas de visión por computadora (p. 6).

Desde un enfoque práctico para la investigación, Rosebrock (2017) plantea que cuando se trabaja con conjuntos de datos de imágenes, primero se debe considerar el tamaño total del conjunto de datos en términos de bytes, esto nos brinda pautas en cuanto a hardware y software importantes para la implementación del objetivo general que propone la presente investigación, por lo que reconocer el tamaño de los datos antes de comenzar con el entrenamiento de los algoritmos, en especial con algoritmos de clasificación de imágenes puede ser un aspecto crítico en su construcción, por lo que hay que tomar el tiempo para organizar, preprocesar y cargar los datasets (p. 69).

## Visión Computacional

Según Dadhich (2018) la visión por computadora es uno de los subcampos más ampliamente estudiados de la informática. Tiene varias aplicaciones importantes, como la detección de rostros, la búsqueda de imágenes y la conversión artística de imágenes. Con la popularidad de los métodos de aprendizaje profundo, muchas aplicaciones recientes de la visión por computadora se encuentran en automóviles autónomos, robótica, medicina, realidad virtual y realidad aumentada.

Tal progreso en varios dominios hace que esta maravilla no experta sea capaz de inferir esta información de las imágenes. La motivación radica en la percepción humana y cómo podemos realizar análisis complejos del entorno que nos rodea. Podemos estimar la cercanía, la estructura y la forma de los objetos, así como las texturas de una superficie. Incluso bajo diferentes luces, podemos identificar objetos y reconocer algo si lo hemos visto antes. Por lo que para comenzar la discusión sobre la visión por computadora se observa la figura 3.

### Figura 3:

*Caso práctico de visión computacional general.*



*Nota.* Adaptado de Practical computer vision (p. 6), por A. Dadhich, 2018, Packt Publishing Limited, Reino Unido.

Incluso si nunca hemos hecho esta actividad, podemos decir claramente que la imagen es de personas esquiando en las montañas nevadas en un día nublado. Esta información que percibimos es bastante compleja y puede subdividirse en inferencias más básicas para un sistema de visión por computadora.

La observación más básica que podemos obtener de una imagen es de las cosas u objetos que contiene. En la imagen anterior, las diversas cosas que podemos ver son árboles, montañas, nieve, cielo, personas, etc. La extracción de esta información a menudo se conoce como clasificación de imágenes, donde nos gustaría etiquetar una imagen con un conjunto predefinido de categorías. En este caso, las etiquetas son las cosas que vemos en la imagen (p. 6).

Así mismo, Davies (2018) plantea que la visión por computadora ha mostrado un progreso impresionante en su corta historia. Esto se debe en parte a la tecnología: las computadoras son mucho más rápidas y la memoria ahora es mucho más barata que en los primeros años de investigación. Ha habido muchos logros y muchos desarrollos.

Podemos mirar hacia un futuro en el que los sistemas automatizados de visión por computadora nos facilitarán la vida y los enriquecerán también. Ya hay muchas aplicaciones de visión computacional en la industria alimentaria y los automóviles robóticos que estarán con nosotros muy pronto. Luego, hay avances continuos en el análisis de imágenes médicas, donde las técnicas de visión por computadora se pueden utilizar para ayudar en el diagnóstico y la terapia por medios automatizados. Incluso acceder a un teléfono móvil es considerablemente más conveniente cuando se usa una huella digital y el acceso por reconocimiento facial continúa mejorando. Todo

esto se debe a los avances en las computadoras, la visión por computadora y la inteligencia artificial aplicada.

Adicionalmente, el autor plantea que la visión computacional se las arregla para cubrir muchos aspectos de la tecnología, desde la visión humana hasta el aprendizaje automático, que requieren hardware electrónico, implementaciones informáticas y una gran cantidad de software. Sin duda, es el mejor camino, ya que muchos de sus investigadores continúan mejorando, refinando y desarrollando nuevas técnicas. Un cambio importante aquí es la inclusión de Deep Learning.

De hecho, el deep learning ha sido un cambio crucial en el campo de la visión por computadora y el reconocimiento de patrones. Una implicación del aumento en el poder de cómputo y la reducción del costo de la memoria es lo que ha permitido que las técnicas puedan volverse considerablemente más complejas. No se puede ignorar el desempeño del deep learning y las redes neuronales convolucionales: solo hay que leer detenidamente el programa de las principales conferencias internacionales para percibir su efecto revolucionario en la dirección de la investigación (p. xxiii).

Por su parte, Gollapudi (2019) plantea cómo la tecnología de visión por computadora implementa técnicas de deep learning y en algunos casos también emplea técnicas de PNL como una progresión natural de los pasos para analizar el texto extraído de las imágenes. Con todos los avances del deep learning, las funciones de construcción como la clasificación de imágenes, la detección de objetos, el seguimiento y la manipulación de imágenes se han vuelto más simples y precisas, lo que lleva a explorar aplicaciones autónomas más complejas, como autos sin conductor, humanoides y drones.

Con el deep learning, ahora podemos manipular imágenes, como superponer las características de Tom Cruise en otra cara o convertir una imagen en un modo de boceto o en un modo de pintura de acuarela. Podemos eliminar el ruido de fondo de una imagen y resaltar el sujeto enfocado o tomar una foto estable incluso con las manos más temblorosas. Podemos estimar la cercanía, estructura, forma de los objetos y también podemos estimar las texturas de una superficie. Con diferentes luces o exposición de la cámara, podemos identificar objetos y reconocer un objeto que hemos visto antes.

Así mismo, el autor expresa que, en visión por computadora, al decir que estamos permitiendo que las computadoras "vean", nos referimos a permitir que máquinas y dispositivos procesen datos visuales digitales, que pueden incluir imágenes tomadas con cámaras tradicionales, una representación gráfica de una ubicación, un video, un mapa de intensidad de calor de cualquier dato, y más allá. Como se puede apreciar, las aplicaciones de visión por computadora se están volviendo omnipresentes en nuestra vida cotidiana.

Se puede concluir que parte de esta evolución viene de la mano del desarrollo de computadoras mucho más potentes, hardware y software más rápido y accesible, lo que ha dado lugar a grandes avances en materia de medicina, industria automotriz, vigilancia y seguridad, convirtiéndose así en un campo emocionante con alternativas mucho más complejas por explorar y mejorar (p. 15).

En ese sentido, de acuerdo con el objetivo general de la presente investigación, se busca desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad con visión computacional, que tal como lo plantean los autores Davies (2018)

y Gollapudi (2019) respectivamente, permita que la computadora vea y procese los datos visuales, que junto con los avances en deep learning, funciones como clasificación, detección, seguimiento y la manipulación de imágenes resulta más simple y aún más importante preciso, ya que el dar una respuesta óptima en tiempo y precisión cuando se detecte algún objeto desconocido que pueda significar un factor de riesgo o peligro dentro de un área determinada forma parte fundamental de lo que propone esta investigación.

## **Clasificación**

Según Gollapudi (2019), la clasificación de imágenes es la tarea de etiquetar toda la imagen con un objeto o concepto con confianza. Las aplicaciones incluyen identificar el género dada una imagen de la cara de una persona, identificar el tipo de mascota, etiquetar fotos, etc. (p. 53).

Así mismo, Dadhich (2018) expresa que la extracción de esta información conocida como clasificación de imágenes, etiqueta una imagen con un conjunto predefinido de categorías. En este caso, las etiquetas son las cosas que vemos en la imagen (p. 13).

Una observación más amplia que podemos obtener de la imagen es el paisaje, podemos decir que la imagen consiste en nieve, montaña y cielo, como se muestra en la figura 4.

#### Figura 4:

*Caso práctico de clasificación y etiquetado.*



Nota. Adaptado de Practical computer vision (p. 14), por A. Dadhich, 2018, Packt Publishing Limited, Reino Unido.

Complementado lo anterior, Rosebrock (2017), expresa que la clasificación de imágenes, en esencia, es la tarea de asignar una etiqueta a una imagen de un conjunto predefinido de categorías. Prácticamente, esto significa que la tarea es analizar una imagen de entrada y devolver una etiqueta que clasifique la imagen. La etiqueta siempre proviene de un conjunto predefinido de categorías posibles. Por ejemplo, supongamos que nuestro conjunto de categorías posibles incluye: categorías = {gato, perro, panda}

Luego presentamos la siguiente imagen (figura 5) a nuestro sistema de clasificación:

## Figura 5:

*Imagen de entrada.*



*Nota.* El objetivo de un sistema de clasificación de imágenes es tomar una imagen de entrada y asignar una etiqueta basada en un conjunto predefinido de categorías. Tomado de Deep Learning for Computer Vision with Python (p. 39), por A. Rosebrock, 2017, PyImageSearch.

Lo que plantea el autor como objetivo aquí es tomar esta imagen de entrada y asignarle una etiqueta del conjunto de categorías, en este caso, perro. El sistema de clasificación también podría asignar múltiples etiquetas a la imagen a través de probabilidades, como perro: 95%; gato: 4%; panda: 1%. Más formalmente, dada la imagen de entrada de  $W \times H$  píxeles con tres canales, Rojo, Verde y Azul, respectivamente, la meta se ubica en  $W \times H \times 3 = N$  píxeles de imagen para clasificar correctamente el contenido de la imagen.

En este sentido, “la clasificación y comprensión de imágenes son actualmente (y seguirán siendo) el subcampo más popular de la visión por computadora durante los próximos años. En el futuro, veremos compañías como Google, Microsoft, Baidu y otras que rápidamente adquirirán empresas exitosas en materia de comprensión de imagen.

Se dispondrá de más aplicaciones de consumo en nuestros teléfonos inteligentes que pueden comprender e interpretar el contenido de una imagen. Incluso las guerras probablemente se librarán utilizando aviones no tripulados que se guían automáticamente mediante algoritmos de visión por computadora” (p. 39).

Tal como lo plantea Rosebrock (2017), la clasificación de imágenes es uno de los subcampos con mayor expansión en la actualidad en el área de visión computacional, por lo que entender cómo etiquetar una imagen con un conjunto definido de categorías posibles ya establecidas, es fundamental para la presente investigación, ya que como parte de la extracción de la información que presenta la imagen, se busca comprender e interpretar lo que sucede dentro de la misma, por lo que reconocer la técnica más apropiada que permita la selección de atributos para la clasificación que sean invariantes a factores como posición e iluminación en escenas dinámicas generadas por videos es uno de los retos a lo que se expone la presente investigación.

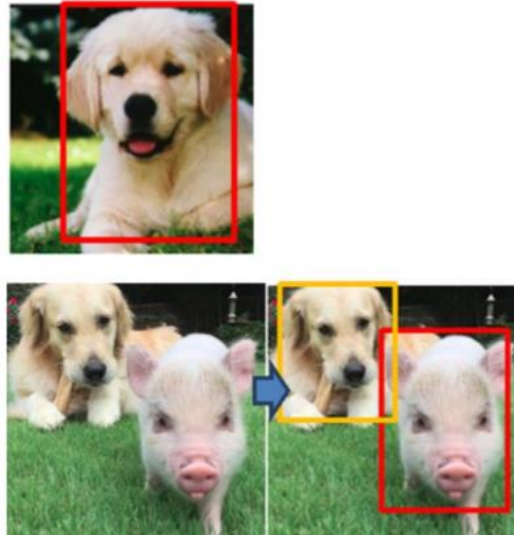
## **Detección y Reconocimiento**

Según Gollapudi (2019), la detección o localización es una actividad que consiste en encontrar un objeto y marcarlo o encuadrarlo. Esto tiene muchas aplicaciones en el mundo real, especialmente en la industria automotriz, donde los autos autónomos detectan objetos a través de los sensores de sus cámaras.

En este sentido, el autor a través de la figura 6, expone que la primera imagen con el recuadro rojo muestra la detección de objetos y la segunda imagen ubicada en la parte inferior muestra la localización (p. 54).

## Figura 6:

*Detección y localización de objetos.*



*Nota.* Adaptado de Learn Computer Vision Using OpenCV. With Deep Learning CNNs and RNNs (p. 54), por S. Gollapudi, 2019, Hyderabad, Telangana.

En este sentido, Dadhich (2018) propone varios métodos para abordar los problemas de velocidad y precisión en la detección de objetos. En general, se dividen en dos categorías principales:

- *Detectores de dos etapas:* aquí, el proceso general se divide en dos pasos principales, de ahí el nombre de detectores de dos etapas. El más popular entre estos es Faster R-CNN (arquitectura para la generación de regiones de interés).
- *Detectores de una etapa:* si bien los detectores de dos etapas aumentan la precisión para la detección, aún eran difíciles de entrenar y eran más lentos para varias operaciones en tiempo real. Los detectores de una etapa corrigieron estos problemas creando una red en una arquitectura única que predice más rápido. Uno de los modelos más populares de este estilo es Single Shot Multibox Detector (SSD: detector para procesamiento en tiempo real) (p. 105).

En este orden de ideas, Alegre, Gonzalo y Escalera (2016) plantean que la tarea de reconocimiento implica necesariamente un proceso de aprendizaje a partir de un conjunto de objetos (datasets), cuyos fundamentos teóricos y prácticos se establecieron en los años sesenta y setenta, con excelentes libros (Duda y Hart, 1973; Fukunaga, 1972), que dieron forma a este campo de investigación. Hay dos grandes tipos de problemas de reconocimiento de patrones: supervisado y no supervisado. En el aprendizaje supervisado se dispone de un conjunto de datos junto con sus etiquetas, indicando a qué clase pertenecen, para cada dato o ejemplo.

El objetivo que se persigue en este caso es conseguir un modelo (clasificador) que pueda etiquetar automáticamente nuevos datos que no se hayan empleado en el ajuste del modelo de clasificación. Este proceso es conocido como entrenamiento del clasificador o creación del modelo. Si las etiquetas no tienen un valor discreto, sino que toman valores continuos, hablaremos, entonces, de regresión. El reconocimiento óptico de caracteres sería un ejemplo de clasificación supervisada, mientras que la estimación de la edad de una persona a partir del rostro constituye un problema de regresión.

En las técnicas de aprendizaje no supervisado solamente se dispone de los datos sin las etiquetas de clase. Tratan de modelar los datos en sí mismos, lo que se traduce normalmente en descubrir grupos (clustering), es decir, determinar si hay ejemplos similares en el conjunto de datos y qué características hacen que estos datos sean similares dentro del grupo y diferentes del resto (p. 160).

De acuerdo con lo planteado anteriormente, se entiende que el área de detección dentro del modelo de visión computacional, busca reconocer la presencia de objetos dentro de una imagen de acuerdo a la apariencia que presenta, respetando los

criterios establecidos que faciliten el reconocimiento de la clase de clasificación a la que pertenecen, en este sentido y como soporte teórico-práctico para la presente investigación, resulta interesante lo expuesto por los autores Alegre, Gonzalo y Escalera (2016), los cuales plantean que la tarea de reconocimiento implica necesariamente un proceso de aprendizaje a partir de un conjunto de objetos (datasets), que en consecuencia permita un reconocimiento óptico de acuerdo a las características del objeto como parte de una clasificación supervisada.

## **Segmentación**

Sucar y Gómez (2011) plantean que separar la imagen en unidades significativas es un paso importante en visión computacional para llegar al reconocimiento de objetos. Este proceso se conoce como segmentación y es una forma de dividir la imagen mediante la determinación de los bordes. El objetivo dual de este problema es determinar las regiones; es decir, las partes o segmentos que se puedan considerar como unidades significativas. Esto ayuda a obtener una versión más compacta de la información de bajo nivel, ya que, en vez de miles o millones de píxeles, se puede llegar a decenas de regiones.

Las características más comunes para delimitar o segmentar regiones son: intensidad de los píxeles, textura, color, gradiente y profundidad relativa.

Una suposición relevante, que normalmente se asume en visión de nivel intermedio, es considerar que los píxeles de un mismo objeto comparten propiedades similares. Por ejemplo, al procesar una imagen de una manzana, suponemos que el color de sus píxeles es aproximadamente homogéneo. En la vida real esto no es

totalmente cierto, el color de los píxeles varía. Para evitar este tipo de variaciones es mejor considerar un color “aproximadamente” similar sobre una región que a un píxel.

Este no es un problema sencillo, ya que es difícil distinguir las variaciones propias del objeto o por cambios de iluminación (por ej., sombras), de las diferencias por tratarse de otro objeto. A continuación, se muestra una imagen sencilla (figura 7), en la que se distinguen claramente las diferentes regiones (p. 107).

**Figura 7:**

*Ejemplo de imagen con las regiones significativas.*



*Nota.* Cada huevo de diferente color corresponde a una región, además se tiene otras 2 regiones que corresponden al canasto y el fondo. Tomado de Introducción a la Visión Computacional (p. 107), por L. Sucar y E. Gómez, 2011, Inaoe.

Adicionalmente, los autores plantean que existen varias técnicas para la segmentación de regiones. Estas se pueden clasificar en tres tipos:

1. **Locales.** Se basan en agrupar píxeles basándonos en sus atributos y los de sus vecinos (agrupamiento).
2. **Globales.** Se basan en propiedades globales de la imagen (división).

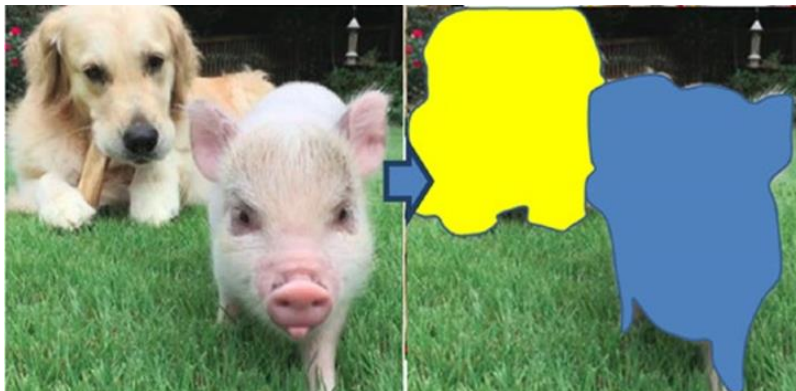
3. **División y agrupamiento (split & merge).** Combinan propiedades de las técnicas locales y globales.

Técnicas que se utilizan con el propósito de facilitar la detección de bordes, localizarlos y luego conectarlos entre sí para formar los límites del objeto.

Así mismo, Gollapudi (2019), plantea que la segmentación es el proceso de hacer una clasificación en píxeles. Esto proporciona una fina separación de objetos, como se muestra en la figura 8, la técnica que resulta muy útil para procesar imágenes médicas e imágenes satelitales (p. 55).

**Figura 8:**

*Segmentación.*



*Nota.* Adaptado de Learn Computer Vision Using OpenCV. With Deep Learning CNNs and RNNs (p. 55), por S. Gollapudi, 2019, Hyderabad, Telangana.

Dicho esto, Dadhich (2018), plantea que la segmentación busca la agrupación de píxeles de una categoría similar. Un ejemplo es el que se muestra en la figura 9, allí vemos que las entradas están a la izquierda y los resultados de segmentación están a la derecha. Los colores de un objeto están de acuerdo con categorías de objetos predefinidos.

Estos ejemplos (figura 9) se toman del conjunto de datos de Pascal VOC que proporciona datasets de imágenes estandarizadas para el reconocimiento de clase de objeto:

**Figura 9:**

*Conjunto de datos de Pascal VOC.*



*Nota.* Adaptado de Practical computer vision (p. 119), por A. Dadhich, 2018, Packt Publishing Limited, Reino Unido.

En la imagen superior a la izquierda, hay varios aviones pequeños en el fondo, por lo tanto, vemos pequeños píxeles coloreados en consecuencia en la imagen correspondiente a la derecha. En la imagen inferior izquierda, hay dos mascotas juntas, en consecuencia, su imagen segmentada a la derecha tiene diferentes colores para los píxeles que pertenecen al gato y al perro respectivamente. Como vemos en esta figura, el límite tiene un color diferente por conveniencia y no implica una categoría diferente.

En las técnicas de segmentación tradicionales, la propiedad clave utilizada son los niveles de intensidad de imagen. Primero, se encuentran diferentes regiones más pequeñas de valores de intensidad similares y luego se fusionan en regiones más grandes. Para obtener el mejor rendimiento, el usuario elige un punto inicial para los algoritmos, aunque los enfoques recientes que se emplean en el aprendizaje profundo han mostrado un mejor rendimiento sin la necesidad de inicialización (p. 118).

Alegre, Gonzalo y Escalera (2016) como se ha indicado previamente, plantean que las técnicas de segmentación permiten encontrar regiones en la imagen más fácilmente.

Para la aplicación de estas técnicas se establecen las propiedades topológicas y de conjunto que definen las regiones, donde  $R$  es la región que incluye la imagen completa. Para la segmentación partimos de un proceso que divide  $R$  en  $n$  subregiones cumpliéndose:

- La unión de todas las regiones forma la región inicial. La segmentación debe ser completa situándose todos los píxeles en alguna región.
- Todas las regiones son regiones conectadas, estando todos los píxeles de una región conectados.
- La intersección entre cualesquiera dos regiones diferentes es vacía.
- Todos los píxeles de una región cumplen determinadas propiedades.
- Los píxeles de regiones diferentes no tienen que cumplir las mismas propiedades (p. 110).

En este sentido, dentro del abanico de posibilidades dadas por los autores antes mencionados, resulta relevante para la presente investigación lo expuesto por Sucar y

Gómez (2011), los cuales dan a entender que no se puede tratar de procesar una imagen asumiendo la homogeneidad de los píxeles sin considerar factores como iluminación, sombras, entre otros generadores de ruido, que hacen que esa homogeneidad sea variable, por lo que proponen que es mejor considerar un color "aproximadamente" que sea similar sobre la región más que en términos de píxeles, clasificando las técnicas de segmentación en 3 tipos, los cuales son: locales, globales y división y agrupamiento (split & merge) que faciliten la detección de bordes y limitaciones de regiones comunes, las cuales sin duda son pautas importantes a tener en cuenta durante el desarrollo de objetivo general que propone esta investigación.

### **Aplicaciones de la Visión Computacional**

Gollapudi (2019) destaca las aplicaciones de visión por computadora en dominios tales como automotriz, atención médica, biomédica y minorista.

**Industria automotriz.** La visión por computadora ha estado en uso constante en la industria automotriz, especialmente para mejorar la seguridad y la funcionalidad de los vehículos modernos. Los vehículos están equipados con sensores y cámaras que pueden recopilar imágenes y datos sobre todo lo que se encuentra alrededor del vehículo.

Por lo tanto, los vehículos pueden detectar señales de límite de velocidad en la carretera, advertir o informar al conductor cuando está estacionado en una zona de no estacionamiento, encontrar de manera proactiva un espacio de estacionamiento abierto y guiar al conductor para llegar a una ubicación. Las advertencias o advertencias

proactivas con respecto a desvíos, estaciones de servicio, obstáculos peligrosos, etc., son solo algunos ejemplos más.

En este sentido y con el propósito de ampliar lo expuesto por el autor, la figura 10 expone un panorama de visión artificial sobre una avenida pública de gran concurrencia vehicular.

**Figura 10:**

*Visión por computadora en la industria automotriz.*



*Nota.* Adaptado de Learn Computer Vision Using OpenCV. With Deep Learning CNNs and RNNs (p. 22), por S. Gollapudi, 2019, Hyderabad, Telangana.

El reciente advenimiento de los automóviles autónomos ha traído una revolución en la forma en que la inteligencia puede incorporarse a los automóviles para controlar la congestión del tráfico, los accidentes de tráfico y el cuidado proactivo de los vehículos.

**Industria de la salud y biomédica.** La industria de la salud y biomédica ha visto la misma cantidad de adopción y tracción para la visión por computadora que la industria del automóvil. Además de los avances en la manera de fotografiar los órganos

humanos y almacenar los datos relacionados, el modo en que se interpretan y visualizan estos datos ha cambiado drásticamente con los algoritmos de visión por computadora.

La atención preventiva ahora es posible para el cáncer y otras enfermedades genéticas porque las computadoras pueden detectar la ocurrencia potencial en imágenes de microscopios con muy alta precisión. Además, los robots pueden realizar cirugías complejas con precisión y eficiencia. En el caso de las cirugías, es importante predecir la cantidad de pérdida de sangre porque puede evitar transfusiones de sangre innecesarias que cuestan aproximadamente \$10 mil millones en todo el mundo. Del mismo modo, almacenar los datos de análisis y obtener información de los informes de salud digitales de los pacientes puede ayudar a mejorar la precisión y la eficacia de los tratamientos.

**Industria de retail.** Tanto las tiendas minoristas en línea como las físicas utilizan ampliamente la visión por computadora para mejorar la experiencia del cliente, proporcionar alternativas competitivas y optimizar los procesos. Amazon, por ejemplo, implementa la visión por computadora para identificar productos similares que están marcados a diferentes precios y proporciona una sugerencia basada en la comparación a los clientes, así como consejos a los vendedores para posicionar un producto en línea en particular.

Una tienda de Amazon Go es una tienda parcialmente automatizada sin estaciones de pago o cajeros que les permite a los clientes comprar productos de su elección y los pagos se realizan fuera de línea a través de sus cuentas de Amazon. Otro ejemplo de una tienda minorista en línea que usa visión artificial es una tienda de

gafas. Los clientes pueden simular cómo se ven los marcos específicos en su rostro o emplear un diseño recomendado para un tipo de rostro y de este modo ayudar a elegir entre las gafas (p. 21).

Aunado a lo anterior, Dadhich (2018) plantea que las aplicaciones de visión por computadora se han vuelto bastante ubicuas en nuestras vidas. Las aplicaciones son variadas, desde juegos de Realidad Virtual (VR) o Realidad Aumentada (AR) hasta aplicaciones para escanear documentos utilizando cámaras de teléfonos inteligentes. En nuestros teléfonos inteligentes, tenemos escáneres de códigos QR y detección de rostros, y ahora incluso tenemos técnicas de reconocimiento facial.

En Internet, ahora podemos buscar empleando imágenes y encontrar imágenes similares. Las aplicaciones para compartir fotos pueden identificar personas y hacer un álbum basado en los amigos o familiares que se encuentran en las fotos. Debido a las mejoras en las técnicas de estabilización de imagen, incluso con las manos más torpes se pueden crear vídeos estables.

Con los recientes avances en las técnicas de deep learning, aplicaciones como la clasificación de imágenes, detección de objetos, seguimiento, etc., se han vuelto más precisas y esto ha llevado al desarrollo de sistemas autónomos más complejos, como drones, automóviles sin conductor, humanoides. Mediante el deep learning, las imágenes se pueden transformar en detalles más complejos; por ejemplo, las imágenes se pueden convertir en pinturas de estilo Van Gogh.

De acuerdo con lo planteado y a propósito de la presente investigación, se destaca el aporte realizado por Gollapudi (2019), el cual resalta los avances en las aplicaciones del modelo visión computacional a través de la incidencia en la calidad de

vida del ser humano, aportes significativos en el área de medicina hoy en día representa alternativas más rápidas y precisas en cuanto a diagnósticos, cirugías y análisis de tratamientos médicos que respondan de forma más favorable y efectiva.

Así como desde el ámbito del retail, permite identificar productos similares que están marcados a diferentes precios y proporcionar una sugerencia basada en la comparación a los clientes, siendo estos solo algunos ejemplos donde la experiencia del cliente y las acciones preventivas son importantes para darle sentido a esta tecnología tan presente en la cotidianidad, cualidades que se persiguen en esta investigación que pretende a través de su implementación brindar una alternativa en cuanto a sistemas de seguridad y vigilancia fácil de usar, económica y precisa, que en especial suma a la calidad de vida de quien lo use.

### **Sistemas de Seguridad y Vigilancia con IA, Basado en el Modelo de Visión Computacional**

Ponce (2011) expresa que en áreas de seguridad que requiere la detección e identificación automática; por ejemplo, en actividad volcánica a partir del monitoreo de anomalías en líneas largas de registro, reconocimiento de caracteres y en los sistemas de vigilancia de video, han sido analizados y probados para ofrecer alternativas paralelas a las tradicionales, mediante el almacenamiento de conocimiento de imágenes. Debido a que el criterio para determinar la actividad volcánica peligrosa es vago, algoritmos basados en la lógica difusa permiten el reconocimiento automático de la actividad, así como la identificación de la morfología respectiva.

En la actualidad se han diseñado diversos dispositivos para el reconocimiento de caracteres impresos, denominados Asociadores Ópticos de Caracteres (OCA), los cuales clasifican entradas de tipo óptico en letras, números y otros caracteres mediante dispositivos de difusificación e inferencias. En el reconocimiento aplicado en cámaras de vigilancia se emplean conocimientos expertos o aprendidos a partir de imágenes previamente grabadas, para determinar mediante la función de asociación cuáles píxeles agrupados en una zona pertenecen a un objeto (p. 36).

En este sentido, Berzal (2018), plantea que un sistema que se utilice de alarma ya sea para la detección de fraudes en operaciones con tarjetas de crédito o la detección de intrusiones en seguridad, debería evitar un excesivo número de falsos positivos. Si estos se producen, probablemente el operador humano deje de prestarles atención a las alertas generadas por el sistema.

Pueden existir situaciones en que los matices del problema puedan influir en nuestra interpretación de la calidad del modelo de clasificación construido. Por ejemplo, no es lo mismo un falso positivo en la detección de cáncer (pruebas posteriores de diagnóstico podrían descartar esa posibilidad) que un falso negativo en la detección de Ébola (dejando que alguien infectado pasee tranquilamente por la ciudad).

Un ejemplo típico de los libros de Estadística es la paradoja de los falsos positivos. Tenemos una prueba diagnóstica que tiene una precisión del 99% a la hora de determinar correctamente si un paciente padece o no una enfermedad no demasiado común, que supongamos que únicamente afecta a una de cada mil personas, un exiguo 0.1% de la población. ¿Cuál es la probabilidad de que tengamos realmente la enfermedad tras recibir un diagnóstico positivo?.

Partiendo de la siguiente tabla de contingencia (figura 11), construida a partir de una población de 100,000 personas, tendremos que la prueba diagnóstica clasifica correctamente 99 de los 100 casos en los que se observa la enfermedad. Esto es,  $TP = 99$  y  $FN = 1$ . Sin embargo, la gran mayoría de la población no padece esa enfermedad. En la segunda columna de la tabla,  $FP = 999$  y  $TN = 98,901$  (para conservar nuestro 99% de precisión global). Es decir, la prueba diagnóstica da positivo para un total de 1,098 casos (99 para los que tienen la enfermedad TP, más 999 para los que no la tienen).

Si recibimos un diagnóstico positivo, solo en el 9% se tratará de un diagnóstico correcto (99/1098). Aunque el diagnóstico acierte en el 99% de los casos cuando ya tenemos la enfermedad, el diagnóstico positivo cuando no sabemos si tenemos la enfermedad solamente nos da un 9% de probabilidad de tener la enfermedad realmente.

Matemáticamente,  $p(D|E) = 0.99$  pero  $p(E|D) = 0.09$ . Este es el motivo por el cual un médico si tiene que diagnosticar una enfermedad grave, debería realizar múltiples pruebas que confirmen el diagnóstico y minimicen la posibilidad de un falso positivo antes de asustarle como paciente (si no sabe probabilidades).

**Figura 11:**

*Paradoja de los falsos positivos en una prueba diagnóstica con el 99% de precisión.*

		Predicción	
		P	N
Clase real	P	99	1
	N	999	98901

*Nota.* Adaptado de Redes neuronales & Deep Learning (p. 63), por F. Berzal, 2018, Edición independiente.

Así mismo, el autor plantea que en situaciones como las anteriores, se pueden asociar costes a las diferentes decisiones de un clasificador. Es habitual que estos costes se presenten en términos monetarios, de forma que tendrá que estimar cuál es el impacto real de cada una de las situaciones reflejadas en su matriz de contingencia para poder comparar modelos alternativos. Si un falso positivo es mucho más costoso que un falso negativo, su matriz de costes debería reflejar este hecho. Por ejemplo, para un banco, un crédito hipotecario concedido a quien no debería, porque terminará no pagándolo, es, en principio, más costoso que un falso negativo (un caso en el que deja de ganar los réditos que le reporta el interés de la hipoteca, descontada inflación). Asumiendo, claro está, la ausencia de riesgo moral.

Por lo que todos los sistemas de detección de anomalías presentan características similares, ya sea para realizar el control de calidad en una fábrica, detectar fallos en sistemas mecánicos, marcar como sospechosas operaciones financieras fraudulentas, detectar intrusiones en un sistema o identificar el comportamiento anómalo de un programa que podría indicar la presencia de virus informáticos. En todas esas aplicaciones, tendremos que afinar y no resumir la calidad de nuestro modelo con una simple medida de precisión o error (pp. 63-64).

En este sentido y como aporte al tema sistemas de seguridad y vigilancia con IA usando el modelo visión computacional, Guido Müller (2017) expresa que una tarea común en el reconocimiento de rostros es preguntar si un rostro nunca antes visto pertenece a una persona conocida de una base de datos. Una forma de resolver este problema sería construir un clasificador donde cada persona sea una clase separada. Sin embargo, generalmente hay muchas personas diferentes en las bases de datos faciales y muy pocas imágenes de la misma persona (es decir, muy pocos ejemplos de capacitación por clase). Eso dificulta entrenar a la mayoría de los clasificadores.

Además, a menudo desea poder agregar nuevas personas fácilmente, sin necesidad de volver a capacitar a un modelo grande. Una solución simple es utilizar un clasificador de un vecino más cercano que busque la imagen de la cara más similar a la cara que está clasificando. En principio, este clasificador podría funcionar con un solo ejemplo de entrenamiento por clase.

Por lo que si obtenemos una precisión del 26,6%, lo que en realidad no es tan malo para un problema de clasificación de 62 clases (las suposiciones aleatorias darían

alrededor de  $1/62 = 1,5\%$  de precisión), pero tampoco es excelente. Únicamente identificamos correctamente a una persona cada cuatro veces.

En este sentido, la vigilancia inteligente sigue siendo un campo de gran desarrollo en la actualidad, tanto en términos de investigación, como comercial, con principal aplicación en China, la cual obtiene la identidad de los ciudadanos en tiempo real, lo que permite clasificarlos, empleando la mayor cantidad de datos posibles (big data), convirtiéndolo en el país con mayor número de patentes en IA, seguido de EEUU y Europa, es pionera en avances significativos en tecnologías de voz o reconocimiento facial, con empresas líderes como iflyTek, SenseTime, Megvii o Cloudwalk Technology, con crecientes exploraciones en sistemas de vigilancia y seguridad, hoy en día expande sus aportes a países como Ecuador, Zimbabue o Tailandia, entre otros, produciendo una tecnología eficiente, competitiva y exportable (p. 149).

De acuerdo con lo planteado por los autores, en seguridad, la detección rápida y precisa en tiempo real de situaciones de riesgo o peligro sigue siendo un tema con muchas problemáticas, subproblemas como la detección y seguimiento de objetos en movimiento, análisis y clasificación de los mismos, siguen en etapas de perfeccionamiento, exploración e investigación, los cuales son retos y oportunidades a los que se enfrenta la presente investigación, que a partir de los avances basados en las diversas técnicas de IA en este campo, en especial el modelo de visión computacional, busca desarrollar un sistema que de forma automática emita alertas de seguridad cuando detecte dentro de un área específica la aparición de un elemento sospechoso que sea detectado como desconocido y que en consecuencia implique

algún riesgo para el usuario, evitando la generación de falsos positivos que sesguen la atención de lo importante de dichas alarmas.

## **Python**

Tal como lo plantea García (2017), Python es un lenguaje de código abierto, multiparadigma, orientado a objetos, imperativo, funcional y reflexivo. Ofrece la flexibilidad necesaria y a la vez es un lenguaje con una curva de aprendizaje muy rápida. Frente a otros lenguajes, como Java o C, Python tiene la ventaja de ser un lenguaje interpretado y no necesita una compilación previa, lo que nos facilita el ciclo de programación y pruebas. A cambio, el rendimiento de la aplicación puede verse afectado. La primera vez que Python ejecuta una aplicación, genera un código intermedio al estilo de Java que hace que en subsiguientes ejecuciones el rendimiento y la velocidad de ejecución sean mejores.

Python en sí es un lenguaje de tipado dinámico, es decir, que no necesitamos declarar el tipo de cada variable que utilizaremos, sino que se referirá automáticamente en el momento de utilizarla por primera vez. También es un lenguaje fuertemente tipado, lo que significa que, una vez que una variable ha sido usada y tiene asignado un tipo concreto, ya solo puede contener datos de ese tipo.

De acuerdo con el autor, a pesar de ser un lenguaje muy utilizado para crear scripts de sistemas, Python es un lenguaje de propósito general que puede utilizarse también para crear complejos programas orientados a objetos y con interfaces gráficas de usuario. Al ser un lenguaje multiplataforma podremos generar aplicaciones que

funcionarán en multitud de sistemas operativos, incluso en más plataformas que el propio lenguaje Java (p. 244).

Por su parte Guido, Müller (2017) plantean que Python se ha convertido en la lengua franca para muchas aplicaciones de ciencia de datos. Combina el poder de los lenguajes de programación de uso general con la facilidad de uso de lenguajes de secuencias de comandos específicos de dominio como MATLAB o R. Python tiene bibliotecas para la carga de datos, visualización, estadísticas, procesamiento de lenguaje natural, procesamiento de imágenes y más. Esta amplia caja de herramientas proporciona a los científicos de datos una gran variedad de funciones de propósito general y especial.

Una de las principales ventajas de usar Python es la capacidad de interactuar directamente con el código, usando un terminal u otras herramientas como el Jupyter Notebook. El aprendizaje automático y el análisis de datos son procesos fundamentalmente iterativos, en los que los datos impulsan el análisis. Es esencial que estos procesos tengan herramientas que permitan una iteración rápida y una interacción fácil. Como lenguaje de programación de uso general, Python también permite la creación de complejas interfaces gráficas de usuario (GUI), servicios web, y facilidad de integración en sistemas existentes (p. 5).

En este sentido, Singh (2019) expone que lo primero que debemos entender sobre Python es cómo guardar los datos y en qué formato se deben guardar los mismos. Necesitamos establecer nuestras variables de imagen como un contenedor, dentro del cual almacenamos los datos. Si no utilizamos las variables, es posible que

podamos hacer cálculos, pero no podremos guardar nuestra salida. Veamos un ejemplo:

```
nombre = 'Saurav'  
edad = 20  
altura = 6.5
```

En este ejemplo, nombre, edad y altura son las variables que almacenan los valores Saurav, 20 y 6.5, respectivamente. Hay algunas reglas que debemos seguir al nombrar una variable:

- Los nombres de las variables deben comenzar con una letra o un guión bajo.
- El resto del nombre de la variable puede consistir en letras, números y guiones bajos.
- Los nombres distinguen entre mayúsculas y minúsculas.
- No debemos usar nombres integrados de Python.

Ahora tenemos que mirar los tipos de datos. En el ejemplo anterior, almacenamos tres tipos de datos: texto, que se incluye entre comillas simples; un número entero y un valor con un decimal. Python sabe automáticamente que cualquier cosa entre comillas es de tipo String, cualquier cosa sin decimal es de tipo Int y cualquier cosa con decimal es de tipo float. Estos son los tres tipos de datos en Python.

Ahora que sabemos cómo guardar un valor dentro de una variable, es posible que necesitemos imprimirlo. La impresión se puede hacer como:

```
name = 'Saurav'  
age = 20  
height = 6.5  
print(name, age, height)  
Output: Saurav 20 6.5
```

Otra forma de imprimir los valores usando formato:

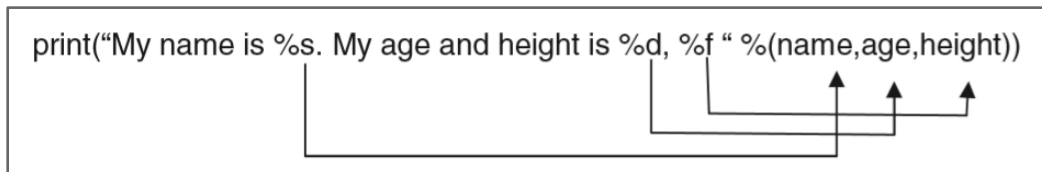
```
name = 'Saurav'  
age = 20  
height = 6.5  
print("My name is %s. My age and height is %d, %f" % (name, age,  
height))
```

Output: My name is Saurav. My age and height is 20, 6.5

En el ejemplo anterior, %s representa String, donde "s" significa string; %d representa Int, donde "d" significa dígito o número entero; y %f representa decimales, donde "f" significa float. Entonces, el primer conector está conectado a la primera variable, el segundo al segundo y el tercero al tercero. Todo se une utilizando "%", tal como se observa en la figura 12.

### Figura 12:

*Cómo funcionan los conectores.*



*Nota.* Adaptado de Practical Machine Learning and Image Processing. Primera edición (p. 32), por H. Singh, 2019, Allahabad, Uttar Pradesh.

En la sección anterior, se vio cómo guardar un valor dentro de una variable. Pero, si se quiere guardar más de un valor, se debe usar estructuras de datos de Python, que incluyen lo siguiente:

- Listas
- Diccionarios

## – Tuplas

Siendo estas estructuras las más empleadas en el procesamiento de imágenes.

En conclusión, todos los autores exponen las cualidades y ventajas de usar Python como lenguaje principal en el desarrollo de IA, gracias a sus grandes habilidades, al ser un lenguaje interpretado, no necesita una compilación previa, lo que facilita el ciclo de programación y pruebas, cualidades ideales para el desarrollo del sistema de reconocimiento de imágenes para emitir alertas de seguridad que propone esta investigación (p. 30).

En este sentido, funciona como guía ideal para el proceso de desarrollo del código lo planteado por Singh (2019), el cual expone que lo primero que debe entender sobre Python es cómo guardar los datos y en qué formato, planteando ejemplos para establecer variables de imagen como un contenedor, dentro del cual almacenamos los datos, estructuras de datos, sentencias, brindando así una visión profunda de los conceptos básicos del procesamiento de imágenes y diversas metodologías y algoritmos de procesamiento de imágenes, aplicaciones que utilizan varias bibliotecas de Python e implementación de casos de uso en tiempo real utilizando enfoques de aprendizaje automático.

## **OpenCV**

Según Vaidya (2018), OpenCV es una biblioteca de visión por computadora desarrollada teniendo en cuenta la eficiencia computacional y el rendimiento en tiempo real. Está escrito en C/C++ y contiene más de cien funciones que ayudan en las aplicaciones de visión por computadora. La principal ventaja de OpenCV es que es de

código abierto y se lanzó bajo la licencia BSD (Berkeley Software Distribution) que permite el uso gratuito de OpenCV en aplicaciones comerciales y de investigación. Esta biblioteca tiene una interfaz en los lenguajes C, C++, Java y Python y se puede usar en varios sistemas operativos, como Windows, Linux, macOS y Android, sin modificar una sola línea de código.

Esta biblioteca también puede aprovechar el procesamiento multinúcleo, OpenGL y CUDA para el procesamiento paralelo. Como OpenCV es liviano, también se puede utilizar en plataformas integradas como Raspberry Pi. Esto lo hace ideal para implementar aplicaciones de visión por computadora en sistemas integrados en escenarios de la vida real. Estas características han hecho de OpenCV una opción predeterminada para los desarrolladores de visión por computadora. Tiene una amplia base de desarrolladores y una comunidad de usuarios que ayuda constantemente a mejorar la biblioteca. Las descargas para OpenCV son de millones y aumentan día a día (p. 189).

En este mismo orden de ideas, Rosebrock (2017) plantea que no es necesario conocer la biblioteca de procesamiento de imágenes y visión por computadora de OpenCV de primera mano, ya que el mismo al igual que Python poseen una gran curva de aprendizaje que facilita su rápida comprensión en operaciones básicas de procesamiento de imágenes, cargar una imagen desde el disco, mostrarla en nuestra pantalla y algunas otras operaciones básicas, por lo que si se es nuevo en OpenCV y la visión por computadora, es recomendable trabajar sobre ejemplos básicos donde se exploren poco a poco sus cualidades (p. 19).

En conclusión, OpenCV al ser una plataforma liviana, resulta ideal para implementar aplicaciones basadas en el modelo de visión computacional en sistemas integrados en escenarios de la vida real, lo que permite la integración de varias plataformas como Raspberry Pi, además de una interfaz en los lenguajes C, C ++, Java y Python, siendo este último un lenguaje tipado de programación sobre el cual se apoyará la presente investigación, además de la ayuda en cuanto a soporte técnico gracias a que cuenta con una amplia comunidad de desarrolladores que participan activamente en la retroalimentación y mejoramiento de esta biblioteca, lo que sin duda facilitará la experiencia de desarrollo e implementación del sistema de reconocimiento de imágenes propuesto en esta investigación.

### **Glosario de términos**

- **Inteligencia artificial.** Disciplina académica relacionada con la teoría de la computación cuyo objetivo es emular algunas de las facultades intelectuales humanas en sistemas artificiales. (Benítez, Escudero y Kanaan, 2014, p. 10)
- **Robótica.** Disciplina informática que se ocupa del diseño, la programación, la ingeniería y el desarrollo de robots o máquinas físicas que se construyen para ejecutar tareas que generalmente realizan los humanos. (Gollapudi, 2019, p. 10)
- **Redes neuronales.** Las redes neuronales artificiales o RNAs son un intento de emular la forma de trabajar del cerebro humano. (García, 2017, p. 209)

- **Machine learning.** Es una manera de construir inteligencia en una máquina para que pueda aprender con el tiempo y mejorar utilizando su propia experiencia. (Gollapudi, 2019, p. 11)
- **Deep learning.** Es un área de aprendizaje automático que se centra en unificar el aprendizaje automático con inteligencia artificial. (Gollapudi, 2019, p. 12)
- **Datasets.** Son un conjunto de datos del que la IA puede extraer conocimiento.
- **Visión computacional.** Es uno de los subcampos de la IA más ampliamente estudiados de la informática, con aplicaciones importantes en detección de rostros, búsqueda y conversión de imágenes.
- **Clasificación.** Es la tarea de etiquetar toda la imagen con un objeto o concepto con confianza. (Gollapudi, 2019, p. 53)
- **Detección y reconocimiento.** Es una actividad que consiste en encontrar un objeto y marcarlo o encuadrarlo. (Gollapudi, 2019, p. 54)
- **Segmentación.** Es una manera de dividir la imagen mediante la determinación de los bordes. (Sucar y Gómez, 2011, p. 107)
- **Sistemas de seguridad y vigilancia.** Sistemas de reconocimiento y detección automática, en tiempo real, basado en imágenes y videos.
- **Python.** Es un lenguaje de programación tipado dinámico, de código abierto.
- **OpenCV.** Es una biblioteca de visión por computadora desarrollada teniendo en cuenta la eficiencia computacional y el rendimiento en tiempo real. (Vaidya, 2018, p. 189)

## Operacionalización y Conceptualización de las Variables

**Tabla 2:**

*Tabla de variables.*

<b>Objetivo General:</b> Desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad.			
<b>Objetivos específicos</b>	<b>Variables</b>	<b>Dimensiones</b>	<b>Indicadores</b>
Identificar las herramientas de hardware y software a requerir para desarrollo de la IA.		<b>Datasets.</b> <b>Python.</b> <b>OpenCV.</b>	Colección de datos. Lenguaje de tipado dinámico. Biblioteca de visión por computadora.
Aplicar algoritmos inteligentes de reconocimiento de imágenes, bajo el modelo de visión computacional.	<b>Inteligencia artificial.</b>	<b>Visión computacional.</b>	Visión computacional.
Entrenar la IA en el reconocimiento de imágenes basado en patrones establecidos.	<b>Visión computacional.</b>	<b>Clasificación.</b> <b>Detección y reconocimiento.</b> <b>Segmentación.</b>	Clasificación. Detección y reconocimiento. Segmentación.
Estudiar métricas que permitan comprobar el funcionamiento de la IA desarrollada.		<b>Sistemas de seguridad y vigilancia con IA, basado en el modelo de visión computacional.</b>	Falsos positivos. Medidas de calidad como la precisión o la tasa de error. Clasificador por defecto.
Realizar las pruebas de la IA en el envío de alertas de seguridad, en un ambiente real.			Precisión en los resultados.

## **CAPÍTULO III**

### **MARCO METODOLÓGICO**

En este capítulo se presenta la metodología empleada por el investigador con el propósito de responder a las interrogantes planteadas en el presente estudio. En ese sentido se señala, el tipo y diseño de investigación, técnicas para la recolección de datos y presentación de los resultados.

El campo de la investigación, en especial el de la investigación holística, la cual según Hurtado (2010) “es una propuesta que presenta la investigación como un proceso global, evolutivo, integrador, concatenado y organizado” (p. 11), propone la invención y formulación de propuestas novedosas a través del análisis, interpretación y clasificación de fuentes de información relevantes, que de acuerdo al objetivo general permite soluciones prácticas a problemas dados en entornos generales o específicos, generando así nuevos conocimientos a través de los resultados obtenidos.

#### **Tipo de Investigación**

De acuerdo con lo planteado, se tomará como referencia lo propuesto en investigación holística por Hurtado (2012), como parte de una comprensión integradora de la investigación, a través de un modelo del proceso investigativo, que comprende teorías, técnicas y análisis a partir del tipo y diseño de investigación que más se oriente a la consecución de los objetivos propuestos, abriendo el panorama en cuanto a las

posibilidades de aplicación dentro del ámbito práctico-social del objetivo general propuesto.

En este sentido y de acuerdo con el objetivo del estudio, el tipo de investigación es proyectiva, la cual y según plantea Hurtado (2012) “propone soluciones a una situación determinada a partir de un proceso de indagación. Implica explorar, describir, explicar y proponer alternativas de cambio; abarcando todas las investigaciones que implican el diseño o creación de algo con base en un proceso investigativo” (p. 160). Es así, que desde el punto de vista metodológico el estudio recomendado responde a un tipo de investigación proyectiva, el cual responde al objetivo principal de la investigación el cual es desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad.

### **Diseño de la Investigación**

Según Baena (2014) plantea que sin ordenar y sistematizar los conocimientos no hay ciencia. Cada autor, pensador, corriente, tienen su manera propia para integrar lo que percibe como la realidad. La metodología ejerce el papel de ordenar, se apoya en los métodos, como sus caminos y estos en las técnicas como los pasos para transitar por esos caminos del pensamiento a la realidad y viceversa (p. 43).

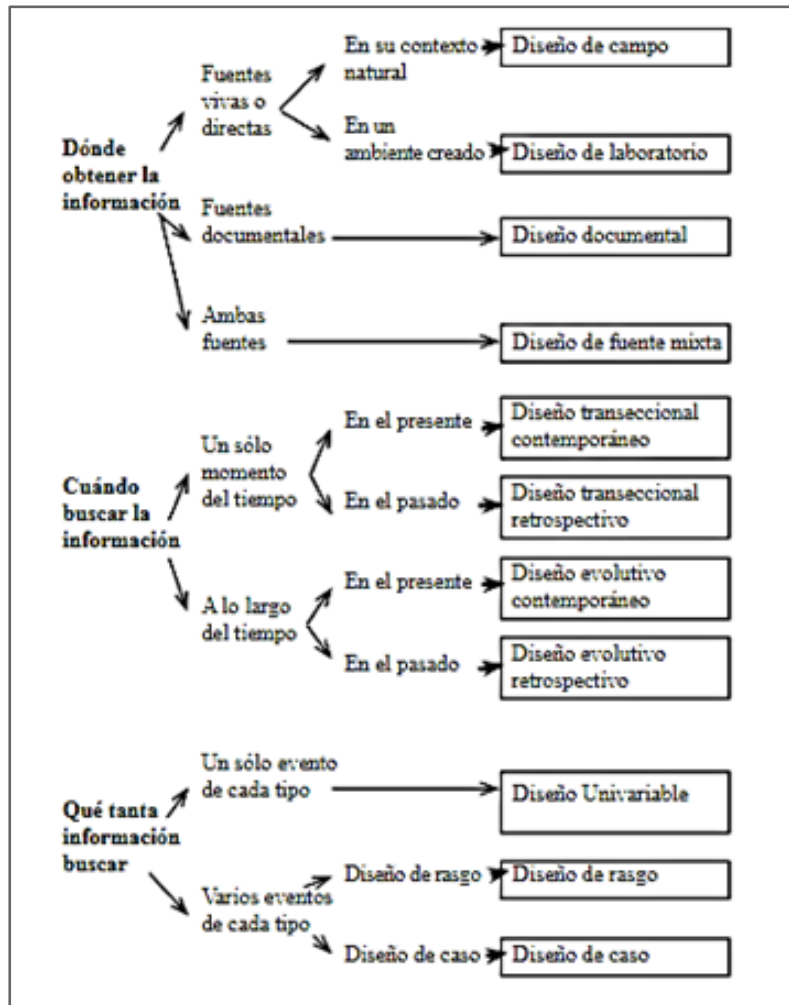
Por su parte, Hurtado (2012) dice que “el diseño de la investigación hace explícitos los aspectos operativos de la misma. Si el tipo de investigación se define con base en el objetivo, el diseño se define con base en el procedimiento” (p. 155).

Significando así, un esquema de trabajo o plan de acción del investigador, donde se organiza de manera formal cada una de las etapas que permitirán el logro de los

objetivos propuestos, que incluyen diversos subprocesos que describen con detalle qué se debe hacer y cómo realizarlo (ver figura 13).

**Figura 13:**

*Esquema para seleccionar los diseños de investigación.*



*Nota.* Adaptado de El Proyecto de Investigación Holística. Comprensión Holística de la metodología y la investigación (p. 175), por J. Hurtado, 2012, Ediciones Quirón. Fundación SyPAL.

En este sentido, en cuanto a obtención de información el diseño de investigación es de fuente mixta, relacionado con la integración del diseño documental y de

laboratorio. Siendo documental, ya que a través del bosquejo e indagación sobre información dada por investigaciones previas, documentales y demás fuentes bibliográficas, se busca complementar y descubrir nuevas herramientas y técnicas asociadas al reconocimiento de imágenes con visión computacional, que permita comprender y analizar posibles optimizaciones en el diseño de los algoritmos inteligentes para su posterior entrenamiento, así como reconocer y aplicar mejores recursos en cuanto a hardware y software.

Siguiendo los lineamientos que propone el diseño de investigación de fuentes mixtas, es de laboratorio, ya que se plantea dentro de un ambiente controlado propuesto por el investigador, el cual dispone un área controlada como entorno principal donde se focalizan las pruebas, tanto como el estudio y análisis de los resultados obtenidos.

### **Técnicas de Recolección de Datos**

Las técnicas e instrumentos de recolección de datos son las distintas formas y maneras de obtener la información. Arias, (2006) indica que “un instrumento de recolección de datos es cualquier recurso, dispositivo o formato (en papel o digital), que se utiliza para obtener, registrar o almacenar información” (p. 69).

Sin embargo, para Hurtado (2010) la selección de técnicas e instrumentos de recolección de datos implica determinar por cuáles medios o procedimientos el investigador obtendrá la información necesaria para alcanzar los objetivos de la investigación.

El *cómo* alude a las técnicas que tienen que ver con los procedimientos usados para la recolección de datos, las cuales pueden ser de revisión documental, observación, encuesta y técnicas sociométricas, entre otras.

Así mismo, los instrumentos representan las herramientas con las cuales se va a recoger, filtrar y codificar la información, es decir, *con qué*. Dichos instrumentos pueden estar ya elaborados e incluso normalizados, aunque si se tratasen de eventos poco estudiados, puede ser necesario que el investigador elabore sus propios instrumentos, los cuales pueden ser listas de cotejos, escalas o cuestionarios, entre otros (p. 161).

Para la presente investigación las técnicas de recolección de datos a aplicar se basan en la recopilación y verificación de fuentes de información a través de los distintos medios disponibles (físicos, digitales), creación de cuadros descriptivos que permitan organizar la información y su fuente, enunciando así las teorías que sustentan la investigación, siguiendo con la elaboración de un diario de campo donde se registran los distintos aspectos observados en cada una de las pruebas realizadas dentro del entorno controlado, describiendo así las actividades, procesos y sucesos, categorizando y sistematizando posteriormente las observaciones obtenidas.

### **Análisis y Presentación de Resultados**

Dentro de la fase confirmatoria del proceso metodológico, Hurtado (2010) plantea que una vez aplicados los instrumentos y finalizada la recolección de datos, el investigador deberá organizarlos y aplicar un tipo de análisis que le permita llegar a una conclusión, en función a los objetivos que se planteó al principio, a fin de resolver el problema de investigación (p. 181).

En este sentido y de acuerdo con el proceso de tratamiento de resultados que plantea la investigación holística, se realiza la categorización de la información basada en el objeto de estudio mediante un enfoque descriptivo y analítico, donde se organizan y agrupan los datos obtenidos del diario de campo, revelando los cambios, problemáticas y avances derivados del diseño y ejecución de las pruebas realizadas en diferentes estadios de la investigación de forma ordenada y resumida.

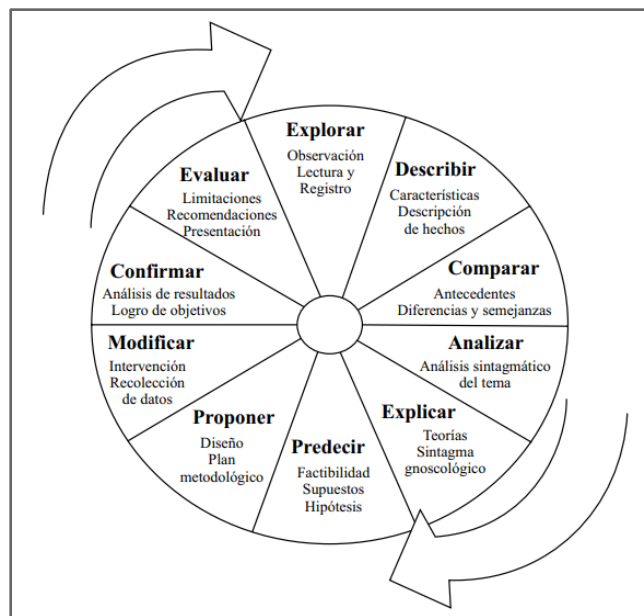
Así mismo, debido a su valor cualitativo, una vez clasificada y unificada información, se hace necesario el análisis de los resultados para dar lugar a la organización y presentación de evidencias que demuestran los hallazgos obtenidos durante todo el proceso investigativo.

### **Metodología Holística: Investigación Proyectiva**

La metodología holista de acuerdo con Hurtado (2012), propone que cada evento refleja y contiene a la vez las dimensiones de la totalidad que lo comprende. Cada modo u holotipo de investigación comprende de una forma trascendente, estadios investigativos anteriores (acción integradora) y posibles desarrollos futuros (acción proyectiva), desplegando de esta forma el modelo de Ciclo Holístico como circuito global, continuo, concatenado e integrado, que ofrece soporte metodológico y epistémico al investigador (ver figura 14).

**Figura 14:**

*Ciclo Holístico de la Investigación. Investigación proyectiva.*



*Nota.* Adaptado de El Proyecto de Investigación Holística. Comprensión Holística de la metodología y la investigación (p. 21), por J. Hurtado, 2012, Ediciones Quirón. Fundación Sypal.

Estos niveles expresan la profundidad del estudio, y al igual que los objetivos, son integradores porque buscan haber logrado objetivos de menor profundidad para plantearse un objetivo más complejo, dando lugar a logros sucesivos dentro del mismo proceso de investigación (p. 156).

Es así y de acuerdo con las fases que propone la investigación proyectiva, en las etapas iniciales de la investigación se **delimitó el tema o evento a modificar**, se identificaron las fuentes de información y antecedentes, que permitieron diagnosticar la situación actual, lo que llevó a la determinación del enunciado holopráxico orientado al siguiente cuestionamiento ¿Es posible desarrollar a bajo costo y con recursos propios un sistema de reconocimiento de imágenes para emitir alertas de seguridad de forma

rápida y precisa usando inteligencia artificial (IA)?. Seguidamente, tal como lo presenta la fase descriptiva del proceso metodológico, se procedió al **desarrollo de la justificación y planteamiento de los objetivos**, tanto general como específicos.

Así mismo, se desarrolló el **sintagma gnoseológico**, que comprende las fases comparativas, analíticas y explicativas del proceso metodológico, donde se conceptualizó y desarrolló la teoría acerca del evento a modificar mediante una revisión exhaustiva de las fuentes de información disponibles, revisión de teorías, definiciones y comparación de los distintos aportes y apreciaciones dadas por los distintos autores e investigaciones previas relevantes, permitiendo ampliar y profundizar sobre las ideas que justifican la necesidad de llevar a cabo el objeto propuesto.

En este sentido y para satisfacer la fase predictiva relacionada a la **revisión de la factibilidad de la investigación**, una vez finalizada la revisión bibliográfica y construido el sintagma gnoseológico, se estimó si es necesario el diseño de una propuesta de solución dadas las condiciones actuales, posibles dificultades, estrategias, instrumentos y diseños que en consecuencia permitan establecer como posible la consecución exitosa de los objetivos planteados.

Como fase proyectiva, se precisó **la determinación de los lineamientos metodológicos**, que permitieron identificar el evento a modificar, su delimitación y definición de los procesos explicativos desde un punto de vista conceptual y operacional, seguido de la selección del diseño de investigación, determinado de acuerdo al contexto donde se obtiene la información, derivando en un diseño de fuente mixto que aborda el diseño documental en un ambiente controlado (laboratorio),

finalizando con la selección de técnicas y construcción de instrumentos de recolección de datos.

En este orden de ideas, la fase interactiva del proceso metodológico consistirá en la **intervención para la recolección de datos**, organizando, interpretando y accediendo a información relevante, con el propósito de responder de forma efectiva a cada uno de los objetivos planteados, así como describir la situación actual y la evolución a lo largo de las pruebas realizadas sobre el objeto o evento a modificar.

**Análisis, integración y presentación de los resultados**, comprenden la fase confirmatoria del proceso metodológico donde se utilizarán técnicas de análisis y elaboración de conclusiones que encajen con la configuración del plan de acción, analizando descriptivamente cada uno de los objetivos específicos que en consecuencia permita presentar de manera ordenada e integral los resultados obtenidos en cada uno de los objetivos propuestos.

Finalmente, la **evaluación del proceso**, la cual es una fase que se desarrolla a lo largo de la discusión de los resultados en el capítulo V (Conclusiones y recomendaciones) de la investigación, planteando las limitaciones y recomendaciones, correspondiendo de esta forma a la evaluación del proceso investigativo, determinando en qué medida se han alcanzado los objetivos y bajo cuáles condiciones, permitiendo definir las dificultades y líneas estratégicas disponibles para la etapa de aplicación.

En este sentido, una vez establecidas las especificaciones de la metodología holística en cada uno de sus estadios y para sintetizar la aplicación de la metodología planteada se aplica el método científico desde el análisis hasta la proyección, a través de las siguientes etapas:

**Etapa I: Análisis.** Esta etapa comprende los niveles *exploratorios, descriptivos, comparativos y analíticos* descritos en la investigación holística como aspectos iniciales de la investigación proyectiva, permitiendo exponer los parámetros y requerimientos previos necesarios para el diseño de la propuesta, brindando un enfoque práctico y conciso de las condiciones previas necesarias para la consecución de los objetivos propuestos.

**Etapa II: Diseño.** Dado los resultados obtenidos en la etapa de análisis, la etapa de diseño toma dichos resultados para dar comienzo al nivel *explicativo, predictivo y de propuesta* del ciclo de investigación holística, donde se destaca el uso de las herramientas de diseño implementadas en los algoritmos de IA, así como las técnicas de recolección de datos para confirmar especificaciones y detalles de diseño.

**Etapa III: Pruebas.** Representa la fase integradora de los resultados obtenidos en las etapas previas de análisis y diseño, exponiendo una mirada definitiva del proceso de pruebas de la propuesta, comprendiendo de este modo la fase evaluativa que dará pie al capítulo V de conclusiones y recomendaciones basadas en los resultados obtenidos en cada uno de los objetivos propuestos.

## **CAPÍTULO IV**

### **APLICACIÓN DE LA METODOLOGÍA**

Con el propósito de cumplir con los lineamientos propios de una investigación científica bajo el enfoque holístico, se plantea un proceso organizado y evolutivo que tal como lo expone la investigación holística surge como una necesidad de proporcionar criterios de apertura y una metodología más completa a las personas que realizan investigaciones en diversas áreas del conocimiento, “teniendo como clave centrarse en los objetivos como logros sucesivos de un proceso continuo, más que como un resultado final, presentando la investigación como un proceso global, evolutivo, integrador y organizado” (Hurtado, 2010, p. 11).

Debido a que el tipo de investigación es proyectiva, se consideran todos los aspectos que integran un problema o situación, para así dar una respuesta global y efectiva con miras al futuro. De acuerdo con la naturaleza del diseño mixto, documental y de laboratorio que presenta la investigación, se busca que a través de la exploración científica y considerando los antecedentes de investigaciones relacionadas obtener nuevos conocimientos para su posterior aplicación, por lo que, partiendo de los diferentes estadios propuestos en el ciclo holístico de la investigación, se describen las siguientes etapas:

#### **Etapas I: Análisis**

La presente etapa comprende la fase exploratoria y descriptiva de los criterios, características (hardware y software) y parámetros mínimos que resultan

fundamentales para el proceso de diseño del sistema de reconocimiento de imágenes para emitir alertas de seguridad.

## **Situación Actual**

Para comprender la situación actual es importante tener en cuenta que la investigación se delimita dentro de un área controlada por el investigador (su domicilio), ya que se adecua al objetivo central de la investigación que establece *Desarrollar un sistema de reconocimiento de imágenes para emitir alertas de seguridad*, teniendo como referencia principal la practicidad en cuanto a diseño, aplicación y costos de un mecanismo de vigilancia que permita detectar en tiempo real la presencia de personas ajenas al domicilio o área donde se implemente dicho sistema bajo el enfoque de visión computacional, que en consecuencia le brinde información precisa, analizada con elementos de inteligencia artificial que le permita al usuario actuar de forma más temprana ante cualquier problema de seguridad.

Aunado a lo anterior, el entorno no sufrió modificaciones, ya que al pretender desarrollar un sistema accesible, se pensó en un diseño simple y que se pueda ubicar de manera más cómoda dentro del ambiente donde se use, es así que a través del aprovechamiento de distintos recursos, aportes documentales y científicos se logró el desarrollo de dicho sistema en beneficio de la seguridad y que en consecuencia sirva de referencia para futuras investigaciones que deseen tomar los aportes planteados y mejorarlos.

## **Factibilidad**

La factibilidad de la investigación está orientada sobre la disponibilidad de los diversos recursos inexorables para el desarrollo de la propuesta dado los objetivos planteados, concluyendo su viabilidad de acuerdo con los siguientes criterios:

### **Factibilidad Técnica:**

- En relación con los requerimientos de componentes físicos, por su naturaleza son totalmente accesibles al público general, siendo algunos componentes de uso básico en electrónica, todos los elementos necesarios están disponibles para su compra en diversas tiendas de electrónica y computación, sea de forma física o por internet, con opciones de envío tanto nacional como internacional.
- En cuanto a los conocimientos individuales requeridos, se debe contar con nociones básicas en electrónica, es mandatorio tener conocimientos en programación, así como un entendimiento sobre conceptos de IA y de visión computacional, por lo que la presente investigación se sustenta y alimenta de diferentes tipos de documentación, libros y artículos digitales de autores como: Russell, R. (2018) Machine Learning - Guía Paso a Paso Para Implementar Algoritmos., Rosebrock, A. (2017) Deep Learning for Computer Vision with Python., Guido, S y Müller A. (2017) Introduction to Machine Learning with Python, entre otros.

**Factibilidad Económica:** Siendo uno de los propósitos de la investigación el aportar un producto de bajo costo, se adquirieron complementos y herramientas de alta calidad a bajo costo, para ello se analizó las propuestas disponibles en el mercado sin

perjudicar la calidad de los recursos, dando como resultado un proyecto sostenible y desarrollable en su totalidad en cuanto a hardware.

**Factibilidad Biográfica:** Para validar y sustentar la presente investigación, la misma se apoyó en aportes de distintos autores destacados en temas de machine learning y visión computacional, por medio de documentales, libros y artículos digitales, así como el apoyo en cuanto a implementación de propuestas en tesis de investigación tales como es el caso de López (2016), en su trabajo especial de grado, titulado Sistema de inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos en áreas de almacenamiento de las Pymes, sirvieron de antecedentes al igual que Ibarra (2017) y García (2018), gracias al desarrollo de sus propuestas.

Coincidiendo en sus investigaciones, que como primeros pasos debe existir un análisis de la tecnología existente y antecedentes relacionados, para reconocer vulnerabilidades y la factibilidad del estudio, así como una representación de los resultados de forma gráfica que permite visualizar más fácilmente aspecto de efectividad e inconsistencia halladas por el investigador.

## **Requerimientos del Sistema**

La selección de los elementos de hardware y software es sin duda uno de los pasos fundamentales para llevar a cabo cualquier proyecto tecnológico de forma exitosa, la integración entre ambos componentes debe ser lo más óptima posible, de modo que el software mediante programación y demás recursos como bibliotecas

pueda actuar de la mano con los componentes físicos en el procesamiento de la información que se le proporciona.

## **Hardware**

Tal como se planteó anteriormente es necesario el empleo de componentes físicos que actúen de forma sinérgica con el software para el procesamiento de información, por lo que en este sentido a continuación se describen las especificaciones requeridas en cuanto a hardware:

### *Microcomputador*

Resulta importante la selección de una unidad de procesamiento para ejecutar el sistema operativo y cuyos componentes respondan de forma efectiva con lo que propone la investigación; en este sentido, se optó por el microordenador Raspberry Pi 3B+ que es una computadora de una sola placa (SBC: Single-Board Computer en inglés).

Existe una variedad de accesorios para esta computadora, por lo que se seleccionó un kit de fácil armado que contiene: una placa Raspberry Pi 3 B+ (figura 15) del fabricante ELEMENT14, gabinete transparente de acrílico, fan cooler, disipadores de aluminio adhesivos y fuente de alimentación 2.5 A.

**Figura 15:**

*Kit Raspberry Pi 3 B+.*



*Nota.* Adaptado de página oficial de Mercado libre, 2021, Kit Raspberry Pi 3 B+. Mercadolibre. ([https://articulo.mercadolibre.com.ar/MLA-741441988-kit-raspberry-pi-3-b-plus-uk-gab-dis-fan-fuente-emakers-\\_JM](https://articulo.mercadolibre.com.ar/MLA-741441988-kit-raspberry-pi-3-b-plus-uk-gab-dis-fan-fuente-emakers-_JM)).

**Tabla 3:**

*Características de microcomputador Raspberry Pi 3 B+.*

<b>Características</b>	<b>Detalles</b>
Marca	Raspberry Pi
Modelo	Modelo 3 B+
Microcontrolador	ARM
Voltaje de funcionamiento	5 V

Procesador	Broadcom BCM2837B0, Cortex-A53 SoC de 64 bits a 1,4 GHz
Memoria	1 GB LPDDR2 SDRAM
Conectividad	<ul style="list-style-type: none"> <li>- 2,4 GHz y 5 GHz IEEE 802.11.b / g / n / ac inalámbrico, LAN, Bluetooth 4.2, BLE</li> <li>- Gigabit Ethernet sobre USB 2.0 (rendimiento máximo)</li> </ul>
Puertos USB	4 puertos USB 2.0
Acceso	Encabezado GPIO ampliado de 40 pines
Video y sonido	<ul style="list-style-type: none"> <li>- 1 × HDMI de tamaño completo.</li> <li>- Puerto de visualización MIPI DSI.</li> <li>- Puerto de cámara MIPI CSI.</li> <li>- Salida estéreo de 4 polos y puerto de video compuesto.</li> <li>- Multimedia: decodificación H.264, MPEG-4 (1080p30).</li> <li>- Soporte de tarjeta SD.</li> </ul>
Especificaciones	<ul style="list-style-type: none"> <li>- El Raspberry Pi 3 Model B+ es el último producto de la gama Raspberry Pi 3.</li> <li>- La LAN inalámbrica de doble banda.</li> <li>- La Raspberry Pi 3 Modelo B + mantiene el mismo formato que la Raspberry Pi 2 Model B y la Raspberry Pi 3 Model B.</li> </ul>
<b>Costo</b>	USD 35 \$

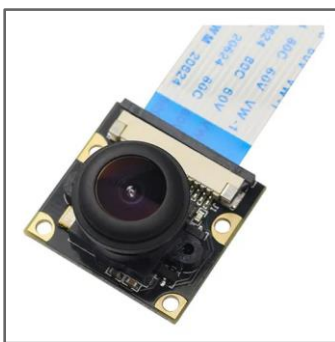
### *Cámara Digital*

Siendo este el dispositivo encargado de la captura de imágenes, se indagó sobre las ofertas actuales en el mercado, existiendo una gran variedad de cámaras

digitales que en menor o mayor medida ofrecen cualidades en cuanto a rendimiento, ahorro de energía y resolución, todo en relación con el costo y el fabricante, eligiendo para la presente investigación un modelo de cámara Raspberry Ov5647 Visión Nocturna Ojo Pez 130° de 5mp (figura 16), con facilidad de conexión con la tarjeta de desarrollo.

**Figura 16:**

*Cámara Raspberry Ov5647.*



*Nota.* Adaptado de página oficial de Mercado libre, 2021, Cámara Raspberry Ov5647 Visión Nocturna Ojo Pez 130 5mp Luz. Mercadolibre. ([https://articulo.mercadolibre.com.ar/MLA-767685590-camara-raspberry-ov5647-vision-nocturna-ojo-pezo-130-5mp-luz-\\_JM](https://articulo.mercadolibre.com.ar/MLA-767685590-camara-raspberry-ov5647-vision-nocturna-ojo-pezo-130-5mp-luz-_JM)).

**Tabla 4:**

*Características de la Cámara Raspberry Ov5647.*

Características	Detalles
Marca	Raspberry Pi
Modelo	Ov5647
Voltaje de funcionamiento	3.3 V

Especificaciones

- Sensor: OV5647
- Visión nocturna
- Tamaño de la lente: 1/4"
- Resolución 1080P / 5MP
- Rango de visión: 130°
- Filtro IR: No
- Dimensión: 25 mm x 24 mm

---

**Costo**

USD 10 \$

---

Adicionalmente, se optó por la adquisición de módulos de luz infrarroja para la cámara Ov5647 (figura 17), que ayudan a la detección de objetos en ambientes de poca luz.

**Figura 17:**

*Módulos de luz infrarroja para cámara Ov5647.*



*Nota.* Adaptado de página oficial de Mercado libre, 2021, Módulo Omnivision Raspberry Para Cámara Ov5647 Visión N. Mercadolibre. ([https://articulo.mercadolibre.com.ar/MLA-817459637-luz-modulo-omnivision-raspberry-para-camara-ov5647-vision-n-\\_JM](https://articulo.mercadolibre.com.ar/MLA-817459637-luz-modulo-omnivision-raspberry-para-camara-ov5647-vision-n-_JM))

**Tabla 5:***Módulos de luz infrarroja Para Cámara Ov5647.*

<b>Características</b>	<b>Detalles</b>
Marca	Genérico
Modelo	OV5647
Voltaje de funcionamiento	5 vcc
Especificaciones	<ul style="list-style-type: none"><li>- Luz doble para módulo de minicámara infrarroja OV5647 Visión Nocturna.</li><li>- Doble foco infrarrojo de alto rendimiento.</li></ul>
<b>Costo</b>	USD 3 \$

### *Sensor de Movimiento*

Uno de los elementos que complementa el funcionamiento del proyecto, es sin duda el sensor de movimiento que responde a factores como distancia, profundidad y resolución del objeto en la escena, transformando la luz incidente en señales eléctricas. Existen diversas tecnologías que llevan a cabo dicha tarea, como sensores por luz infrarroja pasiva (PIR), sensores para detección de frecuencia de microondas (efecto Doppler), sensores ultrasónicos, entre otros; también existen sensores que combinan diferentes tecnologías.

De este modo y fundamentado en las cualidades antes mencionadas, se optó por el sensor de movimiento infrarrojo PIR de bajo costo, el Hc Sr501 (figura 18), el cual detecta el movimiento de hasta 7 metros de distancia mediante el uso de una lente

de Fresnel y el elemento sensible al infrarrojo para detectar cambios en los patrones de radiación infrarroja emitida por los objetos en su entorno. Tiene un bajo costo y es fácil de usar, ideal para sistemas de alarmas e iluminación activada por movimiento.

**Figura 18:**

*Módulo sensor de movimiento infrarrojo PIR Hc Sr501.*



*Nota.* Adaptado de página oficial de Mercado libre, 2021, Módulo Sensor De Movimiento Infrarrojo Pir Hc Sr501 Arduino. Mercadolibre. ([https://articulo.mercadolibre.com.ar/MLA-727862643-modulo-sensor-de-movimiento-infrarojo-pir-hc-sr501-arduino-\\_JM](https://articulo.mercadolibre.com.ar/MLA-727862643-modulo-sensor-de-movimiento-infrarojo-pir-hc-sr501-arduino-_JM)).

**Tabla 6:**

*Características del Módulo sensor de movimiento infrarrojo Pir Hc Sr501.*

<b>Características</b>	<b>Detalles</b>
Marca	Genérico
Modelo	HC-SR501
Voltaje de funcionamiento	4.5 V - 20 V
Especificaciones	<ul style="list-style-type: none"><li>- Sensor infrarrojo con controladora en circuito.</li><li>- Sensibilidad y tiempo de disparo.</li><li>- Rango de Trabajo: 4.5 V – 20 V</li><li>- Consumo: &lt;60uA</li></ul>

- Voltaje de Salida: High/Low nivel de señal: 3.3 V TTL output
- Distancia de Detección: 3-7M
- Rango de Detección: menor a 140°
- Método de activación: disparador no repetible / gatillo H repetible.
- Tiempo de Salida: 5-200 seg.
- Temperatura de Trabajo: -20/+80 °C

---

<b>Costo</b>	USD 2,5 \$
--------------	------------

---

## **Software**

Debido a que se requiere desarrollar un sistema que procese grandes volúmenes de datos (imágenes) y como consecuencia emitir una respuesta al usuario de forma rápida y precisa, es necesario contar con ciertas especificaciones en cuanto a lenguajes de programación, bibliotecas y set de datos para llevar a cabo el objetivo propuesto, las cuales se detallan a continuación:

### *Lenguajes de Programación*

La elección del lenguaje y recursos de programación van de la mano de las necesidades específicas del proyecto, debido al enfoque de uso de la inteligencia artificial y visión computacional algunos lenguajes y herramientas destacan unas más que otras gracias a su facilidad de instalación y aplicación práctica, por lo que a continuación se describen los lenguajes seleccionados para el desarrollo del objetivo de esta investigación que cumplen con esas características:

- **Python.** Es un lenguaje de programación interpretado y de código abierto, con énfasis en la legibilidad del código. Resulta una gran opción para la investigación gracias a su cualidad multiparadigma, es uno de los más utilizados en el desarrollo de programas de IA con un amplio soporte de comunidades de usuarios y una amplia colección de bibliotecas.
- **Bash.** Es una popular interfaz de usuario para la ejecución de línea de comandos basado en un lenguaje de scripting, específicamente para sistemas Unix, usado para ejecutar tareas mediante comandos específicos.
- **HTML, CSS y JavaScript.** Sin excepción representan la combinación ineludible para el desarrollo web de la interfaz gráfica que presentará los resultados del análisis al usuario.

### *Herramientas y Bibliotecas de Programación*

Dado los lenguajes de programación antes mencionados, se complementan con el uso de las siguientes herramientas y bibliotecas, algunas de uso general y otras enfocadas al campo de la visión computacional:

- **Visual Studio Code.** Se elige este entorno de desarrollo gracias a su licencia de código abierto, forma simple de instalación y compatibilidad con distintas plataformas y lenguajes, además de contar con un buen editor e integración con el intérprete de Python.
- **OpenCV.** Es una de las bibliotecas más importantes y conocidas en el área de Visión por Computador y Procesamiento de Imágenes, de fácil instalación y

multiplataforma, originalmente disponible para C/C++ cuenta con un wrapper para Python, que la hace ideal para el proyecto.

- **Tensorflow Lite.** Es un framework de código abierto de aprendizaje profundo para la inferencia en diversos dispositivos. Resultando ideal para la presente investigación, ya que permite crear aplicaciones en dispositivos móviles y pequeños integrados/microcontroladores con recursos limitados para realizar el trabajo requerido con IA.
- **MTCNN-opencv.** Es una biblioteca para la detección de rostros aplicando redes neuronales convolucionales (CNN) con OpenCV.
- **Sanic Framework.** Es un framework de desarrollo web y servidor web, elegido gracias a sus características en cuanto a ligereza, flexibilidad, rendimiento y escalabilidad, con un diseño de API simple e intuitivo, listo para producción.
- **Jinja2.** Es un lenguaje de plantillas para el renderizado de páginas HTML, de gran rendimiento y flexibilidad, permite escribir código similar a la sintaxis de Python.
- **Bulma.** Es una biblioteca CSS que provee estilos predefinidos.
- **Peewee.** Es un ORM (Object Relational Mapping) para bases de datos de tipo SQL, minimalista y ligero, pero a su vez cuenta con una amplia API que permite el fácil modelado de tablas, así como formas complejas, pero flexibles de realizar consultas a la base de datos.
- **Huey.** Es un administrador de tareas que permite fácilmente el control y ejecución de procesos de manera independiente, siendo ideal para la investigación gracias a que permite el desarrollo de tareas con ejecución programada, reintentando estas en caso de fallos.

- **Gpiozero.** Es una biblioteca en Python que permite la comunicación con el GPIO (General Purpose Input/Output) de Raspberry Pi, permitiendo rápidamente la conexión rápida de cualquier componente.
- **Picamera.** Es una biblioteca usada para controlar el módulo de cámara Raspberry Pi, con una interfaz basada en Python. Dispone de métodos para interactuar con la cámara, así como la fácil integración con OpenCV.
- **Pillow.** Es una biblioteca de imágenes de Python, la cual agrega capacidades de procesamiento de imágenes al intérprete de Python.

#### *Sistemas Operativos para Raspberry PI 3B+*

Existe diversos sistemas operativos especialmente diseñados para el uso en Raspberry Pi, entre los que destacan tenemos a Linux en su distribución Ubuntu:

- **Ubuntu Server 20.04 32 bits para ARM:** Ubuntu al estar certificado para Raspberry PI, incluye optimizaciones para el uso del hardware del microordenador, garantizando actualizaciones de seguridad durante gran parte de la vida útil del equipo.

#### *Base de Datos*

Dada la necesidad de acceder y almacenar datos de forma rápida y sin perjudicar la integridad de los datos y el rendimiento del sistema, se optó por la biblioteca **SQLite3**, el cual permite la gestión de bases de datos relacionales desde archivos, resultando ligero y de código abierto.

## **Etapas II: Diseño**

Dadas las especificaciones de hardware y software necesarias para el desarrollo, la misma comprende un proceso de ensamblaje y pruebas sobre el funcionamiento del prototipo, seguido de la fase de programación del sistema, que comprende diversas etapas como la comunicación con el hardware, el flujo, procesamiento y almacenamiento de los datos, así como la visualización de los resultados y su correspondiente envío de alertas.

En medio de estas etapas encontramos uno de los puntos más importantes que es la elección de los modelos en Deep Learning a utilizar, modelos que están comprendidos dentro de la visión computacional donde la mayoría de los más conocidos hacen uso de las redes neuronales convolucionales (CNN), tales como MobileNet, MTCNN y FaceNet, diseñadas para trabajar en la detección y reconocimiento de imágenes, extrayendo sus características principales para posteriormente realizar una evaluación y diferenciación de los resultados.

En este sentido, a continuación, se detalla el proceso de desarrollo del sistema de IA, describiendo los pasos en cada una de las etapas:

### **Diseño y construcción del hardware**

Una vez adquiridos los componentes de hardware necesarios para dar inicio al desarrollo del proyecto, se procede al armado del sistema de captura de imágenes, el cual es activado por medio del sensor de movimiento.

*Paso 1. Configuración y habilitación del módulo de placa de cámara Raspberry Pi*

En esta fase se procede a conectar en el puerto SCI la cámara Raspberry Ov5647 (figura 19), dicho puerto SCI se encuentra ubicado entre los puertos HDMI y el puerto jack, por lo que una vez conectado correctamente se debe habilitar por software desde el sistema operativo.

**Figura 19:**

*Kit Raspberry Pi 3B+.*

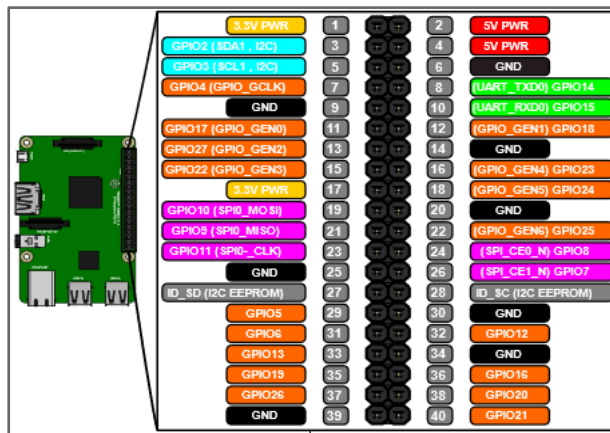


*Paso 2. Configuración de sensores*

Seguidamente, se conecta el sensor de movimiento vía GPIO (General Purpose Input/Output) los cuales son los contactos en forma de pines que permiten la comunicación de entrada y salida con otros dispositivos, tal como se muestra en la figura 20.

**Figura 20:**

*Diagrama de la GPIO.*

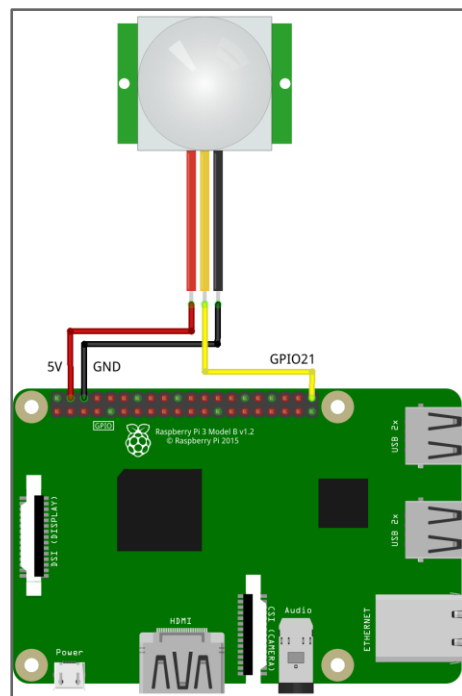


*Nota.* Adaptado de Device Plus, documentación oficial, 2020, DevicesPlus. (<https://www.deviceplus.com/raspberry-pi/using-pir-motion-sensors-with-the-raspberry-pi-and-python>).

En el caso del sensor de movimiento (PIR) se realizó la conexión de alimentación 5 V y tierra, finalmente el pin de comunicación de datos en el puerto GPIO 21. Debido a que hay 3 pines en el PIR estos deben estar etiquetados como Vcc, Gnd y Out, estas etiquetas por lo general están ocultas dentro de la tabla blanca que recubre el lente (ver figura 21).

## Figura 21:

*Cableado de un sensor PIR.*

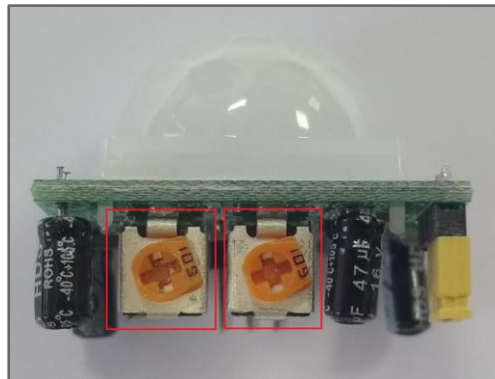


1. Como se muestra en la figura 21, el pin **Vcc** debe estar conectado a un pin de 5 V en la Raspberry Pi.
2. El pin **Gnd** del sensor PIR se puede conectar a cualquier pin de tierra de la Raspberry Pi.
3. Por último, el pin de salida debe estar conectado a cualquiera de los pines de datos **GPIO**.

Adicionalmente, se controló la sensibilidad del sensor y el tiempo de espera que el PIR indicará cuando se detecte movimiento, girando levemente los potenciómetros en sentido contrario a las agujas del reloj, siendo el de la derecha el que controla la sensibilidad y el de la izquierda el tiempo de espera (ver figura 22).

## Figura 22:

Control de sensibilidad (izquierda) y tiempo de espera (derecha) de un PIR.



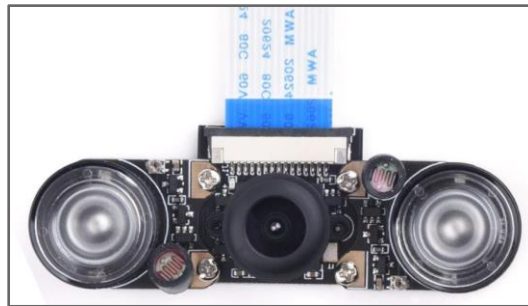
### *Paso 3. Módulos de iluminación*

Debido a la naturaleza de un sistema de seguridad, este debe seguir funcionando en diversos escenarios, por lo que es crucial que la operatividad no se vea afectada por la ausencia de luz, por ejemplo, en horas nocturnas.

Por ende, para evitar inconvenientes en este sentido se eligieron componentes de luz infrarroja como sistema de iluminación, debido a que la cámara capta correctamente ese rango de luz que no es reconocible a simple vista, haciéndolos ideales para capturar imágenes con poca luz visible o nocturnas. Estos componentes se alimentan directamente de la conexión con la cámara y cuentan con un sensor para detectar el nivel de luz disponible, activándose automáticamente al detectar escenarios de poca iluminación (ver figuras 23-24).

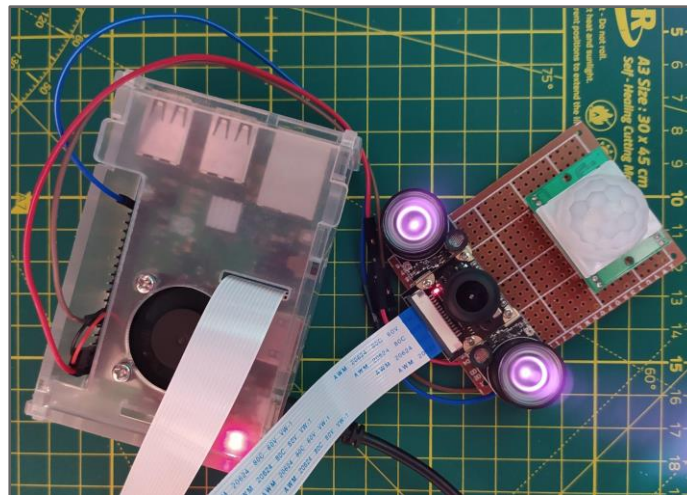
**Figura 23:**

*Módulos de iluminación.*



**Figura 24:**

*Prototipo funcional ensamblado.*



## **Diseño y construcción del software**

A continuación, se describen los pasos asociados a esta fase.

### *Paso 1. Instalación del sistema operativo para la placa Raspberry PI*

En algunos casos, al adquirir la placa, esta ya viene con la memoria microSD con el SO preinstalado, de lo contrario es posible descargarlo y descomprimirlo en una

memoria microSD que funcionará como disco duro, el cual es un proceso simple que dispone de la documentación oficial para lograrlo: <https://ubuntu.com/tutorials/how-to-install-ubuntu-on-your-raspberry-pi>

Al finalizar los pasos descritos en la documentación, se tendrá el sistema operativo en funcionamiento y con los accesos de usuario y administrador configurados para los siguientes pasos.

### *Paso 2. Habilitación de los componentes de hardware a través del software*

Una vez instalado el módulo de la cámara Raspberry Pi e integrado los sensores y módulo de iluminación, se procede a habilitar la cámara de Raspberry desde Ubuntu.

Dado que Raspberry Pi 3 B+ viene con un puerto para conectar la Raspberry Pi Camera, el problema está en que para Ubuntu el soporte de la cámara no está habilitado por defecto, por lo que para solucionar este problema es necesario realizar lo siguiente:

1. Primero se debe agregar el repositorio de la distribución oficial de Raspberry que está basado en Debian y es compatible con Ubuntu, para ello se debe crear y editar el archivo para la entrada del repositorio.

```
vim /etc/apt/sources.list.d/raspbian.list
```

2. Luego, se agrega la URL.

```
deb http://archive.raspberrypi.org/debian/ jessie main
```

3. Seguidamente, se agrega la clave para validar la firma de los paquetes.

```
apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --  
recv-keys 7FA3303E
```

4. Finalmente, se actualizan los paquetes.

```
apt-get update
```

5. Ya con el repositorio de *Raspbian* disponible, se procede a instalar las bibliotecas que hacen falta.

```
apt-get install libraspberrypi0 libraspberrypi-bin
```

6. Antes de poder hacer uso de la cámara, primero se tiene que habilitar, para ello se edita el archivo: **/boot/firmware/config.txt** y se agrega al final:

```
start_x=1  
gpu_mem=128
```

### *Paso 3. Instalación de picamera y prueba de captura de imagen*

En este punto se hace necesario comprobar que la cámara Raspberry Pi funciona correctamente, para esta tarea se deberá instalar la biblioteca de picamera y ejecutar una pequeña prueba.

```
apt-get install python3-picamera
```

De acuerdo con la documentación de picamera, se puede usar el siguiente código para capturar una imagen:

```
from time import sleep  
from picamera import PiCamera  
  
camera = PiCamera()  
camera.resolution = (1024, 768)  
camera.start_preview()  
# Camera warm-up time  
sleep(2)  
camera.capture('picamera.jpg')
```

Esto activa el módulo de la cámara de la Raspberry Pi, mostrando una vista previa de la imagen y luego de unos segundos toma una foto y la guarda en el directorio de trabajo actual configurado.

**Figura 25:**

*Imagen capturada usando picamera.*



*Paso 4. Captura y reconocimiento de imágenes*

En este apartado se hizo un desarrollo progresivo de características con procesos inicialmente simples como primer paso para su validación y que luego, al comprobar su correcto funcionamiento se fueron agregando mejoras con el objetivo de llegar a un sistema fácil de instalar y usar.

El punto de inicio es la captura de imágenes y su almacenamiento, seguido por la fase de detección y reconocimiento donde se aplican modelos de inferencia de Deep Learning basados en *redes neuronales convolucionales* (CNN) que permiten detectar

formas y puntos de referencia en imágenes, así es como *FaceNet* se utiliza para extraer características particulares del rostro de una persona; por lo que al momento del desarrollo, estos modelos elegidos son unos de los más efectivos para dispositivos móviles, embebidos o con recursos limitados.

Dado que la red neuronal debe ser entrenada para aprender a distinguir elementos dentro de la imagen, es necesario contar con un conjunto de datos homogéneos para que la red pueda discernir de manera más precisa a partir de patrones aprendidos.

**Captura de imágenes:** La captura de imágenes es crucial para el sistema, por lo que se inició con el desarrollo del sistema en la creación del servidor con el framework Sanic, definiendo un módulo dedicado al acceso y captura de imágenes.

```
def get_image(self, device_idx: int = None):
    device_control = self.get_device_control(device_idx)
    grab = None
    frame = None

    if device_control is not None:
        for i in range(self.DEVICE_AVOID_GRAB):
            grab = device_control.grab()

            _, frame = device_control.retrieve(grab)
            _, frame = cv.imencode(self.IMAGE_EXTENSION, cv.rotate(frame,
cv.ROTATE_180), self.ENCODE_PARAMETERS)

        return frame
```

En el código el método *get\_device\_control* se creó para seleccionar la cámara con OpenCV y una vez con el control del dispositivo hacer la captura y posterior conversión a mapa de bits, que luego podrá ser guardada y procesada.

**Captura de imagen controlada por el sensor:** En esta parte del proceso se lee el estado del sensor con Gpiozero, que tal como se visualiza en la figura 21, se eligió el

puerto 21 para la comunicación con el sensor, teniendo como resultado dos posibles estados, *activo* que empieza la captura de imágenes y el estado *inactivo* que lo detiene.

```
def sensor_activity(self):
    sensor = GPIONotionSensor(self.GPIO_PIN)
    sensor.when_motion = self.sensor_on
    sensor.when_no_motion = self.sensor_off
    sigwait([SIGTERM])
    logger.info('Stopping Motion Sensor')
```

Para que el sensor se apague dejando de capturar la imagen, se ajusta físicamente en el sensor PIR, así como la sensibilidad para activarse, ya que no es un valor que se controla por software.

Para el proceso de captura y guardado de la imagen en disco, se presentó un cuello de botella entre las imágenes capturadas y las que se podían guardar, esto se solucionó usando hilos y procesamiento asíncrono, lo que quiere decir que el proceso de captura y guardado van en procesos diferentes, no son bloqueantes entre sí, por lo que el hilo que captura la imagen se comunica con el otro hilo al que le pasa la imagen que después guarda en el directorio configurado, donde cada imagen es renombrada con un identificador único para evitar colisiones entre archivos.

```
def sensor_on(self):
    logger.info('SENSOR ACTIVATED!')

    now = datetime.now()
    timestamp = int(now.timestamp())
    self.SENSOR_ACTIVATION_ID = '{}_{}'.format(timestamp, str(uuid4()))

    self.CAPTURE_THREAD_EVENT = threading.Event()
    self.CAPTURE_THREAD = threading.Thread(
        target=self.CAMERA.continuous_capture_thread,
        args=[self.CAPTURE_THREAD_EVENT,
self.continuous_capture_callback]
    )
    self.CAPTURE_THREAD.start()

    def continuous_capture_callback(self, frame, frames_count):
        saved_image = self.STORAGE.save_image(
            frame,
            self.SENSOR_ACTIVATION_ID,
```

```

        'capture_{}'.format(frames_count),
        self.CAMERA.IMAGE_EXTENSION
    )
    saved_image["binary"] = io.BytesIO(frame.tobytes())

    # add DB record of captured image
    CaptureModel.insert(
        image_file=saved_image.get('image'),
        image_folder=saved_image.get('folder'),
        datetime=datetime.fromtimestamp(saved_image.get('timestamp'))
    ).execute()

    self.IMAGE_ANALYSIS.add_to_queue([saved_image])

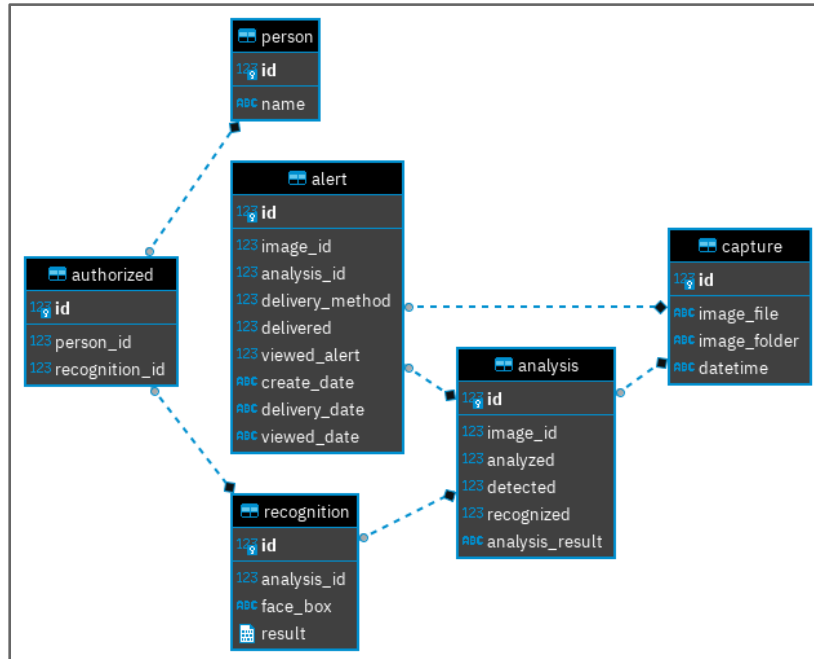
```

En el método *sensor\_on* se puede apreciar el momento que se crea un hilo para la captura de imágenes en *self.CAPTURE\_THREAD*, donde el parámetro *target* es un método similar al de *get\_image* del apartado de "Captura de imágenes" pero con la particularidad de estar adaptado para una ejecución en hilos, dicha ejecución recibe los argumentos definidos en *args*, el primero tiene como objetivo controlar el tiempo de vida del hilo, el segundo argumento es un método que se ejecuta con cada captura de imagen y que se encarga de almacenar en disco los datos así como agregar el registro en la base de datos, finalmente con el método *self.IMAGE\_ANALYSIS.add\_to\_queue* envía la imagen a la cola de otro hilo encargado de hacer un primer análisis de la imagen.

Como se mencionó anteriormente, para llevar un control de las imágenes, se generó una base de datos en SQLite donde se registra cada entrada, guardando la ubicación del archivo (imagen). Así mismo y con el avance de la investigación no solo fue necesario llevar el registro de las imágenes, sino que también se requirió agregar tablas para llevar el control de las imágenes analizadas, detecciones y reconocimientos encontrados, así como las alertas enviadas, quedando con el siguiente modelo de entidad y relación (ER) mostrado en la figura 26.

**Figura 26:**

*Estructura general de la base de datos. Modelo de ER.*



**Análisis de la imagen y detección:** El proceso de detección en las imágenes está basado en modelos de red neuronal convolucional (CNN), específicamente un modelo de MobileNet provisto por tensorflow que provee la capacidad para detectar diversas formas entre ellas de personas.

```
def detect(self, image_data):
    def detection(labels, interpreter):
        interpreter.allocate_tensors()
        threshold = 0.4
        input_details = interpreter.get_input_details()
        input_shape = input_details[0]['shape']
        height = input_shape[1]
        width = input_shape[2]

        self.interpreter.set_input_tensor(interpreter,
        image_data.copy().convert('RGB').resize((width, height)))
        interpreter.invoke()

        boxes = self.interpreter.get_output_tensor(interpreter, 0)
        classes = self.interpreter.get_output_tensor(interpreter, 1)
        scores = self.interpreter.get_output_tensor(interpreter, 2)
```

```

        count = int(self.interpreter.get_output_tensor(interpreter, 3))

        results = []
        for i in range(count):
            if scores[i] >= threshold:
                ymin, xmin, ymax, xmax = boxes[i]
                result = {
                    'bounding_box': [ymin * image_data.height, xmin *
image_data.width, ymax * image_data.height, xmax * image_data.width],
                    'class_id': labels[classes[i]],
                    'score': scores[i]
                }

                if result.get('class_id') in self.entities_allowed:
                    results.append(result)
        return results

    label_file_path = join_path('detection', 'labelmap.txt')
    model_file_path = join_path('detection', 'detect.tflite')
    return self.interpreter.detect_callback(detection, label_file_path,
model_file_path)

```

Del código se puede resumir que se realiza la configuración de Tensorflow con los parámetros de entrada acordes al modelo que se va a usar, en este caso ajustar el tamaño de la imagen al requerido por el modelo de inferencia. Finalmente, se analiza los resultados entregados por el intérprete y se compara la distancia definida para dar como válido que se ha hecho una detección.

**Reconocimiento:** Luego de haber hecho la detección y determinar que en la imagen se encuentra alguna persona, se procede a la detección de algún rostro para seguidamente hacer un reconocimiento basado en esa detección, teniendo en cuenta que para efectos del sistema se nombró como *autorizados* los rostros conocidos.

Es así, que por medio del siguiente código se establece el mecanismo de detección de rostros usando la biblioteca *MTCNN-opencv*:

```

# find faces in the image
image_array = cv.cvtColor(np.array(raw_image), cv.COLOR_BGR2RGB)
faces_info = self.face_detector.detect_faces(image_array)

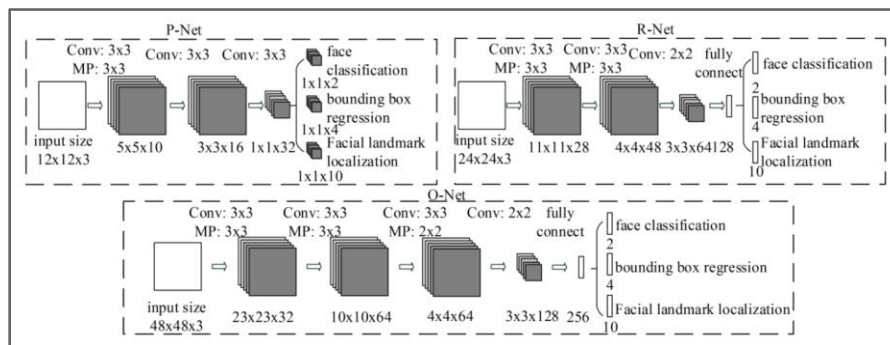
```

La biblioteca es una implementación de la red neuronal *Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks*, también conocida como *MTCNN*, el cual a diferencia del conocido mecanismo de detección *haar cascade*, *MTCNN* hace empleo de redes neuronales convolucionales, permitiendo tener un mejor rendimiento y precisión, así como la posibilidad de detectar otras características del rostro que no son posibles con *haar cascade*.

El modelo se denomina red multitarea (figura 27) porque lo componen tres modelos en cascadas independientes (P-Net, R-Net y O-Net) y están entrenados en tres tareas: clasificación de rostros, cuadro delimitador y localización de puntos de referencia faciales.

**Figura 27:**

*Arquitectura de MTCNN.*



*Nota.* Adaptado de MTCNN (Red de convolución en cascada multitarea), 2021, ICHI.PRO (<https://ichi.pro/es/mtcnn-red-de-convolucion-en-cascada-multitarea-186543706018893>).

Para hacer la detección de rostros, primero se debe convertir la imagen en una matriz numérica la cual permite igualar la imagen del rostro a una representación que luego es usado por el método *detect\_faces* de la biblioteca MTCNN-opencv.

Si como resultado se detectaron rostros, lo siguiente es iniciar el proceso de reconocimiento, como se muestra en el código a continuación:

```
for face_info in faces_info:
    [X, Y, W, H] = face_info['box']
    recognized_image =
self.detection_phase.recognize(Image.fromarray(image_array[Y:Y+H, X:X+W]))
```

Como en la imagen se puede detectar más de una persona y por ende más de un rostro, cada uno de los rostros detectados son llevados al método de reconocimiento como el siguiente:

```
def recognize(self, image_data):
    def recognition(interpreter):
        interpreter.allocate_tensors()
        input_details = interpreter.get_input_details()
        output_details = interpreter.get_output_details()
        input_shape = input_details[0]['shape']
        height = input_shape[1]
        width = input_shape[2]

        normalized = image_data.resize((width, height))
        normalized = (np.float32(normalized) - 127.5) / 127.5
        normalized = np.expand_dims(normalized, axis=0)

        interpreter.set_tensor(input_details[0]['index'], normalized)
        interpreter.invoke()

        output_data = interpreter.get_tensor(output_details[0]['index'])
        return np.squeeze(output_data)

    model_file_path = join_path('recognition', 'MobileFaceNet.tflite')
    return self.interpreter.recognize_callback(recognition,
model_file_path)
```

Como se puede apreciar, es algo similar al código de la primera fase de detección para encontrar formas de personas en las imágenes, esto es debido a que se trabaja con la misma biblioteca de *Tensorflow*, la diferencia en este caso es que no se busca encontrar formas específicas en la imagen, sino obtener una valoración única para cada imagen realizada con el modelo de red neuronal *FaceNet* y cuyo resultado nos servirá para hacer una comparación con otros análisis ya previamente guardados.

En este sentido, lo que nos va a permitir decidir si hemos logrado reconocer una imagen es la distancia numérica más corta encontrada entre la actual imagen y las que han sido previamente guardadas como reconocidas por el usuario (acción que está disponible desde la interfaz de usuario).

El proceso de cálculo para encontrar la distancia entre imágenes se puede hacer por diversas operaciones matemáticas, los resultados arrojados por el modelo de *FaceNet* son matrices multidimensionales y una de las maneras más conocidas es calcular la distancia euclidiana entre matrices donde se aplica una fórmula matemática por cada coordenada respectiva de cada matriz.

Ecuación para calcular la distancia euclidiana:

$$\sqrt{\square} \tag{5}$$

```
distance = round(cv.norm(recognized_image, stored_recognition), 3)
if distance <= 0.799:
    authorized_matched = authorized
```

En cuanto al código, para calcular la distancia entre dos matrices se utilizó el método de OpenCV llamado *norm*, por lo que una vez obtenida esa distancia se compara contra un valor arbitrario de referencia de cercanía, el cual se obtuvo a partir de las comparaciones efectuadas entre rostros similares.

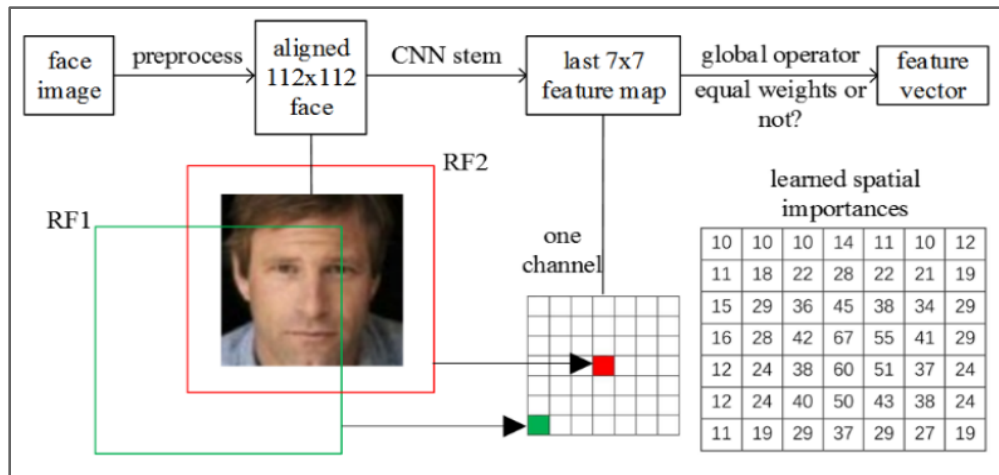
Para llegar a este punto, es necesario contar con un modelo entrenado de *FaceNet*, debido a que Tensorflow no cuenta con uno fue necesario usar una implementación independiente de la red neuronal y realizar el proceso de entrenamiento del modelo en un equipo aparte al destinado para el proyecto con suficiente poder de cómputo. A continuación, se describen los pasos llevados a cabo para la preparación y entrenamiento de la red neuronal.

**Elección del Modelo de Deep Learning:** Dado que dentro del deep learning existe una gran variedad de modelos para el reconocimiento facial, es prioritario elegir uno que se ajuste a los requisitos del hardware y reducir al mínimo la necesidad de nuevas dependencias, por ejemplo, servicios de APIs de productos de otras empresas que ofrecen la capacidad de reconocimiento facial como ML kit de Google o Amazon AWS Rekognition.

Entre las alternativas acordes a los requerimientos se encontraban: dlib, DeepFace y FaceNet. Y que, por una valoración rápida en implementaciones disponibles, precisiones reportadas y un mejor rendimiento en dispositivos con escasos recursos, se decidió usar el modelo de FaceNet, en una implementación llamada MobileFaceNet diseñada específicamente para un óptimo funcionamiento en dispositivos móviles (ver figura 28). La implementación elegida fue hecha utilizando Tensorflow el mismo framework elegido para el proyecto y liberado bajo licencia de código abierto, cuyo proyecto está disponible en: [https://github.com/sirius-ai/MobileFaceNet\\_TF](https://github.com/sirius-ai/MobileFaceNet_TF).

**Figura 28:**

*Arquitectura de MobileFaceNet.*



*Nota.* Adaptado de Arquitectura de MobileFaceNet (arXiv:1804.07573), 2021, Aixpaper (<http://static.aixpaper.com/pdf/0/3d/1804.07573.pdf>).

**Set de Datos:** Este paso está asociado a la recolección de imágenes previo al proceso de entrenamiento, es importante saber que entre más variedad de imágenes expuestas a diferentes tipos de iluminación y planos se tenga, mejor será el entrenamiento, esto con el objeto de que la red neuronal pueda reconocer características únicas y clasificarlas a partir de estas.

En este sentido, se cuenta con una biblioteca de imágenes obtenidas desde el sitio web <https://github.com/deepinsight/insightface> de nombre MS1M-ArcFace con más de 5 millones de imágenes disponibles para su uso de forma libre.

Los sets de datos para el entrenamiento están disponibles en un formato distinto al que reconoce Tensorflow, por lo que el proyecto de MobileFaceNet\_TF incluye una utilidad para hacer la conversión al formato reconocido por Tensorflow, el procedimiento y documentación están disponibles en el comando `utils/data_process.py` del proyecto.

**Entrenar la Red Neuronal:** Con los sets de datos preparados, iniciamos el entrenamiento con el comando `train_nets.py` del proyecto `MobileFaceNet_TF`, este proceso conlleva una gran cantidad de tiempo para ser completada, primero por la cantidad de parámetros posibles para configurar el entrenamiento de la red neuronal, tarea que requirió de varios ajustes para un correcto funcionamiento en el hardware destinado para tal fin y por último, el tiempo consumido por el propio proceso de cálculo realizado por la red neuronal sobre los sets de datos se requirió de varios días ininterrumpidos para ser completada.

Los parámetros relevantes en la ejecución fueron:

`--class_number 85742` valor necesario para identificar el set de datos usados.

`--train_batch_size` y `--test_batch_size`, hubo la necesidad de disminuir el tamaño al valor de 30, debido a que el entrenamiento se hizo empleando una GPU (tarjeta gráfica Nvidia GTX-2060 de 6gb) que cuenta con una memoria reducida a la esperada por los valores por defecto del comando de entrenamiento.

Fragmento de la salida del proceso de entrenamiento:

```
epoch 20, total_step 934500, epoch_step 24700, total loss 10.50 ,
inference loss 9.87, reg_loss 0.63, training accuracy 0.164062, rate 174.047
samples/sec
epoch 20, total_step 934600, epoch_step 24800, total loss 10.16 ,
inference loss 9.53, reg_loss 0.63, training accuracy 0.187500, rate 172.168
samples/sec
epoch 20, total_step 934700, epoch_step 24900, total loss 10.45 ,
inference loss 9.82, reg_loss 0.63, training accuracy 0.210938, rate 174.137
samples/sec
epoch 20, total_step 934800, epoch_step 25000, total loss 10.45 ,
inference loss 9.83, reg_loss 0.63, training accuracy 0.171875, rate 174.163
samples/sec
epoch 20, total_step 934900, epoch_step 25100, total loss 10.33 ,
inference loss 9.71, reg_loss 0.63, training accuracy 0.148438, rate 174.312
samples/sec
epoch 20, total_step 935000, epoch_step 25200, total loss 10.76 ,
inference loss 10.13, reg_loss 0.63, training accuracy 0.148438, rate 173.489
samples/sec

Iteration 935000 validating...
```

```

thresholds max: 1.34 <=> min: 1.1
---Total time 8.013s to evaluate 12000 images of lfw
- Accuracy: 0.993+-0.006
- Validation rate: 0.98300+-0.01169 @ FAR=0.00100
- FPR and TPR: 0.503 0.888
- Area Under Curve (AUC): 0.999
-----
thresholds max: 1.34 <=> min: 1.1
---Total time 8.054s to evaluate 12000 images of agedb_30
- Accuracy: 0.935+-0.014
- Validation rate: 0.52567+-0.05451 @ FAR=0.00133
- FPR and TPR: 0.533 0.779
- Area Under Curve (AUC): 0.981
-----

```

Una vez finalizado el entrenamiento, solo nos queda convertir al formato de Tensorflow Lite uno de los últimos modelos pre-entrenado resultantes en los checkpoint, primero se debe hacer una transformación al formato protobuf (.pb) con el comando *freeze\_graph.py* y luego usar el comando *tflite\_convert* para generar el modelo final que usaremos en el sistema.

Ejemplo del comando para convertir el modelo final:

```

tflite_convert \
--output_file MobileFaceNet.tflite \
--graph_def_file MobileFaceNet.pb \
--input_arrays "input" \
--input_shapes "1,112,112,3" \
--output_arrays embeddings \
--output_format Tflite

```

### *Paso 5. Envío de alertas*

Luego de la fase de intento de reconocimiento y de acuerdo a los resultados obtenidos, se procede a crear la alerta que será controlada por una tarea creada en *Huey*, por lo que si entre los reconocimientos hechos alguno no se encuentra entre los autorizados, se procede a crear la alerta, tal como se muestra en el código a continuación:

```

def create_alert(image_record, analysis_record, authorized_records):
    all_persons = list(map(lambda x: x.person if x.person is not None
else None, authorized_records))

    if len(all_persons) == 0 or None in all_persons:
        alert_record, created = AlertModel.get_or_create(
            image=image_record,
            analysis=analysis_record,
            delivery_method=AlertModel.DeliveryChoices.EMAIL
        )
    return alert_record

```

Finalmente, para enviar la alerta se implementó una tarea gestionada también por *Huey* que ejecuta de forma periódica (cada 60 segundos) la búsqueda de alertas creadas para imágenes analizadas que hayan tenido alguna detección, pero que no haya logrado hacer algún reconocimiento.

```

@task_queue.periodic_task(crontab(minute='*/1'), name='alert')
def send_alerts_task():
    with app_db:
        alerts = AlertModel.select().where(AlertModel.delivered ==
False)

        images = []
        image_folders = []
        image_count = 0

        for alert in alerts:
            image_folder = alert.image.image_folder
            if image_folder != '' and image_folder not in image_folders:
                image_folders.append(image_folder)

            if image_count < (MAX_IMAGE_COLLAGE_COLUMNS *
MAX_IMAGE_COLLAGE_ROWS):
                image = Image.open(join_path(
                    app_cfg.get('STORAGE_SETTINGS').get('DATA_FOLDER'),
app_cfg.get('STORAGE_SETTINGS').get('CAPTURE_FOLDER'),
                    alert.image.image_folder,
                    alert.image.image_file
                ))
                images.append(image.resize((IMAGE_COLLAGE_WIDTH,
IMAGE_COLLAGE_HEIGHT)))
            else:
                break
            image_count += 1

        if image_count > 0:
            image_count = 0
            image_length = len(images)
            collage_columns = image_length if image_length <
MAX_IMAGE_COLLAGE_COLUMNS else MAX_IMAGE_COLLAGE_COLUMNS

```

```

        collage_rows = 1 if image_length <=
MAX_IMAGE_COLLAGE_COLUMNS else
math.ceil(image_length/MAX_IMAGE_COLLAGE_COLUMNS)
        collage = Image.new(
            'RGB',
            (collage_columns * IMAGE_COLLAGE_WIDTH, collage_rows *
IMAGE_COLLAGE_HEIGHT),
            color=(255, 255, 255)
        )

        for i in range(0, collage_rows * IMAGE_COLLAGE_HEIGHT,
IMAGE_COLLAGE_HEIGHT):
            for j in range(0, collage_columns * IMAGE_COLLAGE_WIDTH,
IMAGE_COLLAGE_WIDTH):
                if image_count < image_length:
                    collage.paste(images[image_count], (j, i))
                    image_count += 1

        collage_image = BytesIO()
        collage.save(collage_image, format='JPEG')
        collage_image.seek(0)

        email_body = {
            'TEXT': 'Security Alerts\nReports:\n{0}',
            'HTML': """\
<html>
<body>
<div><h3>Security Alert</h3></div>
<div><strong>Reports:</strong></div>
<div>{0}</div>
</body>
</html>
"""\
        }

        report_links = []
        for folder in image_folders:
            report_links.append('{0}://{1}:{2}/reports/{3}'.format(
                app_cfg.get('SCHEMA'),
                app_cfg.get('HOST'),
                app_cfg.get('PORT'),
                folder
            ))
        email_body['TEXT'] =
email_body['TEXT'].format('\n'.join(report_links))
        email_body['HTML'] =
email_body['HTML'].format('\n'.join(list(map(lambda x: '<a href="{0}">VIEW
REPORT</a>'.format(x), report_links))))

        result = {'delivered': False}
        send_mail(
            app_cfg.get('EMAIL_ALERT_SETTINGS'),
            app_cfg.get('EMAIL_ALERT_SENDER'),
            app_cfg.get('EMAIL_ALERT_DESTINATION'),
            app_cfg.get('EMAIL_ALERT_SUBJECT'),
            email_body,
            collage_image,

```

```

        result
    )

    if result['delivered']:
        today =
datetime.fromtimestamp(datetime.now().timestamp())
        for alert in alerts:
            alert.delivered = True
            alert.delivery_date = today
            alert.save()

```

El código anterior muestra cómo se toman las imágenes, se arma el cuerpo de un correo, se realiza el envío a una dirección predeterminada y se actualiza en la base de datos el estado de todas las imágenes enviadas, esto último con el propósito de que las imágenes queden marcadas y no sean tomadas por el proceso nuevamente.

Para el envío de correos se usó la biblioteca SMTP integrada de Python con los valores de un servidor de prueba que será el encargado de entregar el correo.

```

def send_mail(smtp_config, sender, destination, subject, body,
collage_image, result):
    context = ssl.create_default_context()
    message = MIMEMultipart()
    message["Subject"] = subject
    message["From"] = sender
    message["To"] = ", ".join(destination)
    message.attach(MIMEText(body.get('TEXT', ''), 'plain'))
    message.attach(MIMEText(body.get('HTML', ''), 'html'))
    img = MIMEImage(collage_image.getvalue())
    img.add_header('Content-Disposition', 'attachment',
filename='motion_detected.jpg')
    message.attach(img)

    try:
        server = smtplib.SMTP(smtp_config.get('SMTP_ADDRESS'),
smtp_config.get('SMTP_PORT'))
        with server:
            if smtp_config.get('STARTTLS', False):
                server.starttls(context=context)
            server.login(smtp_config.get('ACCOUNT_ADDRESS'),
smtp_config.get('ACCOUNT_PASSWORD'))
            server.sendmail(
                smtp_config.get('ACCOUNT_ADDRESS'), destination,
message.as_string()
            )
            result['delivered'] = True
    except Exception as e:
        print(e)

```

El código anterior muestra cómo se arma el cuerpo del mensaje y la configuración de la llamada al servicio de envío de correos a través del protocolo SMTP.

### *Paso 6. Diseño de interfaz gráfica*

Se diseñó una interfaz web simple para la navegación de las imágenes capturadas, donde se puede visualizar el detalle general de cada alerta, así como la cantidad de imágenes capturadas, detecciones y reconocimientos realizados.

Basado principalmente en HTML, CSS y JavaScript, se utilizó el lenguaje de plantillas Jinja2, lo que permitió generar contenido web dinámico de forma mucho más simple, flexible y fácil de mantener a través de una gramática sencilla comprendida en `{%...%}` para identificar la estructura de control, `{{...}}` valor de la variable y para comentarios `{#...#}`, tal como se muestra en la siguiente estructura de código:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <link rel="stylesheet" href="/static/css/bulma.min.css">
    <link rel="stylesheet" href="/static/css/app.css">
    {% block head %}
      <title>{% block title %}{% endblock %} - IRS2ISA</title>
    {% endblock %}
  </head>
  <body>
    <div class="container">
      <nav class="navbar is-dark" role="navigation" aria-
label="main navigation">
        <div class="navbar-brand">
          <div class="navbar-item" style="background-color:
#7a7a7a;">
            <div style="text-align: center; padding-left:
0.5em;">IRS2ISA</div>
          </div>
        </div>
        <div class="navbar-menu">
```

```

        <div class="navbar-start">
          <a class="navbar-item" href="/">Home</a>
          <a class="navbar-item"
href="/reports">Reports</a>
        </div>
      </div>
    </nav>
  </div>

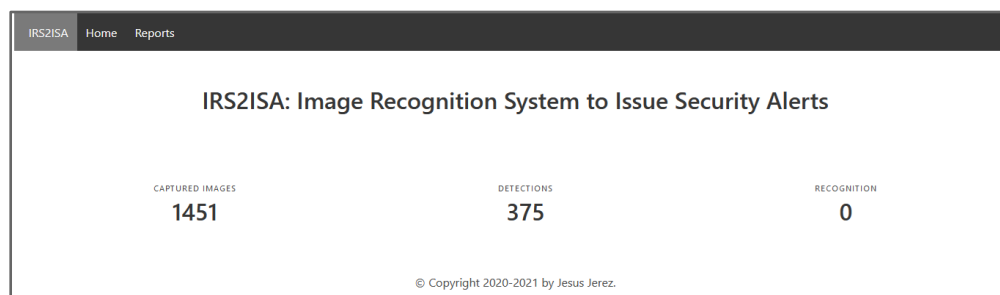
  {% block content %}{% endblock %}
<footer>
  <div class="content has-text-centered">
    {% block footer %}
      &copy; Copyright 2020-2021 by Jesus Jerez.
    {% endblock %}
  </div>
</footer>

{% block js %}{% endblock %}
</body>
</html>

```

**Figura 29:**

*Vista principal de la interfaz web.*



*Nota.* Página principal del ambiente de administración disponible para los usuarios, donde se visualizará el detalle de la cantidad de capturas, detecciones y reconocimientos llevados a cabo.

## Etapa III: Prueba

### Detección y Reconocimiento

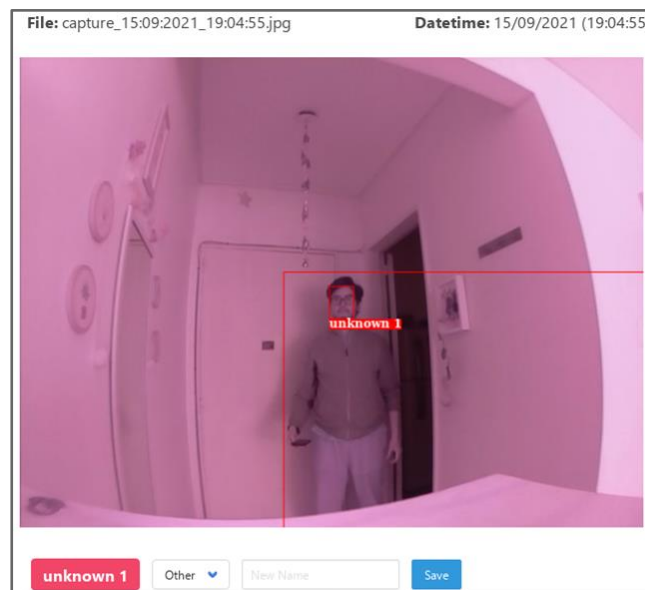
Una vez probado el modelo sobre el conjunto de imágenes reales, se compararon y contrastaron los resultados de las inferencias obtenidas, hallando así las

métricas cuyos valores de predicción permitirán saber si el modelo entrenado está generalizando bien.

En este sentido, con base en comparaciones de distintas fotos del rostro de la misma persona se pudo obtener un valor aceptable de la distancia euclidiana que será usado para las consiguientes comparaciones. De este modo, a través de la figura 30 se observa la detección del rostro sin reconocer, el cual se encuentra cubierto en un recuadro de color rojo, seguido de la etiqueta *unknown 1*, que lo identifica como *no autorizado o desconocido*.

### Figura 30:

*Detección del rostro.*



*Nota.* Detección y etiquetado automático, indicando *unknown* para usuarios desconocidos.

Dentro de la interfaz web y tal como se observa en la figura 31, se tendrá la oportunidad de asignar un nombre a los rostros desconocidos.

### Figura 31:

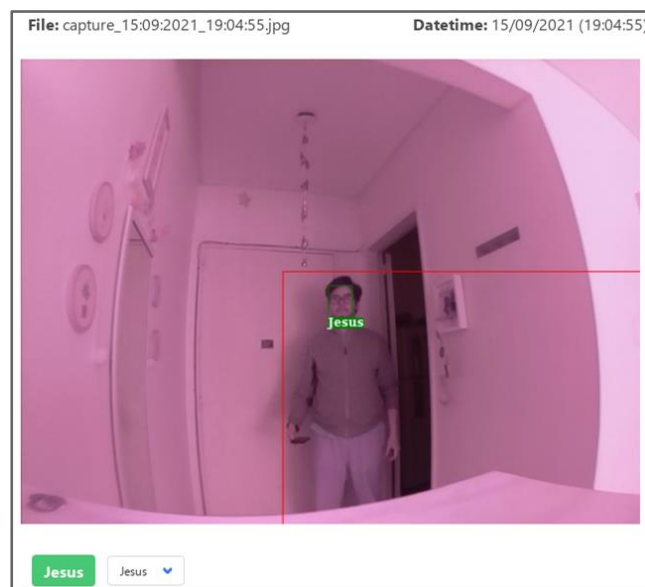
*Asignación de identidad.*



Una vez guardados, la imagen pasará a ser reconocida y se visualiza un recuadro de color verde sobre el rostro con una etiqueta que indica el nuevo nombre asignado, tal como se muestra en la figura 32.

### Figura 32:

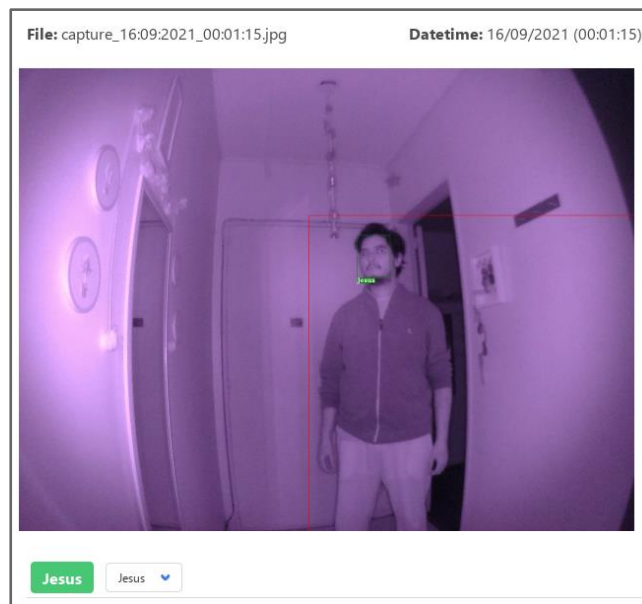
*Reconocimiento del rostro.*



Consecuentemente, en futuros intentos de reconocimiento el sistema colocará automáticamente la identidad del valor más cercano dada por la distancia euclidiana.

### Figura 33:

*Nuevo reconocimiento del rostro en ambiente nocturno.*



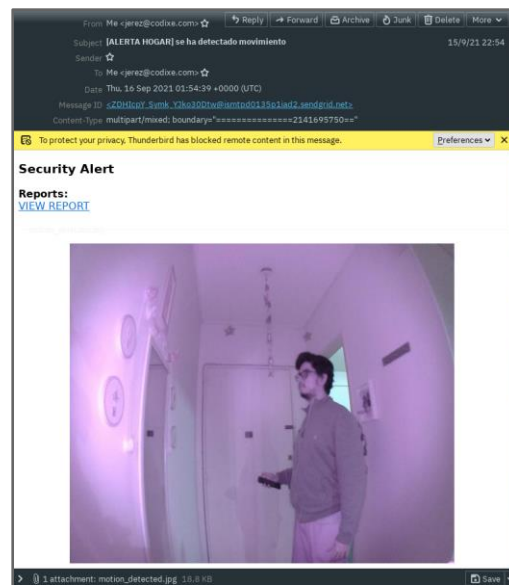
*Nota.* Reconocimiento del rostro y etiquetado automático, dentro de un entorno con ausencia de iluminación.

### Envío de Alertas

Una vez realizado el proceso de detección y reconocimiento de los rostros dentro de la imagen, se verifica que en dicha imagen todos los rostros hayan sido reconocidos, de lo contrario, la información obtenida del análisis es enviada a una dirección de correo electrónico definida por configuración, siendo el contenido un mensaje genérico con una galería de capturas como muestra y con el link (url) que le permitirá al usuario acceder a la vista web donde visualizará el detalle general del reporte con la detección encontrada por el sistema, tal como se observa en la figura 34.

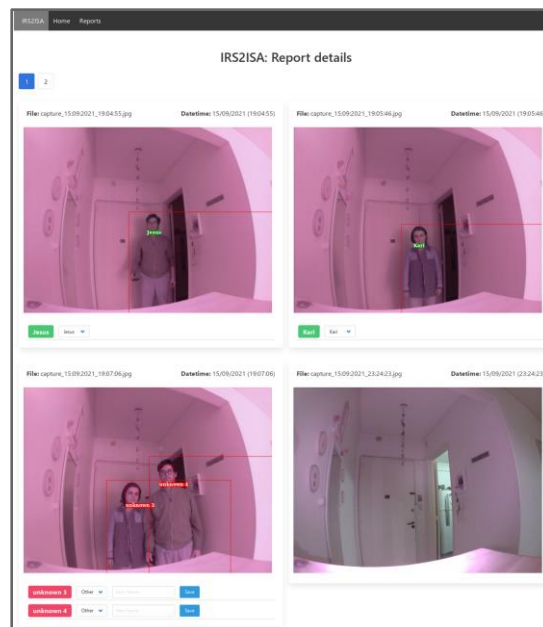
**Figura 34:**

*Mensaje de alerta recibido en el correo electrónico.*



**Figura 35:**

*Detalle del acceso al reporte enviado por correo.*



*Nota.* Detalle general de las detecciones y etiquetado, visible desde la interfaz web del usuario.

## **CAPÍTULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

El presente capítulo contempla el desarrollo de las conclusiones y recomendaciones finales dadas por el investigador, así como los resultados obtenidos en cada uno de los objetivos planteados en la investigación.

#### **Conclusiones**

Una vez realizada la investigación relativa al Desarrollo de un sistema de reconocimiento de imágenes para emitir alertas de seguridad, de acuerdo al primer objetivo propuesto orientado a *Identificar las herramientas de hardware y software a requerir para el desarrollo de la IA*, se concluye la validez de este objetivo analizando factibilidad técnica, económica y bibliográfica, ya que en relación con la adquisición de los elementos de hardware resultan accesibles desde cualquier tienda física o virtual, por el hecho de que son en su mayoría componentes básicos de electrónica con un bajo costo sin comprometer la calidad.

Lo que significó que no existieran problemas al momento de acceder a los componentes necesarios para construir el hardware de pequeñas dimensiones y acorde a los requerimientos para el procesamiento de datos, que, gracias a una comunidad amplia de desarrollo en el área de visión computacional, se obtuvo la documentación necesaria para armar toda la unidad de procesamiento y captura de imágenes sin muchas complicaciones técnicas.

Así mismo, con respecto al software, las especificaciones en cuanto a lenguajes de programación, bibliotecas y set de datos necesarios son todos con licencia libre, disponibles en forma gratuita. Destacando gracias a su facilidad de instalación y aplicación práctica, herramientas como Python, openCV y Huey por mencionar solo algunas, las cuales cuenta con un gran soporte por la comunidad de desarrollo, lo que facilitó la curva de aprendizaje al implementar cada uno de estos recursos, culminando en un software totalmente compatible con los componentes de hardware seleccionados, con un código depurado, entendible y ágil.

En este orden de ideas, con respecto al segundo objetivo relacionado con *Aplicar algoritmos inteligentes de reconocimiento de imágenes*, dado que el campo de visión computacional ofrece múltiples opciones en el ámbito de deep learning, se seleccionaron dos modelos de redes neuronales más populares en tareas de detección y reconocimiento de rostros dentro de imágenes, los cuales son: los modelos de red neuronal convolucional MTCNN y FaceNet.

Gracias a la vasta documentación y soporte técnico disponible se desarrollaron algoritmos para la detección y reconocimiento de rostros, haciendo uso de las redes neuronales MTCNN para la detección y FaceNet para la tarea de reconocimiento, lo cual permitió realizar análisis con gran poder de cálculo y con un buen rendimiento pese a los recursos limitados por el hardware.

En otro sentido, con respecto al tercer objetivo orientado a *Entrenar la IA en el reconocimiento de imágenes basado en patrones establecidos*, para obtener el modelo que permitiera estimar el valor de predicción basado en el entrenamiento de una red neuronal en el reconocimiento de imágenes, se efectuaron los ajustes en los

parámetros dados; sin embargo, fue un proceso largo y complejo que eventualmente puede tener resultados muy diversos dependiendo de los atributos elegidos y el conjunto de datos empleados para el entrenamiento, afectando la precisión que se obtenga en el análisis de las imágenes.

Así mismo, puede darse la ocasión que se quiera ampliar el conjunto de datos para el entrenamiento o utilizar uno totalmente diferente, la dificultad no sería menor, se podría pensar que para esos casos se pueden usar otros modelos más fáciles de expandir y entrenar, pero posiblemente sacrificando la precisión.

En este sentido y de acuerdo al cuarto objetivo de la investigación *Estudiar métricas que permitan comprobar el funcionamiento de la IA desarrollada*, actualmente existe una gran variedad de modelos de reconocimiento en deep learning que bajo el enfoque de visión computacional facilitan la detección de rostros dentro de una imagen, tal como se anticipó en citas previas, se emplearon para la presente investigación los modelos de redes neuronales CNN y FaceNet, siendo el primero elegido gracias a su precisión al momento de detectar rostros y puntos de referencia faciales en imágenes, así mismo FaceNet permitió evaluar las imágenes tomando sus características más importantes para realizar el reconocimiento del rostro.

Es así, que durante las fases iniciales del proceso de reconocimiento facial se hizo posible la detección de uno o varios rostros dentro de una imagen compuesta por una o más personas, para ello el proceso de detección inicialmente identifica el cuerpo humano disponible en la imagen, seguidamente detecta y extrae el rostro, colocando un recuadro rojo sobre las coordenadas del mismo, dichas coordenadas se conservan

para mejorar el rendimiento al evitar que la imagen sea evaluada cada vez que se quiera visualizar o analizar en fases posteriores.

Seguidamente, sobre el set de datos de pruebas, se analizó cada uno de los rostros previamente detectados, aplicando el modelo de FaceNet extrayendo las características más relevantes del rostro, las cuales son representadas en vectores de 128 números denominados incrustaciones, donde las incrustaciones corresponden a características determinantes de un rostro y que idealmente sean similares en diferentes imágenes de un mismo rostro.

Por lo que al momento de realizar las comparaciones y lograr encontrar la similitud debido a una corta distancia euclidiana, se puede inferir que se logró el reconocimiento de los rostros, disponiendo dentro de la imagen analizada un recuadro verde con una etiqueta con el nombre de la persona a la que pertenece el rostro y en el caso de que el rostro no sea identificado se incrusta un recuadro rojo con la etiqueta unknown (*desconocido*).

En función a los resultados obtenidos con las imágenes capturadas, donde fueron sometidas a diferentes tipos de iluminación y planos, apareciendo uno o varios sujetos dentro de la imagen, todo esto con el propósito de ofrecer una mayor variedad de imágenes para el proceso de pruebas del algoritmo de IA previamente entrenado, obteniendo resultados satisfactorios en el reconocimiento de las personas que ya habían sido previamente etiquetadas por el usuario del sistema como autorizadas.

Sin embargo, se encontraron situaciones donde el reconocimiento de rostros tuvo un peor rendimiento, ocurriendo en casos de imágenes tomadas en ambientes con poca luz, debido a que las fotos obtenidas son menos nítidas, no obstante, una vez

asignadas las identidades en imágenes con esas características la tasa de detecciones en similares condiciones aumentaba.

Aunado a lo anterior y como último objetivo propuesto, se planteó *Realizar las pruebas en el envío de alertas de seguridad en un ambiente real*, para esto se tomó como conjunto de datos para el proceso de pruebas imágenes capturadas dentro del domicilio del investigador, siendo esta la delimitación geográfica establecida para implementar el objetivo de esta investigación.

Tal como se describió en el capítulo destinado al diseño de la propuesta, la alerta se genera solo en los casos en que no se reconozca el rostro detectado, enviando un email al usuario del sistema con la información de la detección con un link que le permitirá acceder a la interfaz gráfica, en la cual podrá visualizar el detalle general de la imagen y asignar un nombre para que posteriormente sea identificado con este por la red neuronal.

Finalmente, se siguieron todos los estándares establecidos en cuanto a recursos de hardware, software y metodología aplicada, siendo la accesibilidad, bajo coste e integridad en los resultados cualidades indiscutibles que persigue la presente investigación, quedando altamente satisfechos con los resultados obtenidos en cada uno de objetivos propuesto por la investigación científica que planteó el desarrollo de un sistema de reconocimiento de imágenes para emitir alertas de seguridad, quedando como referencia para futuras investigaciones dentro del campo de visión computacional bajo el modelo de deep learning.

## Recomendaciones

Una vez cumplidos los objetivos propuestos en la investigación y descrito los resultados obtenidos, se procede a plantear algunas recomendaciones que el investigador considera necesarias para desarrollos futuros donde se tome como referencia la presente investigación:

- Agregar una capa nueva para hacer reconocimientos mediante el perfil del rostro.
- Para la interfaz web diseñada para el usuario, se recomienda agregar un mecanismo de autenticación para el acceso autorizado.
- Al ser un sistema totalmente dependiente del análisis de imágenes, el mismo puede ser engañado (ej. portando máscaras con imagen de otro rostro, algún dispositivo que proyecta imágenes de rostros conocidos), por lo que sería ideal contar con un sistema biométrico adicional, como por ejemplo un sistema de reconocimiento por voz.
- Agregar otros mecanismos adicionales para las alertas, como envío de SMS, llamadas telefónicas o mensajes por redes sociales.
- Agregar integración con **IoT** (*Internet de las cosas*) y programar acciones predeterminadas con base en ciertos comportamientos, por ejemplo: encender luces, activar alarmas, bloquear puertas, entre otros.

## REFERENCIAS

- Alegre, E; Gonzalo, P y Escalera, A. (2016). Conceptos y métodos en Visión por computador. Grupo de Visión del Comité Español de Automática (CEA). España.
- Arias, F. (2012). El Proyecto de Investigación: Introducción a la metodología científica. 6.<sup>a</sup>. Edición. Editorial episteme. Caracas.
- Baena, G. (2014). Metodología de la investigación. 1.<sup>a</sup>. Edición. Grupo editorial Patria. México.
- Benítez, R; Escudero, G y Kanaan, S. (2014). Inteligencia artificial avanzada. Publicado por: Universitat Oberta de Catalunya (UOC). España.
- Berzal, F. (2018). Redes neuronales & Deep Learning. Edición independiente. Granada, España.
- Bostrom, N. (2016). Superinteligencia. Caminos, peligros, estrategias. 3.<sup>a</sup>. Edición en español. TELL Editorial S.L.
- Dadhich, A. (2018). Practical computer vision. Extract insightful information from images using Tensorflow, Keras, and OpenCV. Publicado por Packt Publishing Limited, Reino Unido.
- Davies, E.R. (2018). Computer Vision.Principles, Algorithms, Applications, Learning. Primera Edición. Editorial Elsevier. United Kingdom.
- Eckroth, J. (2018). Python Artificial Intelligence Projects for Beginners. Primera edición. Publicado por Packt Publishing Ltd.
- García, A. (2017). Inteligencia Artificial. Fundamentos, práctica y aplicaciones. 2.<sup>a</sup> Edición. Alfaomega Grupo Editor, S.A. de C.V.

- Gollapudi, S. (2019). Learn Computer Vision Using OpenCV. With Deep Learning CNNs and RNNs. Primera edición. Hyderabad, Telangana, India.
- Guido, S y Müller A. (2017). Introduction to Machine Learning with Python. Primera edición. Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 9547.
- Hurtado de B., J. (2010). Investigación Holística. Tercera edición. Fundación Sypal: Caracas.
- Hurtado de B., J. (2012). El Proyecto de Investigación Holística. Comprensión Holística de la metodología y la investigación. Séptima Edición. Ediciones Quirón. Fundación Sypal. Caracas, Venezuela.
- Kaplan, A y Haenlein, M. (2018). Siri, Siri in my Hand, who's the Fairest in the Land? On the Interpretations, Illustrations and Implications of Artificial Intelligence. Kelley School of Business, Indiana University. Published by Elsevier Inc.
- Kurzweil, R. (2013). Cómo crear una mente. El secreto del pensamiento humano. 2.<sup>a</sup>. Edición en español. PEARSON EDUCACIÓN, S.A.
- Ponce, P. (2011). Inteligencia artificial. Con aplicación a la ingeniería. Primera edición. Alfaomega Grupo Editor, S.A. de C.V., México.
- Rosebrock, A. (2017). Deep Learning for Computer Vision with Python. Primera Edición. Publicado por PyImageSearch.
- Russell, S., Norvig P. (2009). Inteligencia artificial. Un enfoque moderno. 3.<sup>a</sup>. Edición. Pearson Educación, S.A.

Russell, R. (2018). Machine Learning - Guía Paso a Paso Para Implementar Algoritmos. Edición en español. Publicado por Createspace Independent Publishing Platform.

Singh, H. (2009). Practical Machine Learning and Image Processing. Primera edición. Allahabad, Uttar Pradesh, India.

### Referencias Digitales

Bagnato, J. (29 de noviembre de 2018). Aprende Machine Learning. ¿Cómo funcionan las Convolutional Neural Networks? Visión por Ordenador. Aprende Machine Learning. Recuperado el 23 de julio del 2021 de <https://www.aprendemachinelearning.com/como-funcionan-las-convolutional-neural-networks-vision-por-ordenador/>

Bagnato, J. (24 de junio de 2020). Aprende Machine Learning. Detección de Objetos con Python. Aprende Machine Learning. Recuperado el 3 de agosto del 2021 de <https://www.aprendemachinelearning.com/deteccion-de-objetos-con-python-yolo-keras-tutorial/#more-7262>

Barrera, A. (s.f). Beneficios de los avances tecnológicos en la medicina. Nextu. Recuperado el 02 de junio de 2020 de <https://www.nextu.com/blog/beneficios-de-los-avances-tecnologicos-en-la-medicina>

Deviceplus. (26 de junio de 2020). Using PIR Motion Sensors With the Raspberry Pi and Python. Recuperado el 17 de junio de 2021 de

<https://www.deviceplus.com/raspberry-pi/using-pir-motion-sensors-with-the-raspberry-pi-and-python>

Doonamis. (7 de enero de 2015). Aplicaciones móviles con visión por ordenador CV.

Doonamis. Recuperado el 23 de julio de 2020 de

<https://www.doonamis.es/aplicaciones-moviles-vision-ordenador>

Dulčić, L (02 de noviembre de 2019). Face Recognition with FaceNet and MTCNN.

Arsfutura. Recuperado el 23 de julio de 2021 de

<https://arsfutura.com/magazine/face-recognition-with-facenet-and-mtcnn/>

DW Documental. (18 de abril del 2020). Cómo cambia nuestra sociedad la inteligencia artificial. YouTube. <https://www.youtube.com/watch?v=CM71Zu7E00E>

El mundo. (07 de mayo de 2018). Inteligencia artificial. Youtube.

<https://www.youtube.com/watch?v=NSf3o-wxtQ0>

Evolución de la tecnología. (07 de diciembre de 2021). En wikipedia

[https://es.wikipedia.org/wiki/Evoluci%C3%B3n\\_tecnol%C3%B3gica](https://es.wikipedia.org/wiki/Evoluci%C3%B3n_tecnol%C3%B3gica)

García, I. (2018). Métodos de visión por computador para la detección automática de líneas de cultivo curvas/rectas y malas hierbas en campos de maíz. Para optar al grado de Doctor. Universidad Complutense de Madrid. España. E-prints

Complutense. <https://eprints.ucm.es/49697/1/T40479.pdf>

HardZone. (25 de agosto de 2018). Cómo configurar y usar el módulo de cámara en la

Raspberry Pi 3 B +. Recuperado el 3 de junio del 2021 de

<https://hardzone.es/2018/08/25/como-configurar-modulo-camara-raspberry-pi-3/>

Ibarra, E. (11 de febrero de 2017). Artículo científico. Generación de dataset para

problema de visión computarizada. Universidad del Tecnológico de Monterrey.

México. Blog, Inteligencia Artificial ITESM CQ. <https://medium.com/inteligencia-artificial-itesm-cq/generaci%C3%B3n-de-dataset-para-problema-de-visi%C3%B3n-computarizada-a90c77a0dc9a>

Inteligencia Artificial ITESM CQ. (11 de febrero de 2017). Machine Learning y sus aplicaciones. Medium. Recuperado el 22 de mayo de 2020 de <https://medium.com/inteligencia-artificial-itesm-cq/generaci%C3%B3n-de-dataset-para-problema-de-visi%C3%B3n-computarizada-a90c77a0dc9a>

Invertia & El español. (21 de noviembre de 2019). Inteligencia artificial: EEUU domina, China desafía y Europa despierta. Invertia. [https://www.elespanol.com/invertia/disruptores-innovadores/opinion/20191121/inteligencia-artificial-eeuu-dominachina-desafia-europa-despierta/446075395\\_12.html](https://www.elespanol.com/invertia/disruptores-innovadores/opinion/20191121/inteligencia-artificial-eeuu-dominachina-desafia-europa-despierta/446075395_12.html)

Jerez, J. (22 de octubre de 2020). Habilitando cámara de Raspberry en Ubuntu. Recuperado el 17 de junio del 2021 de [https://jerez.link/blog/post/habilitando\\_camara\\_de\\_raspberry\\_en\\_ubuntu/](https://jerez.link/blog/post/habilitando_camara_de_raspberry_en_ubuntu/)

López, G. (2016). Sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos en las áreas de almacenamiento de las Pymes. Para optar al título de Ingeniero en Electrónica y Comunicaciones. Universidad Técnica de Ambato, Repositorio Digital.

[http://repositorio.uta.edu.ec/jspui/bitstream/123456789/23065/1/Tesis\\_t1117ec.pdf](http://repositorio.uta.edu.ec/jspui/bitstream/123456789/23065/1/Tesis_t1117ec.pdf)

f

Robotics Back-End. (s.f). Raspberry Pi – Take a Picture With the Pi Camera Module.

The Robotics Back-End. Recuperado el 3 de julio de 2021 de

<https://roboticsbackend.com/raspberry-pi-camera-take-picture/>

Robotics Back-End. Configuración: cámara Raspberry Pi, biblioteca picamera, Python3.

Recuperado el 2 de junio del 2021 de [https://roboticsbackend.com/raspberry-pi-](https://roboticsbackend.com/raspberry-pi-camera-picamera-python-library)

[camera-picamera-python-library](https://roboticsbackend.com/raspberry-pi-camera-picamera-python-library)

Roman, V. (18 de febrero de 2019). Machine Learning: Cómo Desarrollar un Modelo

desde cero. Medium. Recuperado 03 de agosto de 2021 de

[https://medium.com/datos-y-ciencia/machine-learning-c%C3%B3mo-desarrollar-](https://medium.com/datos-y-ciencia/machine-learning-c%C3%B3mo-desarrollar-un-modelo-desde-cero-cc17654f0d48)

[un-modelo-desde-cero-cc17654f0d48](https://medium.com/datos-y-ciencia/machine-learning-c%C3%B3mo-desarrollar-un-modelo-desde-cero-cc17654f0d48)

Rosebrock, A. (30 de marzo de 2015). Accessing the Raspberry Pi Camera with

OpenCV and Python. Pyimagesearch. Recuperado el 2 de junio del 2021 de

[https://www.pyimagesearch.com/2015/03/30/accessing-the-raspberry-pi-camera-](https://www.pyimagesearch.com/2015/03/30/accessing-the-raspberry-pi-camera-with-opencv-and-python)

[with-opencv-and-python](https://www.pyimagesearch.com/2015/03/30/accessing-the-raspberry-pi-camera-with-opencv-and-python)

Sis, F. (30 de marzo de 2019). Definición de la tecnología. USAC.

[https://tecnologiaaplicadalfsl.blogspot.com/2019/03/definicion-de-tecnologia-](https://tecnologiaaplicadalfsl.blogspot.com/2019/03/definicion-de-tecnologia-segun-los.html)

[segun-los.html](https://tecnologiaaplicadalfsl.blogspot.com/2019/03/definicion-de-tecnologia-segun-los.html)

Sucar, L.E y Gómez, G. (s.f). Introducción a la Visión Computacional. CONACYT.

<https://ccc.inaoep.mx/~esucar/Clases-ia/Laminas2014/intro-vision.pdf>

Synnex Westcon-Comstor. (29 de diciembre de 2017). ¿Qué es la visión

computacional?. Blog México SYNEX Westcon-Comstor.

<https://blogmexico.comstor.com/que-es-la-vision-computacional>

TecnoSeguro, Magazín digital. (21 de mayo de 2019). ¿Qué tan avanzada está la inteligencia artificial en el campo de la seguridad?. Tecnoseguro.

<https://www.tecnoseguro.com/analisis/avanzada-inteligencia-artificial-seguridad>

Tensorflow. (27 de noviembre de 2019). Introducción a las redes neuronales convolucionales (ML Zero to Hero, parte 3). Youtube.

<https://www.youtube.com/watch?v=mO1fNuOhjtw>

Vázquez, C. (s.f). Aplicación de la inteligencia artificial. Visión computacional. Avansis.

<https://www.avansis.es/inteligencia-artificial/aplicaciones-de-la-ia-vision-computacional>