



**UNIVERSIDAD NACIONAL DE  
FACULTAD INGENIERÍA  
CARRERA TELECOMUNICACIONES**

**Implementación de un sistema de control de acceso de personal  
utilizando inteligencia artificial para la Dirección de Tecnologías de  
Información y Comunicación de la UNACH**

**Trabajo de Titulación para optar al título de Ingeniero en  
Telecomunicaciones**

**Autor:**

**Sampedro Altamirano, Stalyn Enrique**

**Tutor:**

**Mgs. Alejandra Del Pilar Pozo Jara**

**Riobamba, Ecuador. 2024**

## DECLARATORIA DE AUTORÍA

Yo, Stalyn Enrique Sampedro Altamirano, con cédula de ciudadanía 160046998-3, autor (a) (s) del trabajo de investigación titulado: Implementación de un sistema de control de acceso de personal utilizando inteligencia artificial para la Dirección de Tecnologías de Información y Comunicación de la UNACH, certifico que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de mí exclusiva responsabilidad.

Asimismo, cedo a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autor (a) de la obra referida, será de mi entera responsabilidad; librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, 8 de febrero de 2024.



---

Stalyn Enrique Sampedro Altamirano

C.I: 160046998-3

## **DICTAMEN FAVORABLE DEL PROFESOR TUTOR**

Quien suscribe, Alejandra del Pilar Pozo Jara catedrático adscrito a la Facultad de Ingeniería por medio del presente documento certifico haber asesorado y revisado el desarrollo del trabajo de investigación titulado: Implementación de un sistema de control de acceso de personal utilizando inteligencia artificial para la Dirección de Tecnologías de Información y Comunicación de la UNACH, bajo la autoría de Stalyn Enrique Sampedro Altamirano; por lo que se autoriza ejecutar los trámites legales para su sustentación.

Es todo cuanto informar en honor a la verdad; en Riobamba, a los 10 días del mes de enero de 2024.



Firmado electrónicamente por:  
**ALEJANDRA DEL PILAR  
POZO JARA**

---

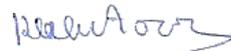
Mgs. Alejandra del Pilar Pozo Jara

## CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación “Implementación de un sistema de control de acceso de personal utilizando inteligencia artificial para la Dirección de Tecnologías de Información y Comunicación de la UNACH”, presentado por Stalyn Enrique Sampedro Altamirano, con cédula de identidad número 160046998-3, bajo la tutoría de Mgs. Alejandra del Pilar Pozo Jara; certificamos que recomendamos la APROBACIÓN de este con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba 8 de febrero de 2024.

Mgs. Klever Hernán Tores Rodríguez  
PRESIDENTE DEL TRIBUNAL DE GRADO



---

Msc. Deysi Vilma Inca Balseca  
MIEMBRO DEL TRIBUNAL DE GRADO



---

PhD. Yesenia Elizabeth Cevallos Villacrés  
MIEMBRO DEL TRIBUNAL DE GRADO



---

# CERTIFICADO ANTIPLAGIO



Dirección  
Académica  
VICERRECTORADO ACADÉMICO

*en movimiento*



UNACH-RGF-01-04-08.15  
VERSIÓN 01: 06-09-2021

## CERTIFICACIÓN

Que, **SAMPEDRO ALTAMIRANO STALYN ENRIQUE** con CC: **160046998-3**, estudiante de la Carrera **TELECOMUNICACIONES**, Facultad de **INGENIERÍA**; ha trabajado bajo mi tutoría el trabajo de investigación titulado "**Implementación de un sistema de control de acceso de personal utilizando inteligencia artificial para la Dirección de Tecnologías de Información y Comunicación de la UNACH**", cumple con el **1 %**, de acuerdo al reporte del sistema Anti plagio **TURNITIN**, porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 29 de enero de 2024.



Firmado electrónicamente por:  
**ALEJANDRA DEL PILAR  
POZO JARA**

Mgs. Alejandra del Pilar Pozo Jara

TUTOR

## **DEDICATORIA**

A mi madre Nelly Altamirano, que ha estado presente en toda mi etapa universitaria con sus cuidados y anhelos de un futuro próspero.

A mi padre Marcelo Sampedro, que siempre encontró un camino para que no me falte herramientas y un ejemplo de que caminando se llega a algún lugar.

A mi hermano Marcelo Sampedro, que ha sido una inspiración en todas las situaciones de la vida.

## **AGRADECIMIENTO**

A mi familia, que me demostraron que con amor, paciencia y dedicación se pueden conseguir todos los sueños de vida.

Agradezco a las personas con las que he compartido mis días en la universidad, a mis amigos y compañeros por siempre estar conmigo, además de compartir la dicha y aventuras de la juventud.

Como estudiante de educación pública, siempre estaré agradecido por la formación académica que me brindó la Universidad Nacional de Chimborazo.

Agradezco a las personas con las que compartí años y sueños durante la formación académica, a los que les deseo los mejores deseos, a los corazones de Nicole Benalcázar, Alejandro Cueva, Frans Macas y Roberto Calderón.

# ÍNDICE GENERAL

DECLARATORIA DE AUTORÍA	
DICTAMEN FAVORABLE DEL PROFESOR TUTOR	
CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL	
CERTIFICADO ANTIPLAGIO	
DEDICATORIA	
AGRADECIMIENTO	
ÍNDICE GENERAL	
ÍNDICE DE TABLAS	
ÍNDICE DE FIGURAS	
RESUMEN	
ABSTRACT	
CAPÍTULO I.....	16
1. INTRODUCCION.....	16
1.1 Antecedentes.....	16
1.2 Problema.....	17
1.3 Justificación.....	17
1.4 Objetivos.....	18
1.4.1 General.....	18
1.4.2 Específicos.....	18
CAPÍTULO II.....	19
2. MARCO TEÓRICO.....	19
2.1 Estado del Arte.....	19
2.2 Marco Teórico.....	20
2.2.1 Sistema de reconocimiento facial.....	20
2.2.2 Inteligencia artificial.....	23
2.2.3 Procesamiento de imágenes utilizando Python.....	32
CAPÍTULO III.....	33
3. METODOLOGIA.....	33
3.1 Tipo de Investigación.....	33
3.2 Diseño de Investigación.....	33
3.3 Técnicas de recolección de Datos.....	42
3.4 Población de estudio y de muestra.....	43

3.4.1	Población .....	43
3.4.2	Muestra .....	43
3.4.3	Variables .....	43
3.5	Hipótesis.....	43
3.5.1	Hipótesis Nula ( <b>H0</b> ) .....	43
3.5.2	Hipótesis Alternativa ( <b>H1</b> ).....	44
3.6	Métodos de análisis, y procesamiento de datos.....	44
CAPÍTULO IV .....		45
4.	RESULTADOS Y DISCUSIÓN .....	45
4.1	Resultados .....	45
4.1.1	Arquitectura del sistema de control de acceso .....	45
4.1.2	Gráfico de exactitud y pérdidas de la Red Neuronal Convolutiva implementada.....	46
4.1.3	Estudio de porcentaje de aciertos .....	46
4.1.4	Estudio de relación entre porcentaje de aciertos y la variable distancias .....	52
4.2	Discusión.....	55
CAPÍTULO V .....		57
5.	CONCLUSIONES y RECOMENDACIONES .....	57
5.1	Conclusiones .....	57
5.2	Recomendaciones.....	57
6.	BIBLIOGRAFÍA .....	59
7.	ANEXOS .....	62
7.1	Anexo 1: Interfaces de la aplicación .....	62
7.2	Anexo 2: Evidencias de la experimentación .....	64
7.3	Anexo 3: Compromiso de emitido por la Dirección de Tecnologías de la Información y Comunicación.....	65
7.4	Anexo 4: Capturas de la aplicación Lux Meter.....	66
7.5	Anexo 5: Datos recogidos en la experimentación .....	66

## ÍNDICE DE TABLAS

Tabla 1: Variables independientes.....	43
Tabla 2: Variable dependiente.....	43
Tabla 3: Porcentaje de aciertos del total de datos obtenidos en la experimentación.....	46
Tabla 4: Porcentaje de aciertos obtenidos en la variable distancia a 0.5m.....	47
Tabla 5: Porcentaje de aciertos obtenidos en la variable distancia a 0.75m.....	48
Tabla 6: Porcentaje de aciertos obtenidos en la variable distancia a 1m.....	48
Tabla 7: Porcentaje de aciertos obtenidos en la variable ángulo a 0 grados .....	49
Tabla 8: Porcentaje de aciertos obtenidos en la variable ángulo a 45 grados .....	50
Tabla 9: Porcentaje de aciertos obtenidos en la variable iluminación en iluminación baja	50
Tabla 10: Porcentaje de aciertos obtenidos en la variable iluminación en iluminación recomendada.....	51
Tabla 11: Descriptivos de la variable distancias en las distancias 0.5m, 0.75m y 1m .....	53
Tabla 12: Prueba de normalidad Kolmogorov-Smirnov en la variable distancia.....	53
Tabla 13: Rangos de la Prueba Kruskal-Wallis.....	55
Tabla 14: Estadísticos de prueba de la Prueba Kruskal-Wallis.....	55
Tabla 15: Base de datos de la variable distancias.....	75
Tabla 16: Base de datos de la variable iluminación .....	88
Tabla 17: Base de datos de la variable ángulo.....	101
Tabla 18: Base de datos del porcentaje de aciertos en las distancias 0.5 metros, 0.75 metros y 1 metro.....	103

## ÍNDICE DE FIGURAS

Figura 1: Módulos en un sistema de reconocimiento facial .....	20
Figura 2: Características de Haar utilizadas en detección facial con el método Haar Cascade .....	22
Figura 3: Etapas en una MTCNN[26] .....	22
Figura 4: Métodos de reconocimiento facial asistidos por computador[15] .....	23
Figura 5: Partes de una neurona biológica[33] .....	25
Figura 6: Estructura de una neurona artificial, que contiene la suma de sus entradas mientras se aplica una función de activación $f(x)$ . Donde $x_1, x_2, x_3$ son entradas. $w_1, w_2, w_3$ son pesos. $y_1$ es la salida[34].....	25
Figura 7: Funciones de activación utilizadas en las ANN. Donde $x$ es una variable, y $U$ es el umbral deseado[34]. .....	26
Figura 8: Capas de entrada, oculta y de salida en una Red Neuronal Artificial. Cada círculo representa un neurona artificial[30], [33]. .....	26
Figura 9 Procesos en un sistema de visión artificial[3]. .....	27
Figura 10: Proceso de Convolución [15]. .....	28
Figura 11: Capas en una Red Neuronal Convolutiva[15]. .....	28
Figura 12: Función Max Pooling en la Capa de Agrupación en una CNN[30]. .....	29
Figura 13: Ejemplo de la función Softmax[15]. .....	30
Figura 14: Estructura de una Red Neuronal Convolutiva[15]. .....	30
Figura 15: Arquitecturas de CNN actuales.....	31
Figura 16: Librerías en Python para visión artificial[15]. .....	32
Figura 17: Etapas en el desarrollo de la investigación .....	34
Figura 18: Funciones de la aplicación creada.....	35
Figura 19: Diagrama de flujo para el registro de Usuarios.....	36
Figura 20: Funciones para la obtención de video a tiempo real .....	36
Figura 21: Interfaz del Registro de Usuarios.....	37
Figura 22: Conjunto de datos para la clase "Desconocidos" .....	37
Figura 23: Diagrama de flujo para el entrenamiento del modelo .....	38
Figura 24: Aumento de datos aplicado al modelo .....	38
Figura 25: Arquitectura para el modelo de la red neuronal convolutiva .....	39
Figura 26: Interfaz de Entrenamiento del modelo .....	40
Figura 27: Compilación y entrenamiento del modelo .....	40

Figura 28: Diseño de la función reconocimiento facial.....	41
Figura 29: Interfaz de reconocimiento de usuarios .....	41
Figura 30: Diagrama de flujo para las funciones MySQL.....	42
Figura 31: Interfaz de descarga del registro de control de acceso.....	42
Figura 32: Software IBM SPSS statistics .....	44
Figura 33: Arquitectura del sistema de control de acceso .....	45
Figura 34: Gráfico de exactitud y pérdidas sobre el entrenamiento implementado .....	46
Figura 35: Diagrama de barras del porcentaje de aciertos del total de datos obtenidos en la experimentación.....	47
Figura 36: Diagrama de barras del porcentaje de aciertos obtenidos en la variable distancia a 0.5m .....	47
Figura 37: Diagrama de barras del porcentaje de aciertos obtenidos en la variable distancia a 0.75m .....	48
Figura 38: Diagrama de barras del porcentaje de aciertos obtenidos en la variable distancia a 1m .....	49
Figura 39: Diagrama de barras del porcentaje de aciertos obtenidos en la variable ángulo a 0 grados .....	49
Figura 40: Diagrama de barras del porcentaje de aciertos obtenidos en la variable ángulo a 45 grados .....	50
Figura 41: Diagrama de barras del porcentaje de aciertos obtenidos en la variable iluminación en iluminación baja .....	51
Figura 42: Diagrama de barras del porcentaje de aciertos obtenidos en la variable iluminación en iluminación recomendada.....	51
Figura 43: Histograma del porcentaje de aciertos a 0.5 metros.....	53
Figura 44: Histograma del porcentaje de aciertos a 0.75 metros.....	54
Figura 45: Histograma del porcentaje de aciertos a 1 metro .....	54
Figura 46: Diagrama de cajas de la variable distancia .....	54
Figura 47: Interfaz principal de la aplicación.....	62
Figura 48: Interfaz de la función de reconocimiento.....	62
Figura 49: Interfaz del registro de usuarios .....	63
Figura 50: Interfaz del entrenamiento del modelo.....	63
Figura 51: Interfaz de la eliminación de usuarios.....	63
Figura 52: Interfaz de la página web implementada.....	64
Figura 53: Registro de trabajadores de la DTIC en el sistema .....	64

Figura 54: Ejecución de las pruebas experimentales con los trabajadores de la DTIC.....	64
Figura 55: Implementación sugerida para la cámara del sistema.....	65
Figura 56: Iluminación para la experimentación en el rango recomendado para oficinas ..	66
Figura 57: Iluminación para la experimentación en el rango bajo para oficinas.....	66

## RESUMEN

El presente trabajo tuvo como objetivo el desarrollo de un sistema de control de acceso automático para el personal de la Dirección de Tecnologías de la Información y Comunicación (DTIC). Su enfoque fue usar inteligencia artificial para crear un sistema biométrico que registre la entrada y salida del personal a la bodega del departamento de redes de la DTIC, proporcionando así una capa de seguridad a los equipos de alto valor económico que se guardan en sus instalaciones.

Al no existir un sistema biométrico para autenticación del personal que trabaja en las instalaciones, la necesidad de una aplicación que cumpla con esta función se hizo urgente desde la Dirección de la DTIC. La solución propuesta se basa en el entrenamiento de redes neuronales convolucionales (CNN) para el reconocimiento de los trabajadores de la DTIC. Se realizó una investigación exhaustiva sobre el funcionamiento de estas redes y el estado del arte en sistemas biométricos similares. Este enfoque permitió incorporar las tecnologías más avanzadas en esta área, como el detector facial MTCNN para la localización de rostros, así como el diseño propio de una CNN profunda de 15 capas, que consta con las últimas recomendaciones de la industria como capas de aumento de datos, y de dropout, para alcanzar un aprendizaje con alto nivel de abstracción.

Adicionalmente, se desarrolló una página web que facilita la descarga remota de los registros generados por el sistema. Esta funcionalidad asegura que el director de la DTIC pueda acceder a estos registros de manera conveniente. En su implementación se utilizó el lenguaje de programación PHP para el manejo de los datos y HTML para el diseño de la página.

Se realizó una investigación experimental evaluando el rendimiento del sistema frente a las variables distancia del sujeto a la cámara, el ángulo del rostro respecto a la cámara, y la iluminación de la habitación. Se obtuvo porcentajes de acierto entre el 93% y el 98.7% en las diferentes configuraciones del experimento. Se demostró además que al variar la distancia en los valores acordados no afectada el porcentaje de acierto, con un 95% de confianza, y presentando una mediana del 100% en las tres distancias medidas.

**Palabras claves:** Red Neuronal Convolucional, Sistema Biométrico, Deep Learning, Visión Artificial.

## ABSTRACT

This work aimed at developing an automatic access control system for the personnel of the Information and Communication Technologies Directorate (DTIC). The approach involved using artificial intelligence to create a biometric system that records the entry and exit of personnel to the warehouse of the DTIC's network department, thereby providing a layer of security to the high-value equipment stored on its premises.

Since the facilities don't currently have a biometric system for personnel authentication, the DTIC Directorate realized how urgently they needed an application that would serve this purpose. Convolutional Neural Networks (CNN) are trained to recognize DTIC personnel as the core of the suggested solution. A thorough examination of these networks' operation and the state-of-the-art in analogous biometric systems was carried out. This methodology made it possible to create a custom deep CNN with 15 layers and to incorporate state-of-the-art technology in this sector, such as the MTCNN facial detector for face localization. To accomplish high-abstraction learning, the architecture incorporates dropout and data augmentation layers in accordance with the most recent industry recommendations.

Additionally, a website was developed to facilitate the remote download of records generated by the system. This functionality ensures that the DTIC director can conveniently access these records. PHP programming language was used for data management, and HTML was employed for the page design.

An experiment was done to evaluate the system's performance in terms of variables such as the subject's distance from the camera, the angle of the face relative to the camera, and room illumination. Accuracy rates ranged from 93% to 98.7% across various experiment settings. With a 95% confidence level, it was shown that altering the distance within agreed-upon values had no significant influence on accuracy, resulting in a median accuracy of 100% over the three examined distances.

**Keywords:** Convolutional Neural Network, Biometric System, Deep Learning, Computer Vision.



---

Revised by  
Mario N. Salazar  
CCL English Teacher

# CAPÍTULO I

## 1. INTRODUCCION

### 1.1 Antecedentes

Un método eficaz para fortalecer la seguridad de bienes materiales es la video vigilancia. En un estudio criminológico[1], donde se compararon los delitos registrados en zonas observadas mediante video con otras que no, se comprobó que la presencia de cámaras de vigilancia reduce el aumento de la delincuencia alrededor del área donde se han instalado. Además, se informó que la población expresó haberse generado un sentimiento de protección población. Una manera para aprovechar este efecto y proteger la integridad de los bienes materiales en la Universidad Nacional de Chimborazo (UNACH) sería la implementación de un sistema de registro automático del acceso mediante video, donde se tenga el registro de las personas que accedieron al área que se desea resguardar, especificando la fecha y hora del ingreso.

El registro de acceso de personal con video se puede implementar mediante reconocimiento facial[2], este ayudaría en la veracidad de la información de entrada a un área de interés, ya que en los métodos tradicionales de formularios físicos que son completados de manera escrita los datos se pueden modificar fácil y directamente del registro, lo que los convierte en sensibles a la suplantación de identidad [3]. Nuevas tecnologías basadas en biometría[4] ayudan a asegurar la validez en el reconocimiento de una persona, ya que utilizan sus rasgos físicos únicos. Una de estas tecnologías, es la visión artificial[5].

En la UNACH si bien han existido implementaciones de registros de personal usando datos biométricos, como escáner de huellas dactilares, técnicas como la visión artificial todavía no han sido exploradas, aunque estas pueden presentar beneficios en rapidez y precisión [6] con una implementación menos físicamente invasiva. La visión artificial consiste en varios procedimientos que involucran hardware y software que tienen el fin de capturar una imagen, procesarla resaltando sus detalles más característicos, y finalmente reconocer el objeto detectado e interpretarlo mediante una asignación entre otros objetos reconocidos[5]. En el caso de realizarse este proceso con datos biométricos de la cara, estos últimos pasos reconocerían al individuo, y se conocen formalmente como detección y reconocimiento facial[2].

El reconocimiento facial tiene como objetivo caracterizar a un individuo mediante las cualidades únicas de su rostro, usando para ello las distancias de su nariz, ojos, orejas, boca, entre otros; a lo largo del tiempo esta tarea se logró automatizar mediante algoritmos que reconocen las diferencias entre los rasgos faciales de cada persona[7]. En la actualidad existe una nueva tecnología que facilita la implementación de tanto la tarea de detección como la de reconocimiento facial, la inteligencia artificial (AI)[8]. La AI basa su funcionamiento en moldear la mente humana como un compuesto de redes neuronales, capaces de aprender mediante coeficientes de ponderación[8]. Varias investigaciones han dado resultados exitosos al utilizar la AI en aplicaciones de reconocimiento y registro facial [3], [9], [10] ,

por lo que se propone entonces la implementación del registro automático de acceso, que reconozca apoyándose en tecnologías basadas en inteligencia artificial. Para el desarrollo de esta se propone utilizar modelos fundamentados en software libre y código abierto[11]. La ejecución del sistema de registro de acceso se propone en la bodega del departamento de redes de la Dirección de Tecnologías de la Información y Comunicación de la UNACH.

## **1.2 Problema**

La Dirección de Tecnologías de la Información y Comunicación de la Universidad Nacional de Chimborazo actualmente no cuenta con un registro automático para el acceso de su personal. En esta área existen equipos y bienes materiales de alto valor económico en la bodega que se encuentra en sus instalaciones, en el caso de la desaparición de alguno de estos, no se tendría una constancia de las personas que tuvieron contacto con los materiales antes del suceso. Por otro lado, los métodos tradicionales como el registro en papel de acceso son altamente sensibles a fraudes por suplantación de identidad, y a modificación del mismo.

## **1.3 Justificación**

Un sistema de registro automático de personal es necesario ya que en la Dirección de Tecnologías de la Información y Comunicación existen equipos y bienes de elevado valor económico en su bodega, este proveerá tanto la identidad del individuo, la fecha y la hora del ingreso. Estos datos son imprescindibles en el caso de existir algún problema con los materiales que se encuentra en el área. En el presente proyecto se propone el control de manera automática del registro de ingreso, mediante una cámara y el procesamiento de las imágenes captadas realizado por AI, que realizará tanto la detección facial como el reconocimiento del personal en una base de datos previamente establecida. La implementación exitosa del sistema propuesto en esta investigación en un ambiente controlado, como la Dirección de Tecnologías de la Información y Comunicación de la UNACH, puede servir además como ejemplo para la implementación de este en otras áreas de la universidad.

## **1.4 Objetivos**

### **1.4.1 General**

- Diseñar e implementar un prototipo para el control de acceso automático del personal de la Dirección de Tecnologías de la Información y Comunicación de la Universidad Nacional de Chimborazo utilizando reconocimiento facial con métodos de inteligencia artificial.

### **1.4.2 Específicos**

- Revisar el estado del arte sobre librerías de inteligencia artificial basadas en código abierto.
- Diseñar un sistema para el registro automático del acceso del personal.
- Implementar un prototipo para el control de acceso en áreas específicas, del personal de la Dirección de Tecnologías de la Información y Comunicación de la Universidad Nacional de Chimborazo, y un sitio web que permita la generación de reportes del sistema implementado.
- Verificar la implementación del proyecto mediante estadísticas de reconocimientos exitosos

## CAPÍTULO II

### 2. MARCO TEÓRICO

#### 2.1 Estado del Arte

En los últimos años se ha explorado la implementación del reconocimiento facial mediante procesos computacionales en donde se utiliza en la base del funcionamiento de sus algoritmos la inteligencia artificial. En su desarrollo se involucra diversos métodos tanto para el procesamiento de las imágenes y para el entrenamiento del modelo. Debido a esto, para la ejecución del proyecto actual se analiza en la presente sección un conjunto de trabajos investigativos, elegidos con una fecha de publicación de hasta cinco años de antigüedad a la redacción de este documento, cuyos métodos y resultados aporten al desarrollo de la investigación.

En un trabajo de grado [10] se diseñó un registro automático para la asistencia a clases en un Centro de Atención Sector Agropecuario. Su sistema utilizó una arquitectura modular, compuesta de seis módulos que realizan funciones de gestión de datos, capturas de imágenes, detección y reconocimiento de rostros, y también incluyó interfaces de usuario web y móvil. En el proyecto se implementaron varios métodos para detección de rostros, como MTCC, RetinaFace y SCRFD, y métodos de reconocimiento facial, como VGG-Face, FaceNet y ArtFace. Todos estos métodos se compararon en función a su exactitud, y la experimentación demostró que los métodos con mejor exactitud para la detección y reconocimiento fueron SCRFD-2.5GF y ArcFace con arquitectura CNN, respectivamente. La implementación del sistema durante un periodo de 10 logró una tasa de acierto del 99.81% en ambientes internos, y 99.36% en ambientes externos.

En otro estudio [12] se evaluó el rendimiento en función del tiempo de detección de facial de tres modelos que funcionan en computadores basados en CPU; DLIB, HAAR Cascade y MTCNN. Las pruebas utilizaron imágenes en diferentes resoluciones y tamaños, capturadas de varias cámaras y utilizando los modelos en sus configuraciones estándar. Se obtuvo que el algoritmo de HAAR Cascade tiene mayor rendimiento del grupo evaluado, por lo que es el que más rápido detectando rostros. Por otro lado, también se observó que su exactitud es baja a comparado a los modelos DLIB y MTCNN, siendo este último el más preciso.

En un trabajo investigativo [3], se propuso un registro de asistencia de alumnos por medio de reconocimiento facial. En este se comparó el desempeño en función al reconocimiento facial realizado por dos tipos redes neuronales, la red neuronal de Histograma de Gradientes Orientados (HOG) y la red neuronal de Convolución (CNN), mediante pruebas donde se varían la iluminación del lugar y la posición del objetivo con respecto a la cámara. La experimentación mostró que en un ambiente con iluminación natural la red HOG tiene una probabilidad de realizar el reconocimiento del 52%, mientras que la red CNN presenta el 92%. Por otro lado, si se aumenta la iluminación de manera artificial, HOG presenta un 48%, mientras que CNN el 98%. Con respecto al tiempo, HOG muestra sus reconocimientos en

un tiempo entre 2 y 3 segundos, mientras que CNN entre 39.64 y 41 segundos. Este estudio concluyó que, aunque la red CNN es mucho más exacta, la red HOG es más rápida.

En una investigación [13] se buscó mejorar la precisión de las redes neuronales convolucionales (CNN) en la tarea de reconocimiento facial. Para ello se manipuló parámetros como el número de capas de convolución, la función de activación utilizada, el valor de dropout y la utilización de un algoritmo de optimización, en el entrenamiento de la CNN. En la experimentación se utilizó un total de 600 imágenes de tres actrices chinas, y una configuración estándar del modelo con 8 capas en total, siendo 4 de convolución, 2 de pooling, una de conexión y una de salida. Se concluyó que los mejores resultados se tienen al aumentar 3 capas de convolución y una de pooling, al utilizar la función de activación de relu, un valor de dropout de 100%, y el algoritmo de optimización Adaptive Moment Estimation (Adam), pudiendo mejorar la precisión del reconocimiento del modelo base de un 68.85% a 79.41%.

En un artículo publicado [14], se desarrolló un sistema automático para el registro de asistencia a clase, mediante reconocimiento facial. Se implementó una aplicación móvil que registra nuevos usuarios, y guarda la asistencia mediante una fotografía. En este estudio se comparó el desempeño en el reconocimiento facial de tres algoritmos: Algoritmo K-nearest neighbor (KNN), Red Neuronal Convencional (CNN) y Support Vector Machine (SVM). Se reportó que el modelo que presentaba una mejor exactitud era el CNN, con un 99.3%, seguido de KNN con 98.6% y SVM con 88%.

## 2.2 Marco Teórico

### 2.2.1 Sistema de reconocimiento facial

Es un tipo de aplicación biométrica, pues utiliza las características físicas en el rostro de los individuos para identificarlos entre una base de datos ya establecida [6]. Los primeros usos de estos sistemas se datan en los años 60, donde se guardaban las coordenadas horizontales y verticales en rangos particulares, como ojos, orejas, nariz y boca, de una fotografía. Esta información se comparaba con registros de los individuos, y se proporcionaba la coincidencia más cercana en un proceso informático. Actualmente, gracias al incremento exponencial de la potencia computacional y a las investigaciones sobre redes neuronales artificiales que se han dado en los últimos años, el reconocimiento se suele implementar mediante inteligencia artificial [15].

#### 2.2.1.1 Módulos en un sistema de reconocimiento facial

Las funciones en un sistema de reconocimiento facial se pueden distinguir como cuatro módulos que operan secuencialmente[16]:

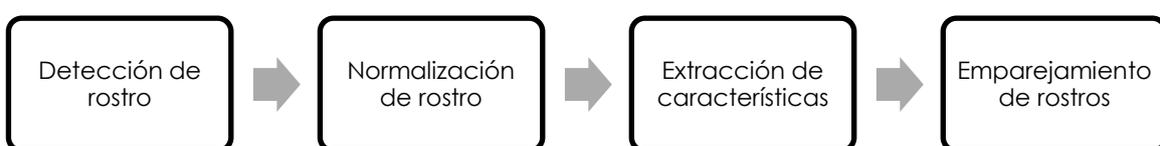


Figura 1: Módulos en un sistema de reconocimiento facial

#### **2.2.1.1.1 Detección de rostro**

La función del módulo de es separar el área donde se encuentra el rostro del resto de la imagen [16]. Para hacerlo se vale de algoritmos que detectan la presencia de rostros en la imagen o video [17].

#### **2.2.1.1.2 Normalización de rostro**

En este se normaliza geométrica y fotométricamente la imagen que contiene el rostro. El primero se consigue recortando todas las imágenes que contienen rostros a un tamaño estándar de píxeles, y el segundo convirtiendo las imágenes en escala de grises [16].

#### **2.2.1.1.3 Extracción de características**

Aquí se extrae información que va a ser útil para la distinción de rostros de diferentes personas[16]. Esta tarea se puede realizar de dos maneras, analizando todo el rostro como un solo conjunto, o sólo extrayendo información de regiones de interés como los ojos, cejas y boca [17].

#### **2.2.1.1.4 Emparejamiento de rostro**

En la etapa final se busca comparar un rostro de entrada con alguno de los registrados en la base de datos. Su objetivo es identificar si la persona está en el conjunto conocido [16].

### **2.2.1.2 Métodos de detección y localización de rostros en imágenes**

Encontrar los rostros en una imagen, o video, es necesario para la ejecución de los demás procesos en el sistema de reconocimiento facial. Algunos de los métodos más utilizados son:

#### **2.2.1.2.1 Detección basada en características locales (LBP)**

Este método consiste en dividir una imagen en varios bloques no superpuestos llamados celdas, para después extraer y concatenar diferentes características en un vector de características [18]. Para ello se forma una vecindad LBP de 3x3 píxeles, donde el valor del píxel en el centro se sustrae de cada píxel vecino. Si el resultado es positivo, se representa con un valor de 1, y en caso contrario 0. Con los resultados de cada operación se forma un número binario [19]. Al analizar el número en binario se pueden encontrar estructuras de bordes y esquinas. En el caso de detección facial, se utiliza para encontrar particularidades en el rostro. Este método suele ser utilizado por mostrar una invariabilidad frente a los cambios de iluminación [20].

#### **2.2.1.2.2 Haar Cascade**

Se utiliza para analizar la intensidad de píxeles en regiones específicas de una imagen en escala de grises [21]. En la detección de rostros, utiliza el criterio que existen ciertas regiones del rostro que tienen más intensidad que otras. Por ejemplo, la zona en los ojos suele ser más oscura que las mejillas, y la nariz que suele ser más brillante que las demás. Para detectar estas características, se colocan rectángulos en las regiones de interés, que suman el valor de los píxeles en su región, y se comparan la diferencia de intensidad que existen entre estas [19].

Los rectángulos usados se conocen como Haar features [22], y pueden tener las siguientes formas:



Figura 2: Características de Haar utilizadas en detección facial con el método Haar Cascade

### 2.2.1.2.3 MTCNN

Sus siglas hacen referencia a red neuronal convolucional en cascada (Multi-task Cascaded Convolutional Network), y esta funciona mediante tres redes neuronales convolucionales en la que cada entrada de una es la salida de la anterior. En su operación se distinguen tres etapas, correspondientes a cada red neuronal convolucional [23].

Antes que la imagen entre en la primera etapa, en este método se redimensiona la imagen a diferentes tamaños, para asegurarse de encontrar rostros que tengas tamaños diferentes [24]. Para la primera etapa, llamada P-Net, se utiliza un filtro que va recorriendo la imagen con el objetivo de detectar rostros empleando poca precisión en esta tarea. La red entrega las coordenadas de cajas que encierran los rostros en la imagen, conocidos como candidatos, donde muchos de estos serán incorrectos. La siguiente etapa, llamada R-Net, tiene como objetivo reducir el número de candidatos mediante mejoras en la delimitación de las cajas, y juntando candidatos superpuestos. Esta tiene como salida las coordenadas de los candidatos, un vector con puntos de referencias faciales, y un nivel de confianza. Finalmente, de los candidatos que queden de la R-net, pasan a una última O-Net que mejora aún más la delimitación, y entrega las coordenadas de la caja delimitadora, otras para los puntos de referencia faciales, y el nivel de confianza de cada caja [25].

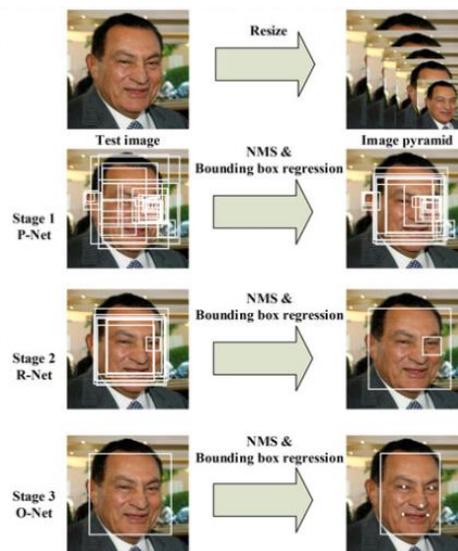


Figura 3: Etapas en una MTCNN[26]

El método MTCNN presenta una ventaja a otros métodos ya que puede reconocer si existen rostros aunque estos sean cubiertos parcialmente con objetos, o manos [24].

### 2.2.1.3 Métodos de Reconocimiento facial asistidos por computador

Una vez detectado el rostro, los métodos que existen para diferenciar un individuo de otro difieren en cómo extraen características, o información, de cada uno. Se pueden dividir los acercamientos en este enfoque en cuatro[15]:

Métodos holísticos	Métodos geométricos	Métodos Descriptores de Texturas Locales	Métodos de Deep Learning
<ul style="list-style-type: none"><li>• Utilizan la imagen completa del rostro, donde eliminan redundancias del conjunto de imágenes mediante descomposición de vectores.</li><li>• Algunos métodos son: Principal Component Analysis (Eigenfaces) y Linear Discriminative Analysis (Fisherfaces)</li></ul>	<ul style="list-style-type: none"><li>• Su base es localizar y focalizar espacios de dónde extraer características como distancias y ángulos entre ellos.</li><li>• Se suele utilizar regiones como los ojos, punta de la nariz, comisuras de la boca, entre otros.</li><li>• Algunos métodos son: Elastic Graph Matching.</li></ul>	<ul style="list-style-type: none"><li>• Extrae características basadas en el valor de los píxeles en zonas locales.</li><li>• El método LBP divide la imagen del rostro en diferentes regiones LBP, que combinan un histograma que se usa como descriptor facial.</li></ul>	<ul style="list-style-type: none"><li>• Utiliza Redes Neuronales para la extracción de características en la imagen.</li><li>• Algunas de las arquitecturas son: AlexNet, VGGNet, GoogLeNet, ResNet y SeNet.</li></ul>

Figura 4: Métodos de reconocimiento facial asistidos por computador[15]

Para este trabajo investigativo se utilizó el acercamiento de métodos de Deep Learning para el desarrollo de sistema propuesto, ya que es el más moderno.

## 2.2.2 Inteligencia artificial

Se describe como la capacidad de computadores para utilizar algoritmos, aprender datos, y utilizar el aprendizaje para tomar decisiones, intentando simular el comportamiento de la inteligencia humana[27]. Esta tecnología ha mostrado resultados notables en aplicaciones para el procesamiento de imágenes, entre otras disciplinas [28].

### 2.2.2.1 Conceptos básicos de inteligencia artificial

A continuación, se describen los componentes principales de una inteligencia artificial.

#### 2.2.2.1.1 Machine learning

También conocido como aprendizaje automático, es una disciplina dentro de la inteligencia artificial que se dedica al desarrollo de algoritmos que mejoren su rendimiento mediante el aprendizaje de datos. Los problemas principales que son resueltos mediante este son los de predicción, agrupación, y clasificación [28].

##### 2.2.2.1.1.1 Clasificación del Machine learning

Según cómo funcione el algoritmo de aprendizaje, se pueden distinguir cuatro clasificaciones: aprendizaje supervisado, aprendizaje no supervisado, aprendizaje semi - supervisado, y aprendizaje de refuerzo [28].

En el aprendizaje supervisado se utiliza datos etiquetados durante en el entrenamiento del algoritmo con el objetivo de predecir el tipo o valor de un nuevo dato. Este tipo se utiliza para resolver problemas de clasificación y regresión. En el caso de clasificación, un ejemplo es el de predecir si en una fotografía se encuentra un cierto específico de animal [28].

En el aprendizaje no supervisado se utiliza datos sin etiquetas durante el entrenamiento. Este tipo es utilizado cuando es necesario agrupar datos, estos se clasifican según diferentes características que no emplean etiquetas [28].

En el aprendizaje semi – supervisado se utiliza tantos datos con etiqueta como sin etiqueta durante el entrenamiento del algoritmo. Se considera como ideal, y no suele tener aplicaciones prácticas [28].

El funcionamiento del aprendizaje de refuerzo se basa en obtener recompensas mediante la interacción con el ambiente. Las acciones que se hayan tomado se juzgan con niveles de recompensa, y luego se entrena el modelo. Este utiliza un mecanismo de prueba y error para encontrar la mejor operación en el algoritmo [28].

#### **2.2.2.1.2 Deep Learning**

También llamado aprendizaje profundo, se considera como un subcampo del machine learning, y se utiliza para resolver problemas que implican una gran cantidad de datos [27].

Su funcionamiento se basa en aprender mediante capas de representación, compuestas de unidades neuronales, donde se alimenta al algoritmo con datos sin procesar y este descubre las representaciones necesarias para la detección o clasificación que se desee realizar. Mientras más capas existan, se pueden aprender funciones más complejas. Por ejemplo, en el caso del procesamiento de imágenes cada fotografía al estar compuesta de una matriz de valores de píxeles, estos se pueden alimentar en el algoritmo, y en la primera capa de representación se aprende sobre la presencia o ausencia de bordes en diversas orientaciones y lugares específicos en la imagen. En la segunda capa se detecta más arreglos de bordes, en la tercera capa se reconocen partes de objetos al combinar los bordes, y consecuentemente en las siguientes capas se detectan objetos al combinar las partes [29].

El objetivo principal en el Deep Learning es que estas capas no sean diseñadas por el ser humano, si no que sean el aprendidas por un procedimiento de artificial por el computador [29].

A esta composición de capas se conoce como redes neuronales [30].

##### **2.2.2.1.2.1 Redes Neuronales Artificiales**

Fueron concebidas en inspiración por el funcionamiento de las redes neuronales biológicas, específicamente las del cerebro humano, en ellas una neurona biológica recibe múltiples

señales a través de sus dendritas, que están conectadas a otras neuronas en un proceso llamado sinapsis, y envía solamente una corriente de acción a través de su axón [31], [32].

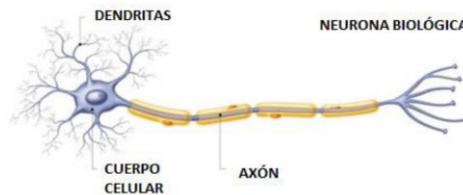


Figura 5: Partes de una neurona biológica[33]

Usando a las redes neuronales biológicas de base, las redes neuronales artificiales (ANN) se forman por unidades, también llamados nodos o neuronas artificiales, que combinan múltiples entradas para producir una sola salida. La ANN se puede ver como un aproximador de funciones matemáticas, donde su salida es la de una función. En cada nodo se realiza una suma de sus entradas, donde cada una tienen un parámetro de peso que las multiplica. El resultado de la suma, antes de pasarse al siguiente nodo, se pasa por una función de activación, cuyo objetivo es traer no linealidad a la predicción, ya que muchos problemas del mundo real no se pueden modelar sólo con funciones lineales, que serían el resultando de utilizar solamente sumas [31], [32], [34].

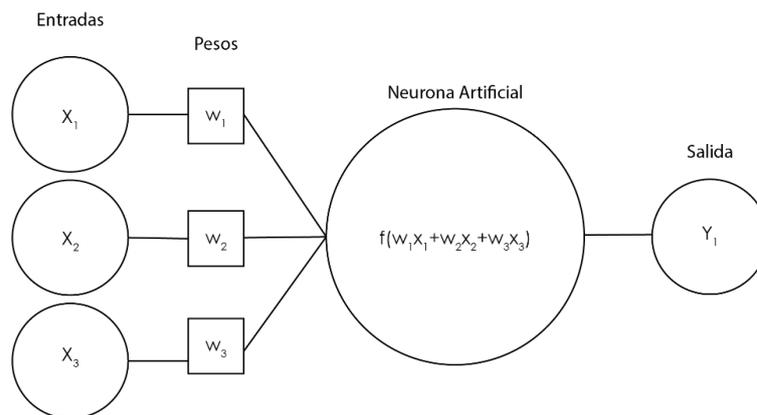


Figura 6: Estructura de una neurona artificial, que contiene la suma de sus entradas mientras se aplica una función de activación  $f(x)$ . Donde  $x_1, x_2, x_3$  son entradas.  $w_1, w_2, w_3$  son pesos.  $y_1$  es la salida[34].

En las ANN se utilizan una variedad de funciones de activación, según el uso que se le va a dar a la red. Entre las más utilizadas están las siguientes[34]:

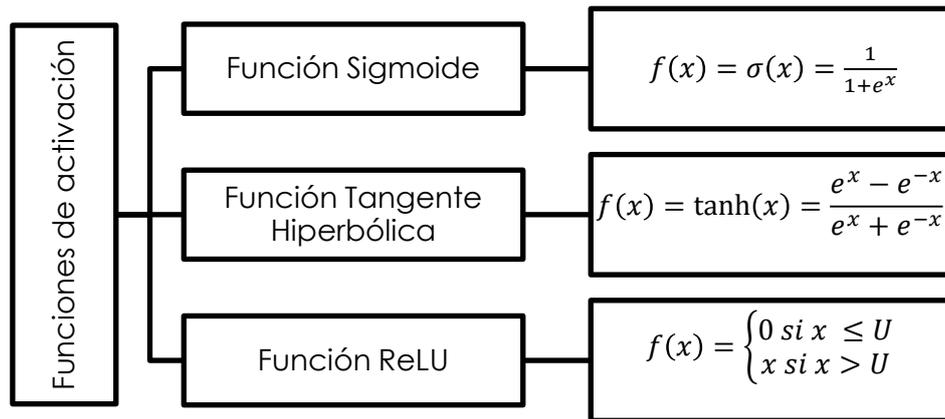


Figura 7: Funciones de activación utilizadas en las ANN. Donde x es una variable, y U es el umbral deseado[34].

Las neuronas artificiales en una ANN están distribuidas en capas, también llamados niveles. Se pueden diferenciar tres tipos de capas en una ANN, estas son[30]:

1. Capa de entrada: Las neuronas artificiales en este nivel se encargan de recibir los datos desde el exterior [33].
2. Capa oculta: Estas neuronas artificiales realizan el procesamiento de la información, mientras presentan interconexión entre ellas. Esta capa tiene la característica de tener conexión directa con la capa de entrada, y presentar un número elevado de neuronas [33].
3. Capa de salida: Las neuronas artificiales en este nivel se encargan de recibir la información procesada y enviarla al exterior [33].

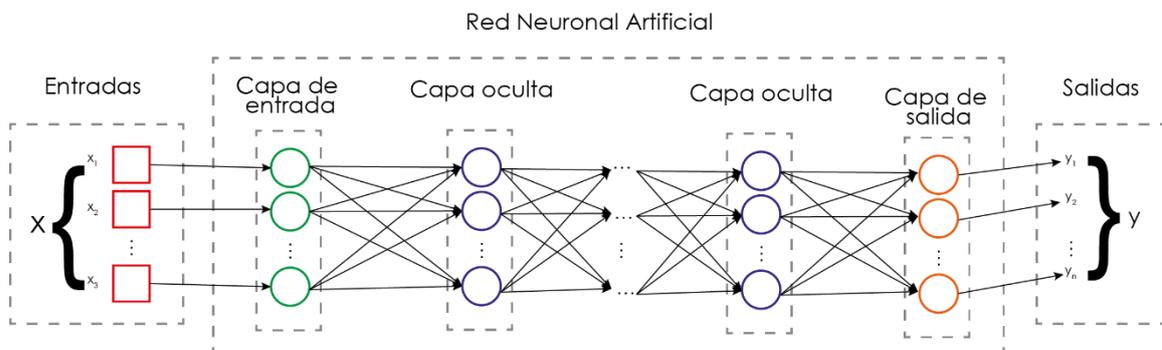


Figura 8: Capas de entrada, oculta y de salida en una Red Neuronal Artificial. Cada círculo representa un neurona artificial[30], [33].

Este modelo que presenta varias capas ocultas se conoce también como modelo de Perceptrón Multicapa (MLP)[33].

Las ANN dependen de tener una gran cantidad de datos durante su entrenamiento, ya que al no tenerla se han mostrado predicciones imprecisas como resultado, por lo que se recomienda utilizarse ANN en problemas donde se tenga una gran cantidad de información de entrada [32].

### 2.2.2.2 Visión Artificial

Son técnicas utilizadas para habilitar a un máquina con la capacidad de analizar imágenes en su entorno, y obtener información específica de ellas [9], [35].

Los sistema de visión artificial se aplican en tareas donde existan repetición en aplicaciones de inspección visual, y suelen realizar tres procesos principales[9]:

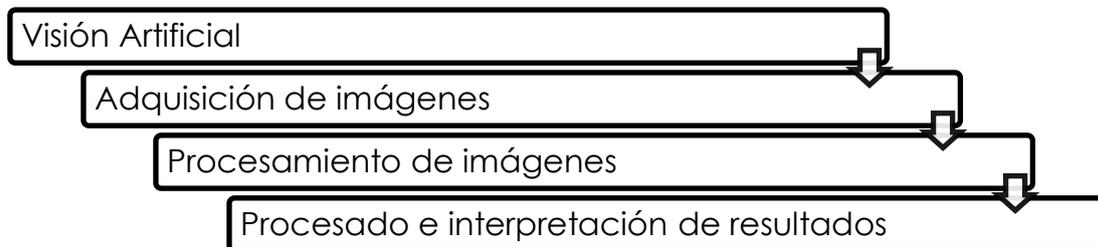


Figura 9 Procesos en un sistema de visión artificial[3].

El proceso de adquisición de imágenes involucra la utilización de un dispositivo de captación, como una cámara, para obtener imágenes digitales. Después estas imágenes son procesadas, separando de la imagen sólo las regiones de interés con respecto a un criterio de forma, color, tamaño o longitud. Finalmente en la interpretación de resultados se clasifica los objetos reconocidos en la imagen [35].

### 2.2.2.3 Redes Neuronales Convolucionales (CNN) y reconocimiento facial

Las redes Neuronales Convolucionales (CNN) son un tipo de ANN y se consideran la técnica de Deep Learning más exitosa en el análisis de imágenes. Con ellas se pueden descubrir características simples en una imagen de entrada, como dónde existen bordes y esquinas, hasta estructuras más complejas como identificar y clasificar objetos [36], [37].

#### 2.2.2.3.1 Arquitectura y funcionamiento de una CNN

##### 2.2.2.3.1.1 Convolución

La operación principal en una CNN es la convolución cuyo objetivo es el almacenamiento de información encontrada en la imagen, en vectores de características. Estos sirven como representaciones del objeto a identificar o de sus partes. Para ello en esta operación se ve a una imagen como una matriz compuesta por el valor de sus píxeles, de un ancho “w” y un alto “h”. En el caso de utilizar una imagen a color, también se considera una profundidad “d”, que toma el valor de 3 si se usa RGB como modelo de color, 4 si es CMY, y 1 si la imagen es en blanco y negro [15].

En el proceso de convolución, se utiliza junto a la matriz que representa la imagen, otra matriz más pequeña, llamada filtro, kernel o también núcleo. Estas dos matrices se combinan mediante operaciones matemáticas, recorriendo el filtro sobre la matriz de la imagen hasta cubrir toda su extensión. Como resultado se obtiene otra matriz llamada mapa de características, con la información relevante de la imagen [15].

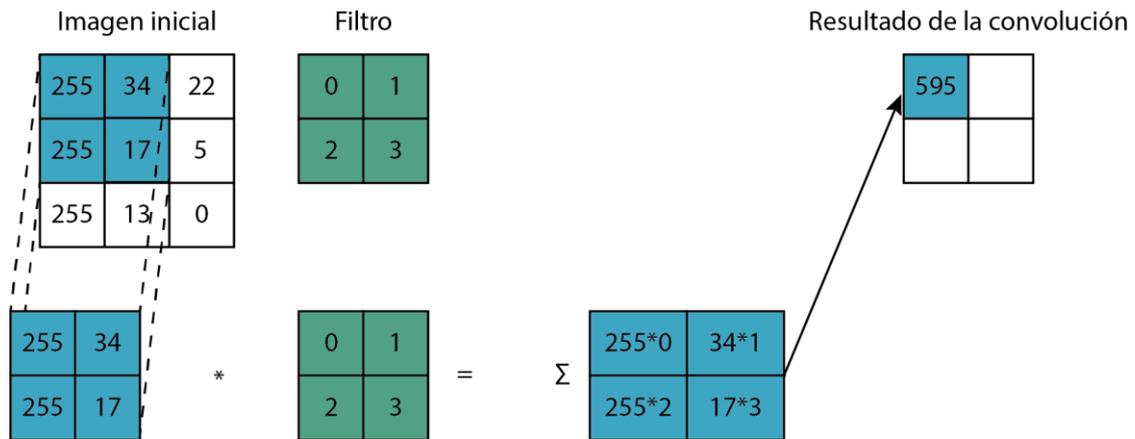


Figura 10: Proceso de Convolución [15].

Un filtro sólo puede detectar una característica de la imagen, como por ejemplo dónde están líneas verticales, por lo que se utiliza varios filtros para recoger toda la información de la imagen[30].

### 2.2.2.3.1.2 Arquitectura de una Red Neuronal Convolutiva

Las funciones en una CNN están distribuidas en capas, donde se encuentran las siguientes[15], [30], [38]:

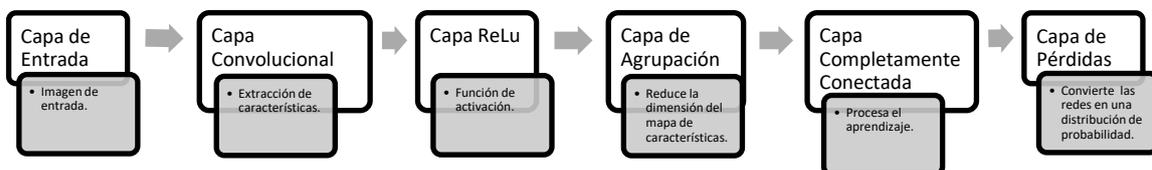


Figura 11: Capas en una Red Neuronal Convolutiva[15]

El trabajo en la capa de entrada es el de contener los valores en píxeles de las imágenes de entrada[15].

En la Capa convolutiva se busca extraer características de los datos proporcionados por la capa de entrada, mediante la convolución. Para hacerlo se ocupan filtros, pueden existir varios en una capa trabajando paralelamente, y generalmente existen varias de estas capas a lo largo de la CNN, con el objetivo de aprender patrones cada vez más compuestos. La elección de qué filtros deben usarse para detectar un tipo específico de característica es parte del proceso de Machine Learning, ya que en estas redes neuronales los pesos aprendidos por la red son justamente los valores que componen la matriz de filtro. La salida de esta capa es el mapa de características, resultado de la convolución, luego de pasarse los filtros por toda la imagen [15].

La Capa ReLU toma el mapa de características de la capa anterior, y le aplica la función de activación ReLU a cada valor de la matriz. El objetivo de esta operación es traer no linealidad al mapa de características [15].

Después, la Capa de Agrupación toma el mapa de características, modificado por la capa ReLU, y realiza una reducción en la dimensionalidad del mapa. Es decir, reduce el tamaño de la matriz. Esta operación tiene varios objetivos, el principal siendo evitar el sobre ajuste del modelo, ya que se buscan destacar los rasgos más invariantes con respecto a su posición en la imagen. Esto se refiere que cuando una característica ya ha sido detectada, su posición en la imagen no es tan importante, como lo es su posición relativa con respecto a otras características, y se busca preservar esta información. Por otra parte, al tener matrices de menor tamaño, se reduce la computación necesaria para el procesamiento en las capas siguientes de la red. A esta capa también se llama de pooling, o subsampling [15], [30], [38].

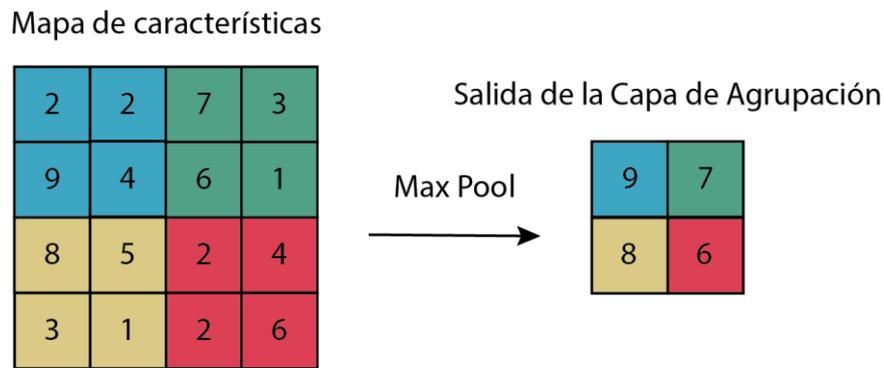


Figura 12: Función Max Pooling en la Capa de Agrupación en una CNN[30].

La función Max Pooling es la más utilizada para cumplir con el objetivo de la agrupación. En ella se toma los valores más altos de una región  $2k \times 2k$  del mapa de características, para así sólo tener la característica más relevante. Este proceso se aplica a cada mapa de características por separado [15], [38].

En una CNN, las capas de Convolución, ReLU, y de agrupación se suelen usar sucesivamente y se utilizan varios bloques de ellas. Después de estos bloques se encuentra la Capa Completamente Conectada, que agrupa la información obtenida por las capas anteriores para realizar una posterior clasificación final. Esta capa tiene la estructura de una ANN tradicional, donde cada neurona que la compone tiene como entrada cada valor de todos los mapas de características entregados por la capa de agrupación que le precede. Los pesos usados en estas neuronas representan la importancia que tiene esa conexión en la clasificación de la imagen dentro del conjunto de categorías para la predicción. La salida de esta capa es un conjunto de valores conocidos como puntaje de clase, que representan la probabilidad que la imagen de entrada pertenezca a cada una de las clases posibles para la predicción [15], [30].

Finalmente, el conjunto de puntajes de clase pasa por una Capa de Pérdidas, que compara estas predicciones con los valores reales de las imágenes, y se elige una de ellas[30]. Usualmente en esta capa se utiliza un clasificador de pérdidas Softmax, que toma los valores

de entrada y los transforma en una distribución de probabilidad con la particularidad que la suma total de todas las probabilidades da uno.

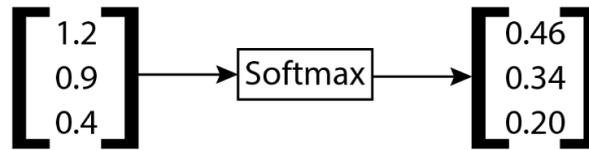


Figura 13: Ejemplo de la función Softmax[15].

La clase con mayor probabilidad se elige como salida de la CNN, y es la predicción final de la red [15], [30].S

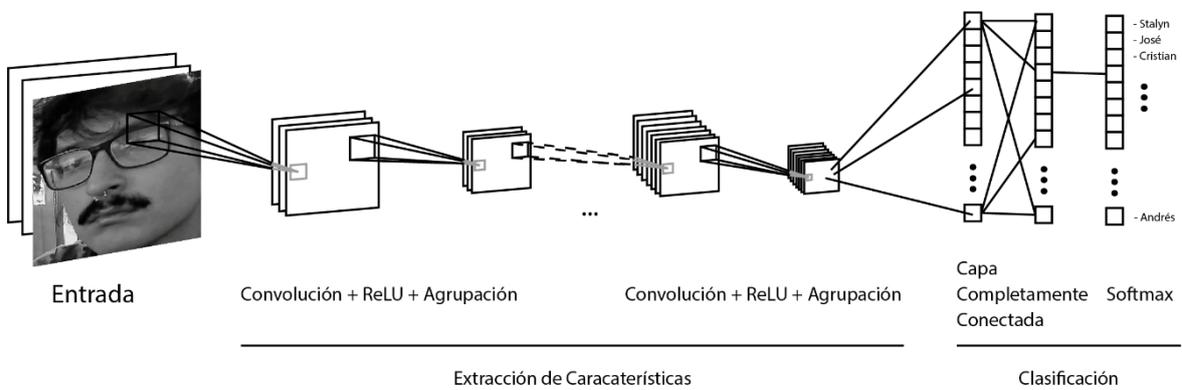


Figura 14: Estructura de una Red Neuronal Convolutacional[15].

El finalizar la predicción se aplica un algoritmo llamado Propagación hacia atrás (Backpropagation), que consiste en calcular el error de la predicción final con respecto a la salida esperada, con el objetivo de corregir los pesos en las neuronas partiendo desde la última capa hasta la primera, para obtener una mejor predicción [39].

### 2.2.2.3.1.3 Modelos de Arquitecturas de Red Neuronal Convolutacional

Dependiendo de la combinación de las capas anteriormente explicadas, se pueden tener diferentes modelos de CNN. Algunas de las más conocidas son:

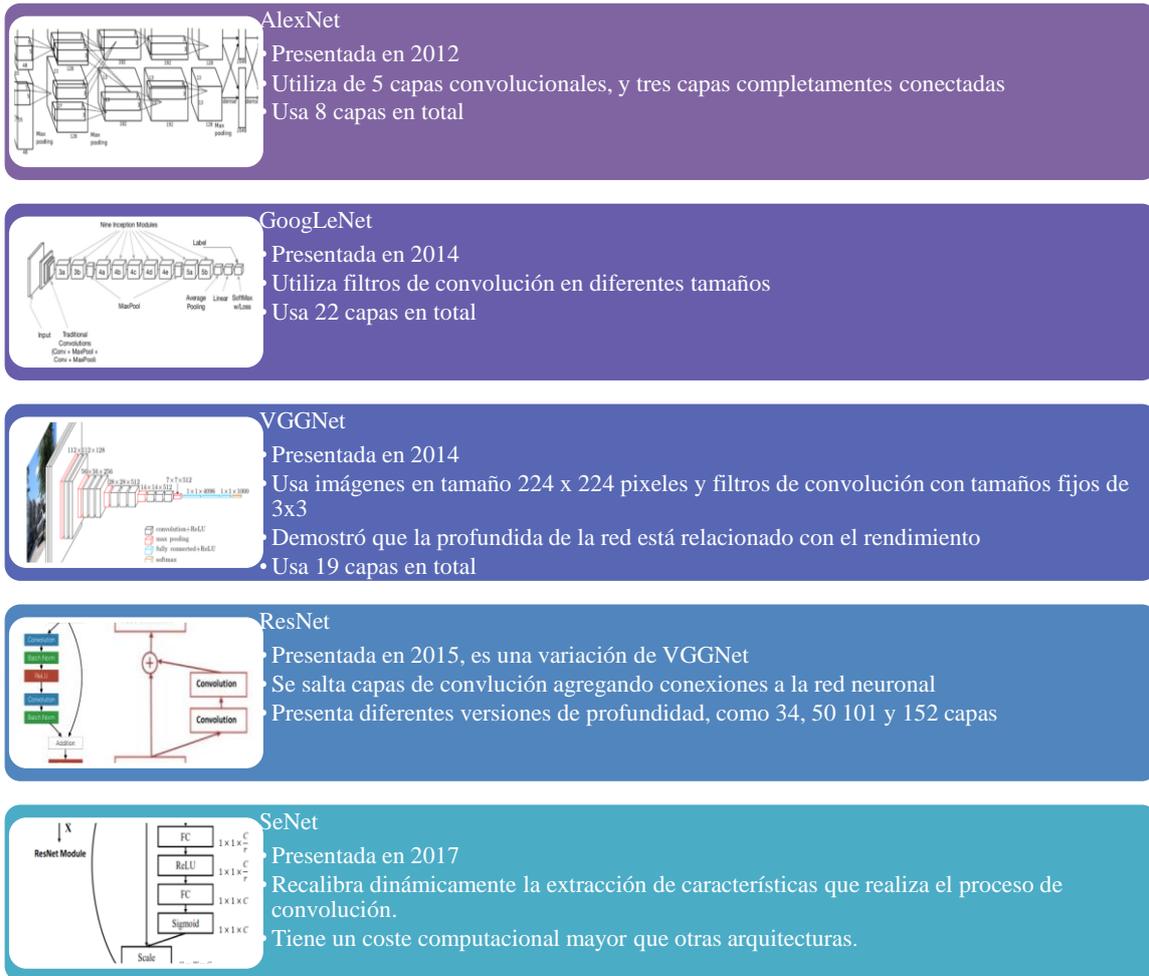


Figura 15: Arquitecturas de CNN actuales

En cuanto a las arquitecturas de CNN, la última fue presentada en el año 2017. Desde entonces las investigaciones publicadas sobre este tema se han centrado en métodos para mejorar el sobreajuste y nuevos algoritmos de pérdida[15].

### 2.2.2.3.2 Mejoras y optimizaciones en el funcionamiento de una CNN

Cuando no hay muchos datos para el entrenamiento de una CNN, puede que el modelo esté muy ajustado a la información aprendida, por lo que no va a poder generalizarlos. A esto se conoce como sobreajuste, para evitarlo existen algunos métodos que se pueden aplicar al aprendizaje en un proceso llamado regularización[15], [30], [36].

El método más utilizado en las CNN para evitar el sobreajuste es el Dropout, que consiste en desconectar neuronas seleccionadas al azar con el objetivo que la red sea menos sensible a los pesos de esas neuronas, ya que ha demostrado que se resultará en una mejor generalización de la red[15], [30].

Otro método que se utiliza cuando los datos de entrenamiento son imágenes es el de Data Augmentation, que consiste en realizar transformaciones en las imágenes, como rotaciones u otras deformaciones, para generar nueva información sin tener que introducir más imágenes al entrenamiento[36].

### 2.2.3 Procesamiento de imágenes utilizando Python

El lenguaje de programación más utilizado en investigaciones y aplicaciones sobre el reconocimiento facial mediante modelos de aprendizaje es Python [40].

#### 2.2.3.1 Librerías y herramientas para el procesamiento de imágenes en Python

Entre librerías disponibles en Python para el procesamiento de imágenes e inteligencia artificial se tienen:

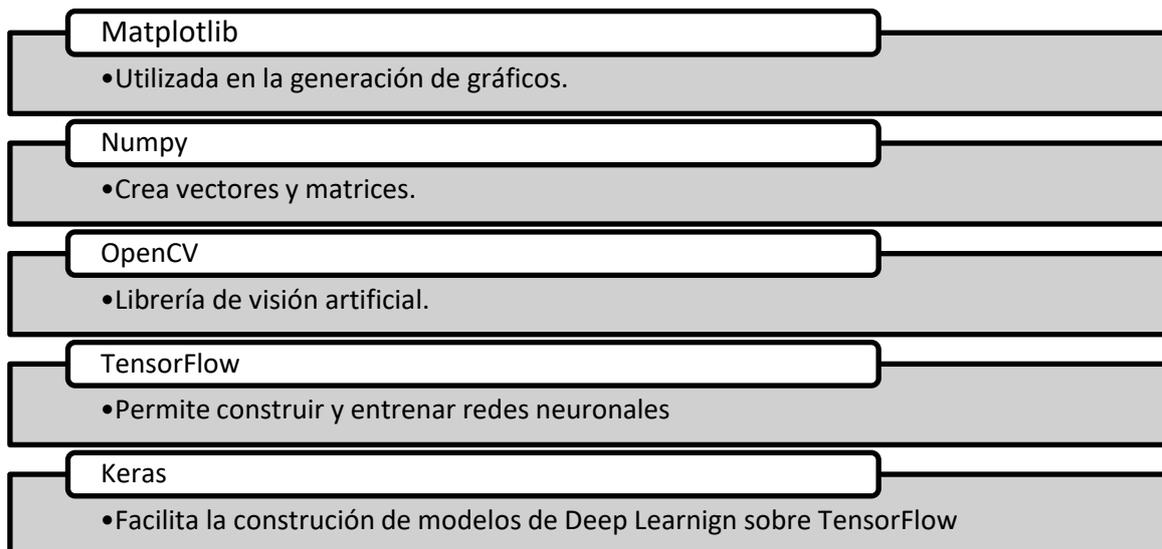


Figura 16: Librerías en Python para visión artificial[15].

## CAPÍTULO III

### 3. METODOLOGIA.

#### 3.1 Tipo de Investigación.

El principal objetivo de este trabajo investigativo fue la implementación de un sistema de control de acceso utilizando inteligencia artificial en la Dirección de Tecnologías de Información y Comunicación de la UNACH, para ello se empleó en un primer acercamiento una investigación descriptiva donde se detalló el funcionamiento de un modelo de Red Neuronal Convolucional, recopilando información sobre las etapas y capas involucradas en su entrenamiento, así como entender el funcionamiento de los métodos de detección facial más modernos. Esta se utilizó para tener información suficiente para crear un diseño propio de inteligencia artificial que sea capaz de clasificar los rostros de los trabajadores del departamento de redes de la DTIC.

En otro enfoque se empleó una investigación aplicada, pues se utilizó el conocimiento adquirido sobre el diseño de las CNN para el desarrollo de un prototipo para el sistema de control de acceso, utilizando el lenguaje de programación Python en conjunto de la librería customtkinter para la creación de la interfaz gráfica de usuario, la librería de Tensorflow para el diseño y entrenamiento de la red neuronal, y la librería de OpenCV para captura de video en tiempo real. En esta parte de la investigación se buscó aplicar todos los conocimientos tecnológicos adquiridos para el desarrollo del proyecto.

Después del desarrollo del proyecto, se aplicó una investigación experimental para conocer estadísticamente el funcionamiento del sistema creado. En esta se manipularon variables como la distancia que existe entre la persona a reconocerse y la cámara, el ángulo que existe entre el rostro de la persona, y la iluminación en el lugar de la implementación. La finalidad de esta parte de la investigación fue validar el sistema desarrollado.

#### 3.2 Diseño de Investigación

Para el desarrollo del presente trabajo se siguió el siguiente esquema:

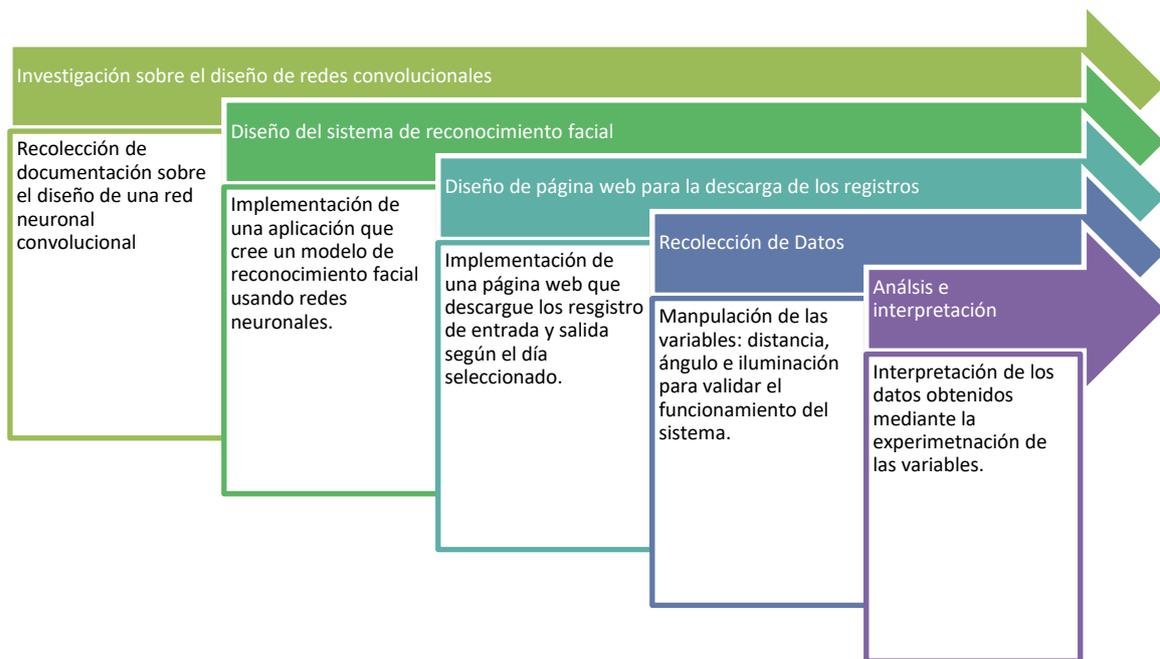


Figura 17: Etapas en el desarrollo de la investigación

En la primera etapa se recolectó información de artículos científicos y trabajos de grado sobre el funcionamiento de las redes neuronales convolucionales y las capas que se involucran en ellas: capas de convolución, capas de agrupación, capas ReLu y capas completamente conectadas, además de métodos para evitar el sobre ajuste como el aumento de datos y las capas de dropout. Esto con el objetivo de desarrollar un diseño propio de red neuronal convolucional para el proyecto, que pueda clasificar correctamente a cada usuario registrado.

En la etapa siguiente se desarrolló una aplicación en el lenguaje de programación Python, utilizando la librería customtkinter para generar una interfaz gráfica de usuario, que satisfaga las necesidades del proyecto:

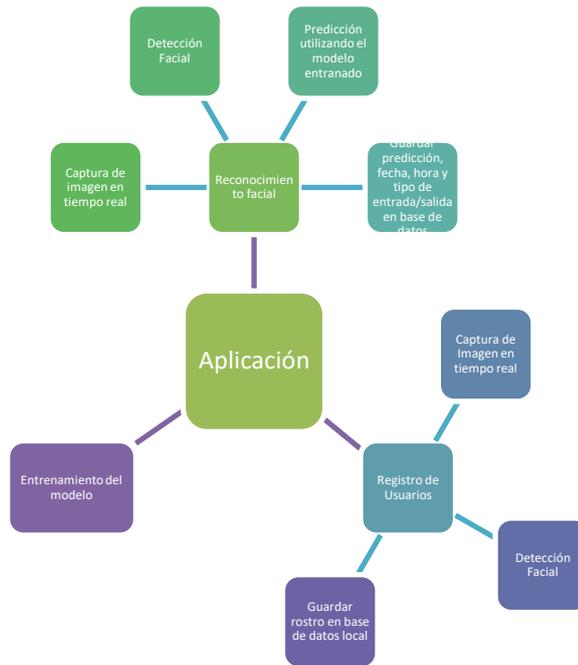


Figura 18: Funciones de la aplicación creada

Se fijaron tres funciones principales que debe cumplir la aplicación: el reconocimiento facial, el registro de usuarios, y el entrenamiento del modelo. Para entrenar el modelo CNN, se necesita una extensa cantidad de imágenes del rostro de cada persona, o clase, que se quiera distinguir en el video a tiempo real. Para el prototipo propuesto se utilizó el siguiente hardware en su diseño:

- Laptop Dell G7 con procesador Intel i7 de 8va generación
- Tarjeta de video Nvidia GTX 1080ti
- Cámara inalámbrica EZVIZ TY1

Se procedió primero a desarrollar la función de Registro de Usuarios. Para su diseño se procedió de la siguiente manera:

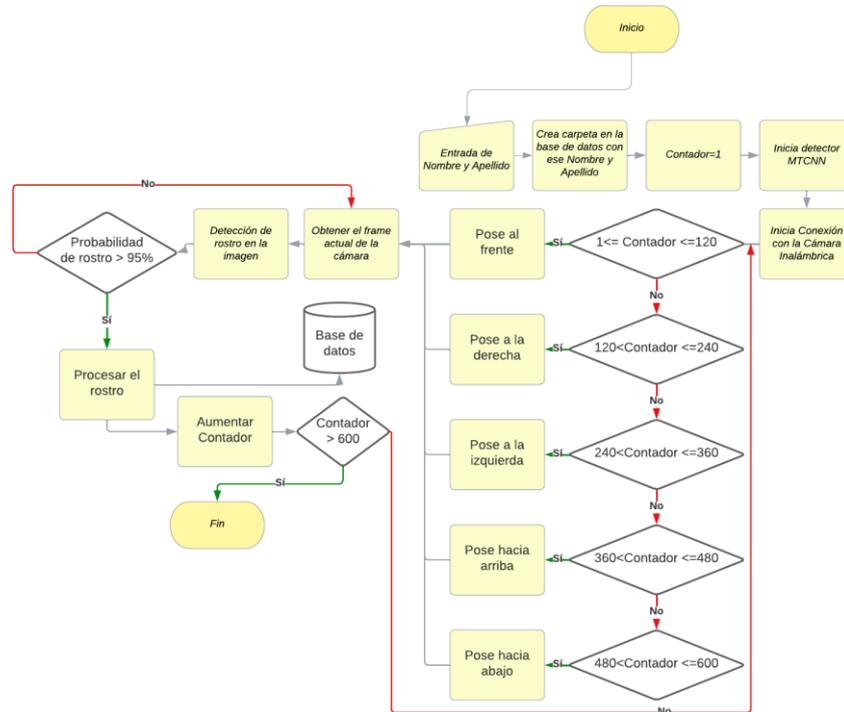


Figura 19: Diagrama de flujo para el registro de Usuarios

Se eligió guardar un total de 600 imágenes por persona, siguiendo las investigaciones revisadas [10], [15], [41], esto debido a que las redes neuronales aprenden mejor a diferenciar clases cuando se tiene una gran cantidad de información.

Para obtener el video a tiempo real, se realizaron funciones para la conexión a una cámara EZVIZ TY1 mediante el protocolo Real Time Streaming Protocol (RTSP)[42]. Debido a las limitaciones de hardware, se realiza la captura de frames mediante un hilo dedicado a obtener constantemente la última imagen disponible. De esta manera, el programa principal puede ocupar sus recursos en procesos más demandantes, como las predicciones con el detector facial, y las predicciones con el modelo propio de reconocimiento facial.

```

def read_camera_frames():
    global latest_frame
    global cap
    global DetenerHilo
    url = "rtsp://admin:FMLAGI@192.168.100.77:554/" # Dirección del stream
    cap = cv2.VideoCapture(url) # Reemplaza la URL con la de tu cámara

    while not DetenerHilo:
        ret, frame = cap.read()
        if ret:
            latest_frame = frame # Actualiza el último fotograma
        cap.release()

def ObtenerFrameActualDeLaCámara():
    global latest_frame
    if latest_frame is not None:
        return(latest_frame)

def ConfigurarCámara():
    global latest_frame
    global cap
    global DetenerHilo
    global frame_thread

    latest_frame = None # Variable para almacenar el último fotograma
    cap = None
    DetenerHilo=False

    # Inicia el hilo para la lectura de fotogramas
    frame_thread = threading.Thread(target=read_camera_frames)
    frame_thread.daemon = True # El hilo se ejecutará como demonio(se
    frame_thread.start()

```

Figura 20: Funciones para la obtención de video a tiempo real

Después, se eligió utilizar el detector facial MTCNN para localizar el rostro de la persona a la que se quiere registrar, ya que según una investigación realizada es el método más preciso entre otros algoritmos populares revisados [12]. Este brinda un dato de confianza que indica el porcentaje de precisión de la predicción sobre que en el área indicada existe un rostro. Se utiliza este dato para procesar y posteriormente guardar la imagen en la carpeta del nuevo

usuario creada, sólo si existe una confianza mayor al 95%. Este valor se eligió para filtrar las predicciones erróneas causadas por patrones complejos encontrados en la ropa, o al fondo del video.

La imagen se procesa recortando la región donde se encuentra el rostro, cambiándola a un tamaño de 224x224 pixeles, siguiendo la arquitectura de las VGGNet[15], y convirtiéndola a escala de grises para aliviar el coste computacional que tendrá el posterior entrenamiento de la red. En la aplicación se implementó una imagen de referencia para variar la pose del rostro cada 120 imágenes registradas.



Figura 21: Interfaz del Registro de Usuarios

Para filtrar el reconocimiento de usuarios no registrados, de manera manual se creó una clase llamada “Desconocidos” que consta de 600 imágenes de personas aleatorias descargadas del conjunto de datos gratuito “Labeled faces in the wild”. A este conjunto de fotos se les aplicó el mismo procesamiento que a las fotos registradas con la función implementada en la aplicación.



Figura 22: Conjunto de datos para la clase "Desconocidos"

En otra pestaña de la aplicación se implementó la función de entrenamiento de la red neuronal. Se siguió su diseño de la siguiente manera:

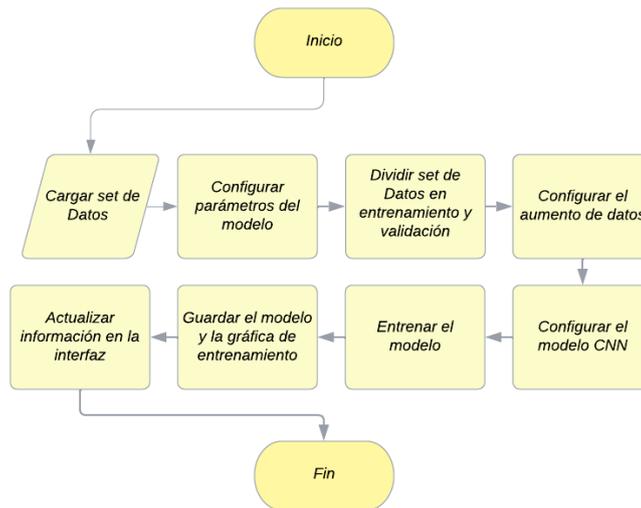


Figura 23: Diagrama de flujo para el entrenamiento del modelo

Para entrenar el modelo, primero se toma la base de datos local donde están localizados las carpetas asignadas a cada usuario registrado, con sus 600 fotografías de rostro respectivas, y se dividen estos datos para utilizarse un 80% en el proceso de entrenamiento y un 20% en el proceso de validación. Esta división se realiza en cada clase, y de manera aleatoria al distribuir las imágenes a los dos tipos. Durante el entrenamiento solo se utiliza esa porción de datos correspondiente, y el resto llamado “validación” se usa para hacer predicciones y ajustar los valores de las neuronas en algoritmo de Propagación hacia atrás.

Para evitar el sobreajuste se implementó técnicas de aumento de datos. Se utilizan cuando el conjunto de datos es pequeño, o sin muchas variaciones. En este proceso se genera nuevas imágenes a partir del conjunto de datos disponible. Se implementó siguiendo las investigaciones revisadas[15], donde se indicaba que tener partes del rostro en las mismas regiones en todas las fotos era mejor para el entrenamiento, por lo que se modificó solamente el brillo y contraste de las imágenes.

```

Data_Augmentation = tf.keras.Sequential(
    [
        tf.keras.layers.RandomBrightness(factor=0.2, input_shape=(Imagen_ancho, Imagen_alto, 1)),
        tf.keras.layers.RandomContrast(factor=0.2, input_shape=(Imagen_ancho, Imagen_alto, 1))
    ]
)
  
```

Figura 24: Aumento de datos aplicado al modelo

Para el diseño de la CNN se varió la arquitectura de una VGGNet para adaptarla al hardware disponible para el proyecto. Se implementó una red neuronal de 15 capas, configurada de la siguiente manera:

```

Modelo = tf.keras.Sequential([
    Data_Augmentation,
    tf.keras.layers.Rescaling(1./255),
    Conv2D(16,11,padding='same', activation='relu',input_shape=(224,224,1)),
    MaxPooling2D(2),
    Conv2D(128,9,padding='same',activation='relu'),
    Conv2D(128,9,padding='same',activation='relu'),
    tf.keras.layers.Dropout(0.2),
    MaxPooling2D(2),
    Conv2D(256,5,padding='same',activation='relu'),
    Conv2D(256,5,padding='same',activation='relu'),
    Conv2D(256,5,padding='same',activation='relu'),
    Flatten(),
    Dense(128, activation='relu'),
    tf.keras.layers.Dropout(0.5),
    Dense(NúmeroDeClases, activation='softmax')
])

```

Figura 25: Arquitectura para el modelo de la red neuronal convolucional

La primera capa corresponde al aumento de datos configurado con anterioridad. Se procede entonces a una capa Rescaling, que divide los píxeles de la imagen en un valor de 255. Esto se realiza para que valores de la imagen estén en un rango entre 0 y 255, ya que esta es una escala manejable para la red neuronal.

Con las imágenes preparadas, se procede la primera capa de convolución. Aquí se tiene 16 filtros de tamaño 11x11 que recorrerán la imagen con el objetivo de descubrir estructuras simples en la imagen. En esta, y todas las capas de convolución, se utiliza la opción de padding “same” para mantener el mismo tamaño de imagen al finalizar la convolución. También en todas las capas de convolución se utiliza la función de activación “ReLU”, que se aplica a todos los elementos de la convolución, para traer no linealidad al modelo. Después de esta, se encuentra una capa MaxPooling de tamaño 2x2, que reduce el tamaño de la imagen a la mitad, con el objetivo de poder aumentar el tamaño de filtro en capas superiores.

Después se tienen dos capas seguidas de convolución, ambas compuestas de 128 filtros y con un tamaño de 9x9. El objetivo de estas capas es descubrir estructuras más complejas en el rostro, como ojos, o boca. Aquí el tamaño de los filtros es menor, y seguirá disminuyendo en capas superiores, para poder extraer detalles más finos en la imagen. A continuación de estas capas se encuentra la primera capa de Dropout en la red, que desconecta aleatoriamente el 20% de las neuronas para evitar el sobreajuste, y la poder realizar mejores predicciones sobre datos no vistos. A continuación, se tiene otra capa de Maxpooling para aliviar el coste computacional de las siguientes capas de convolución.

El último bloque de capas de convolución está compuesto de tres capas de convolución con 256 filtros cada una, y un tamaño de 5x5. El objetivo de estas es descubrir características específicas del rostro de cada usuario. Después del proceso de convolución se utiliza una capa flatten para convertir las matrices aprendidas en los procesos de convolución, en un solo vector unidimensional, esto se realiza para poder conectar las neuronas aprendidas en la convolución en una capa de neuronas densa.

Se utiliza después una capa de neuronas densa de 128 neuronas, con una activación ReLU. Cada una de estas neuronas se conecta a todas las salidas de la capa anterior. Esta capa densa

tiene varios objetivos, como el de integrar la información obtenida por los diferentes filtros, y poder generar representaciones más abstractas de las características aprendidas. A esta capa le sigue la última capa de DropOut, que desconecta esta vez un 50% de las neuronas aprendidas.

La última capa es una capa densa con un número de neuronas igual al de usuarios registrados, y una función de activación softmax. Su función es tomar el vector de estas últimas neuronas y normalizarlo en un vector de probabilidades donde la suma de todas es igual a 1.

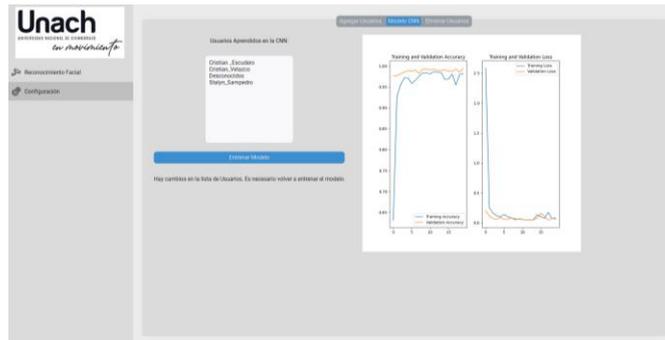


Figura 26: Interfaz de Entrenamiento del modelo

Como último paso se compila el modelo utilizando el algoritmo optimizador Adam, para ajustar los pesos de la red neuronal de manera individual para cada parámetro del modelo, y la función de pérdida Sparse Categorical Crossentropy, que calcula la diferencia entre la distribución de probabilidad predicha por el modelo y la distribución real de las clases. Un vez terminado se guarda tanto el modelo como la curva de aprendizaje generada.

```
#Compilar Modelo
Modelo.compile(optimizer='adam', loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=False), metrics=['accuracy'])

History= Modelo.fit(
    Entrenamiento_ds,
    validation_data=Validacion_ds,
    epochs=epochs
)
```

Figura 27: Compilación y entrenamiento del modelo

La función de reconocimiento se implementó en la página principal de la aplicación, y para ello se siguió el siguiente diseño:

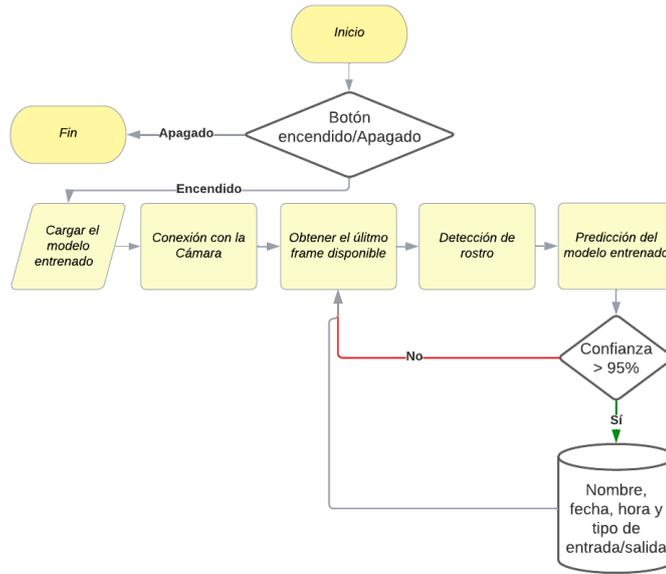


Figura 28: Diseño de la función reconocimiento facial

En esta función de la aplicación se carga el modelo entrenado, y se realiza la conexión con la cámara inalámbrica de la misma manera que en el registro de usuarios. Luego de detectarse el rostro con el detector MTCNN, este clasifica en alguna de las clases registradas en el modelo de CNN.

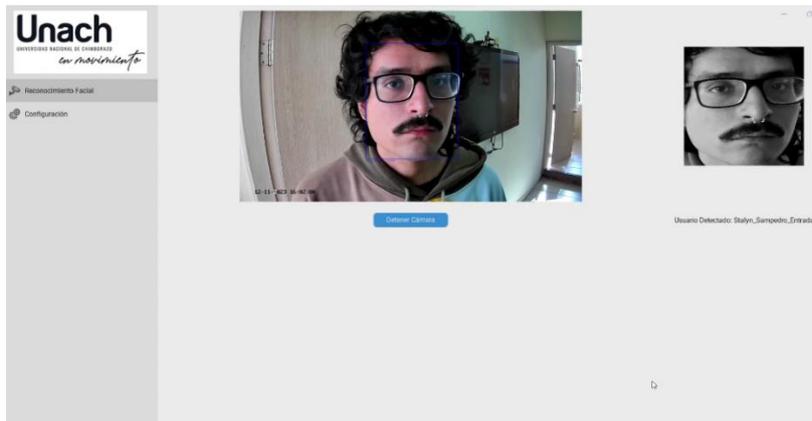


Figura 29: Interfaz de reconocimiento de usuarios

Según la confianza que devuelva el modelo, se guardan los datos de identidad, confianza, fecha, hora y tipo/entrada y salida. Estos datos se guardan en una base de datos MySQL, siguiendo el siguiente algoritmo:

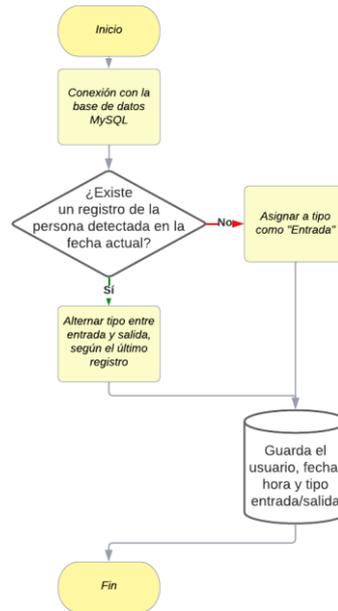


Figura 30: Diagrama de flujo para las funciones MySQL

Dependiendo del último tipo de registro del usuario, es decir si fue una entrada o salida, se cambia este valor y se guardan los demás en el registro control de acceso de los usuarios. Adicionalmente, debido a que es un sistema automático, el algoritmo pregunta si existe un registro en los últimos 10 segundos de la persona detectada, y sólo pasado ese tiempo se registra la nueva entrada o salida.

Para la fácil descarga de este registro se implementó una página web mediante el lenguaje de programación php para el manejo de los datos y html para el diseño de la interfaz.



Figura 31: Interfaz de descarga del registro de control de acceso

En la página se tiene un calendario para elegir el registro del día deseado, y se descarga todos los registros del día en un formato de CSV. La página se alojó en un servidor local mediante el uso de XAMPP.

### 3.3 Técnicas de recolección de Datos

Una vez implementada la aplicación, se utilizó la técnica de observación experimental para obtener datos sobre el funcionamiento del sistema correspondiente a los aciertos que realiza

el modelo en las predicciones, según se modifica las variables distancia entre la persona y la cámara, el ángulo que existe entre el rostro y la cámara, y la iluminación que existe en el cuarto. Estos datos se analizaron para describir estadísticamente los aciertos de la aplicación.

### 3.4 Población de estudio y de muestra

#### 3.4.1 Población

La población de estudio corresponde a los datos comprendidos de aciertos obtenidos en la experimentación de las variables: distancia, ángulo e iluminación. La variable de distancia se midió en tres puntos a 0.5m, 0.75m y 1m, estos fueron elegidos considerando las condiciones reales del área disponible en la bodega del DTIC. El ángulo se midió a 0 grados y 45 grados. Para la iluminación se consideró la iluminación recomendada para oficinas, entre 500 y 1000 lux [43], y una iluminación por debajo de estos valores.

La experimentación se realizó en la bodega del departamento de redes de la DTIC de la Universidad Nacional de Chimborazo, utilizando a dos trabajadores de esta misma y realizando 100 predicciones por cada variable mencionada, teniendo un total de 1200 datos como población.

#### 3.4.2 Muestra

La muestra considerada para el análisis en esta investigación corresponde a una selección aleatoria de los datos obtenidos en la experimentación de las variables.

#### 3.4.3 Variables

##### 3.4.3.1 Variables Independientes

Variable	Descripción	Indicador
Distancia	Espacio medido entre la persona y la cámara del sistema de registros.	• Metros
Ángulo del rostro	Ángulo del rostro con respecto a la posición de la cámara.	• Grados
Iluminación de la habitación	Iluminancia medida en la habitación.	• Lux

Tabla 1: Variables independientes

##### 3.4.3.2 Variables Dependientes

Variable	Descripción	Indicador
Aciertos del modelo	Aciertos en la predicción de la persona reconocida por el sistema.	• Acierto de la predicción • Desacierto de la predicción

Tabla 2: Variable dependiente

### 3.5 Hipótesis

Como hipótesis en esta investigación se realizó un estudio sobre si existe una diferencia significativa entre las distancias a interés de la implementación del sistema y el porcentaje de aciertos obtenido.

#### 3.5.1 Hipótesis Nula ( $H_0$ )

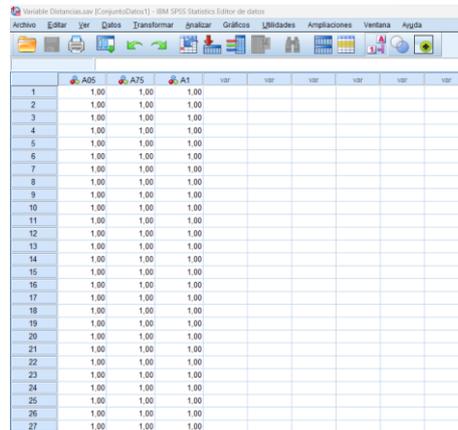
No existe diferencias significativas en las medianas del porcentaje de acierto entre las distancias 0.5m, 0.75m y 1m.

### 3.5.2 Hipótesis Alternativa ( $H_1$ )

Existe al menos una diferencia significativa en las medianas del porcentaje de acierto entre las distancias de 0.5m, 0.75m y 1m.

### 3.6 Métodos de análisis, y procesamiento de datos.

Para analizar los datos recolectados en la experimentación se utilizó el software IBM SPSS Statistics. Estos se codificaron de tal manera que una predicción correcta se representó con el número 1, y una incorrecta con el 0.



	ADS	A75	A1						
1	1.00	1.00	1.00						
2	1.00	1.00	1.00						
3	1.00	1.00	1.00						
4	1.00	1.00	1.00						
5	1.00	1.00	1.00						
6	1.00	1.00	1.00						
7	1.00	1.00	1.00						
8	1.00	1.00	1.00						
9	1.00	1.00	1.00						
10	1.00	1.00	1.00						
11	1.00	1.00	1.00						
12	1.00	1.00	1.00						
13	1.00	1.00	1.00						
14	1.00	1.00	1.00						
15	1.00	1.00	1.00						
16	1.00	1.00	1.00						
17	1.00	1.00	1.00						
18	1.00	1.00	1.00						
19	1.00	1.00	1.00						
20	1.00	1.00	1.00						
21	1.00	1.00	1.00						
22	1.00	1.00	1.00						
23	1.00	1.00	1.00						
24	1.00	1.00	1.00						
25	1.00	1.00	1.00						
26	1.00	1.00	1.00						
27	1.00	1.00	1.00						

Figura 32: Software IBM SPSS statistics

Los métodos aplicados a estos datos fueron primero, en una investigación descriptiva, un estudio de porcentajes de acierto, donde mediante SPSS se encontraron las frecuencias y los porcentajes de acierto según cada variación de las variables independientes explicadas anteriormente.

Posteriormente para comprobar la hipótesis planteada, se procesó los datos para tener una muestra de porcentajes de acierto. A estos se realizó un estudio de normalidad Kolmogorov-Smirnov para conocer si la distribución de los datos es normalizada. Después de obtenido tal resultado, se optó por la prueba Kruskal-Wallis para comprobar la hipótesis. Estos resultados se detallan en el siguiente capítulo.

# CAPÍTULO IV

## 4. RESULTADOS Y DISCUSIÓN

### 4.1 Resultados

#### 4.1.1 Arquitectura del sistema de control de acceso

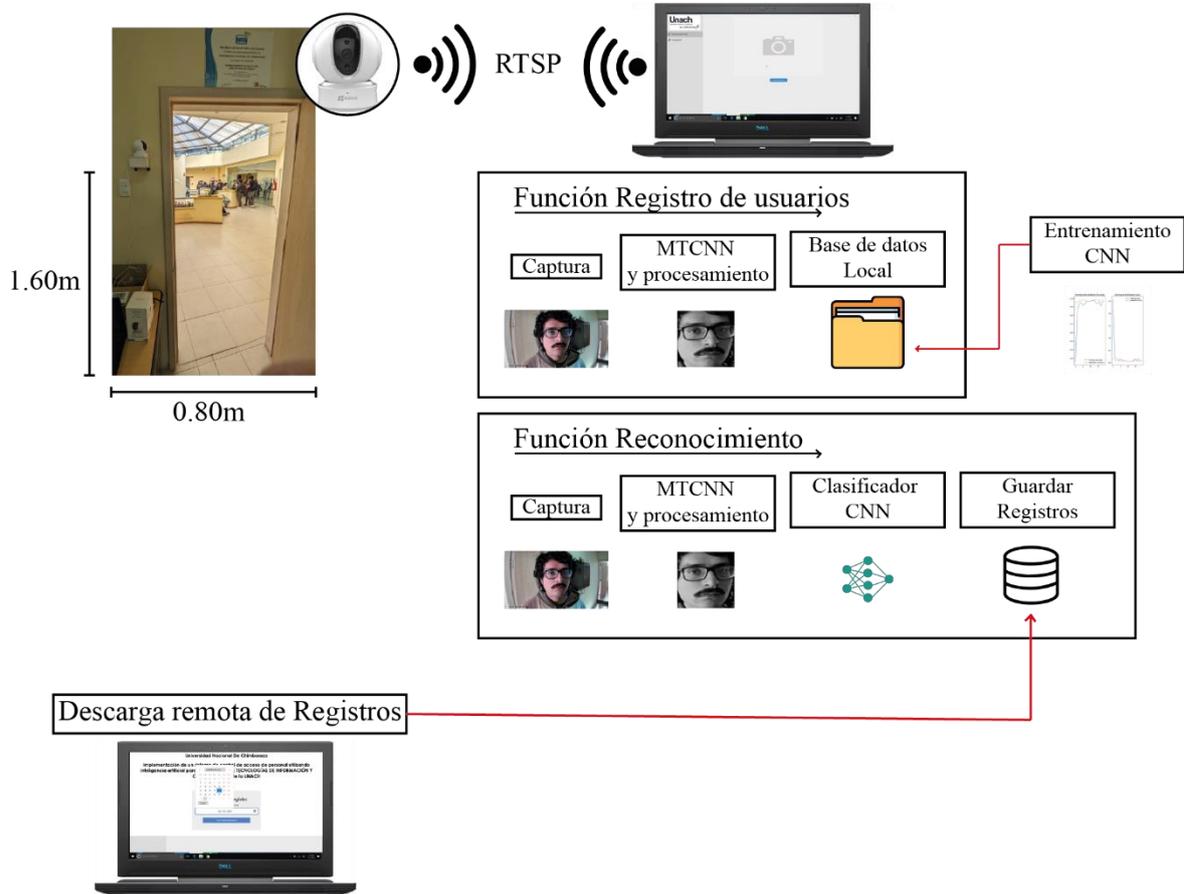


Figura 33: Arquitectura del sistema de control de acceso

Las funciones descritas en el capítulo anterior se resumen en la figura 33. Para la posición de la cámara se tomó en cuenta la altura de los trabajadores que realizan funciones en la bodega del departamento de redes de la DTIC, por lo que se decidió una altura estándar de 1.60m. Se puede notar que la distancia real que existe entre la cámara y la pared, que sería el rango donde se puede colocar los usuarios, es de 0.80m, que está dentro del rango elegido para realizar prueba en el funcionamiento del problema.

## 4.1.2 Gráfico de exactitud y pérdidas de la Red Neuronal Convolutiva implementada

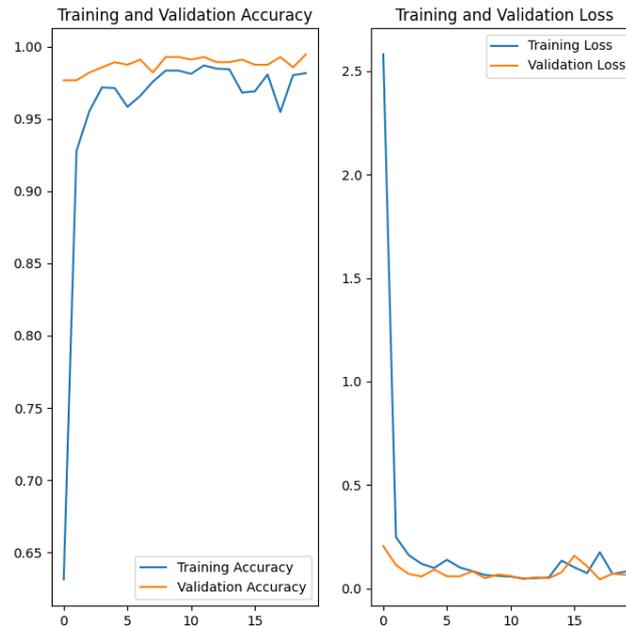


Figura 34: Gráfico de exactitud y pérdidas sobre el entrenamiento implementado

Esta gráfica muestra el desempeño de la red durante el entrenamiento de los datos a lo largo de las épocas fijadas. En la parte izquierda se tiene la exactitud del modelo, en esta la línea azul corresponde a la precisión en la clasificación de datos que la red haya visto en el conjunto de entrenamiento, mientras que la naranja corresponde a los datos que estén en el conjunto de validación, y no los haya visto durante el aprendizaje. En este reporte se puede observar que se tiene una precisión mayor del 95%, por lo que se puede considerar como favorable para este sistema. En la parte derecha se tiene las pérdidas, que es una métrica usada para medir cuánto difieren las predicciones de las etiquetas reales. En ambos casos, tanto en las pérdidas del conjunto de entrenamiento y de validación se puede observar que las pérdidas llegan a casi cero, por lo que sugiere un buen entrenamiento del modelo.

### 4.1.3 Estudio de porcentaje de aciertos

Primero se realizó un estudio del total de todos los experimentos realizados:

		<b>Acierto</b>			
		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	60	5,0	5,0	5,0
	1	1140	95,0	95,0	100,0
	Total	1200	100,0	100,0	

Tabla 3: Porcentaje de aciertos del total de datos obtenidos en la experimentación

En este primer análisis se puede observar que, del total de experimentos realizados, se tiene un 95% de aciertos en las predicciones de los usuarios. Con solo un 5% de errores, se considera como un rendimiento del sistema según la revisión bibliográfica revisada[40].

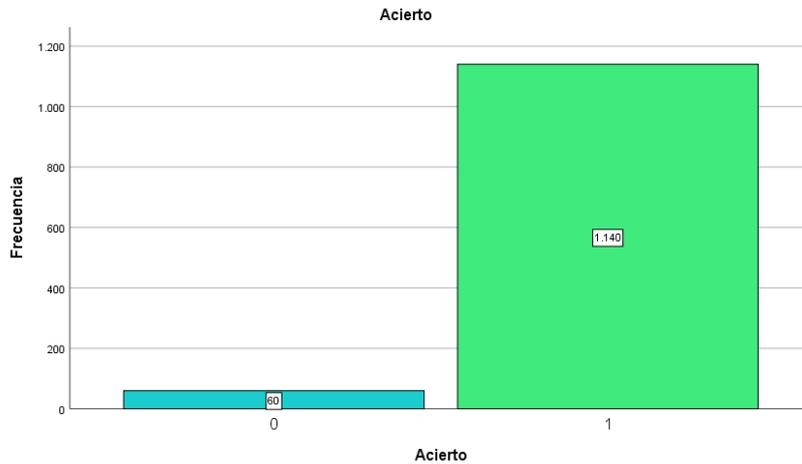


Figura 35: Diagrama de barras del porcentaje de aciertos del total de datos obtenidos en la experimentación

Al ver los resultados de la tabla 3 de manera gráfica con un diagrama de barras se puede corroborar que la distribución de aciertos es notoriamente mayor en los aciertos.

#### 4.1.3.1 Análisis del porcentaje de acierto con respecto a la variable distancia

A continuación, se analizan los resultados según los diferentes valores de la variable independiente distancia:

##### A05

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	,00	21	5,3	5,3	5,3
	1,00	379	94,8	94,8	100,0
Total		400	100,0	100,0	

Tabla 4: Porcentaje de aciertos obtenidos en la variable distancia a 0.5m

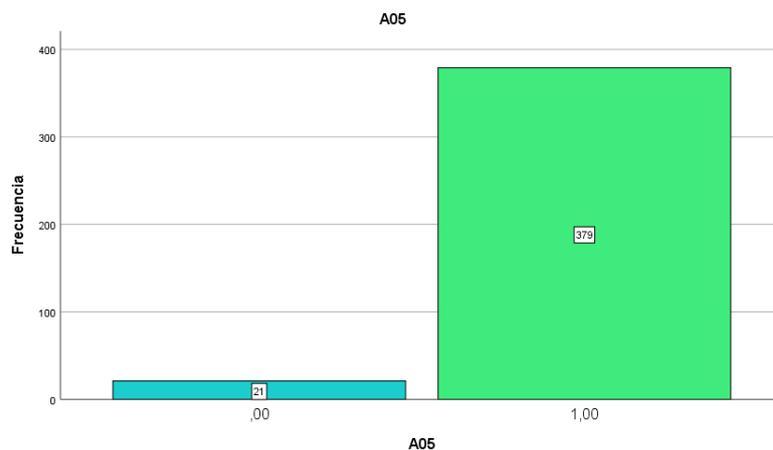


Figura 36: Diagrama de barras del porcentaje de aciertos obtenidos en la variable distancia a 0.5m

En el total de 400 datos recogidos con la distancia a 0.5 metros se obtuvo el 94.8% de aciertos. Se puede ver que a esta distancia se tiene un porcentaje aceptable de aciertos.

### A75

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	,00	28	7,0	7,0	7,0
	1,00	372	93,0	93,0	100,0
	Total	400	100,0	100,0	

Tabla 5: Porcentaje de aciertos obtenidos en la variable distancia a 0.75m

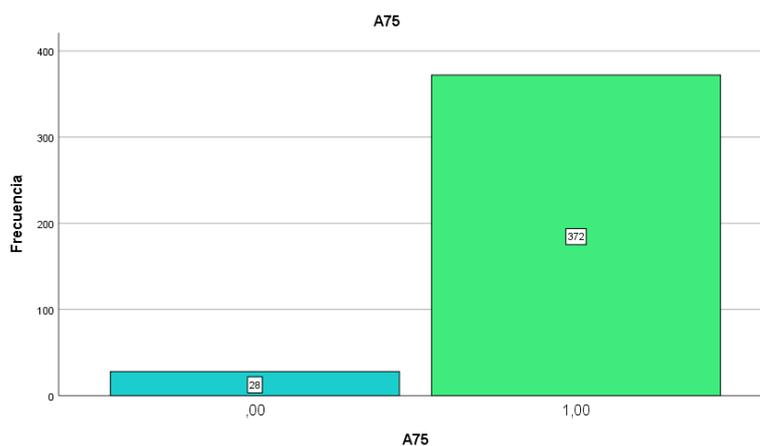


Figura 37: Diagrama de barras del porcentaje de aciertos obtenidos en la variable distancia a 0.75m

Con la distancia de 0.75 metros se obtiene un porcentaje de 93% por ciento de aciertos. Se puede notar que es un poco menor al obtenido por la distancia de 0.5 metros.

### A1

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	,00	11	2,8	2,8	2,8
	1,00	389	97,3	97,3	100,0
	Total	400	100,0	100,0	

Tabla 6: Porcentaje de aciertos obtenidos en la variable distancia a 1m

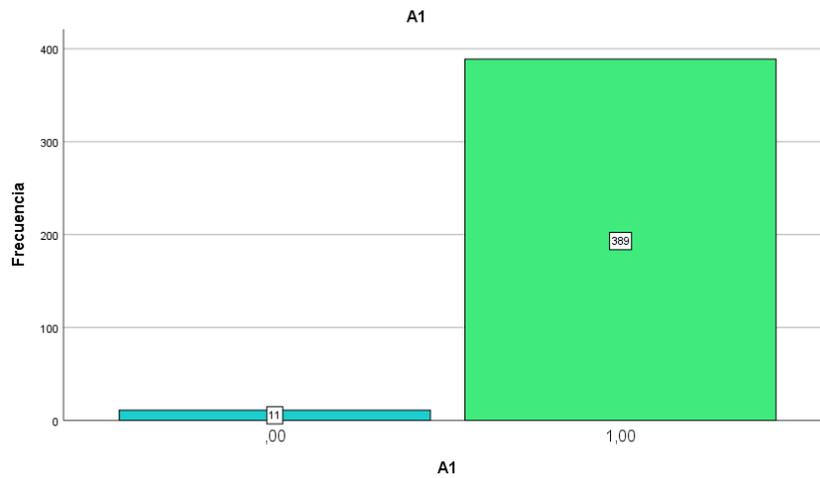


Figura 38: Diagrama de barras del porcentaje de aciertos obtenidos en la variable distancia a 1m

Con la distancia de 1 metro se obtiene un porcentaje de 97.3% de aciertos. Se nota que es la distancia con mayor porcentaje de aciertos en la experimentación de la variable.

#### 4.1.3.2 Análisis del porcentaje de acierto con respecto a la variable ángulo

A continuación, se analizan los resultados según los diferentes valores de la variable independiente ángulo:

**A0**

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	8	1,3	1,3	1,3
	1	592	98,7	98,7	100,0
	Total	600	100,0	100,0	

Tabla 7: Porcentaje de aciertos obtenidos en la variable ángulo a 0 grados

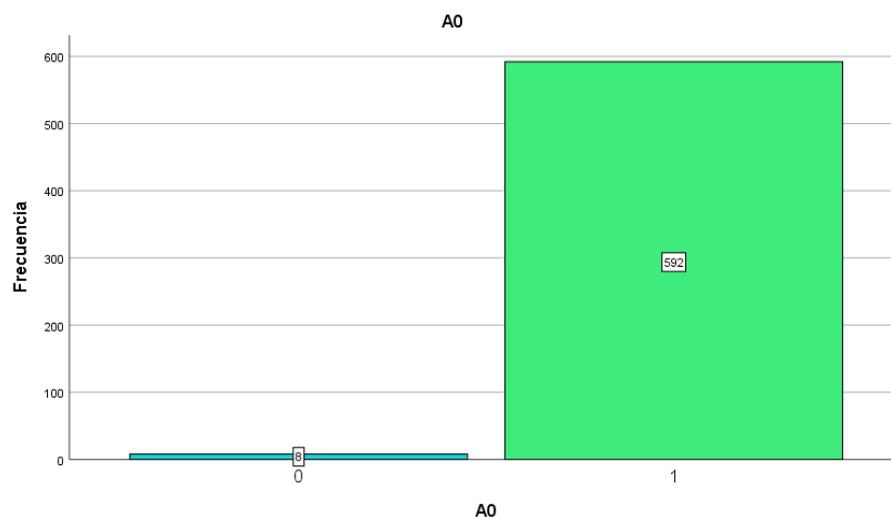


Figura 39: Diagrama de barras del porcentaje de aciertos obtenidos en la variable ángulo a 0 grados

De la totalidad de experimentos que se hicieron con un ángulo de 0 grados, se obtuvo un 98.7% de aciertos. Se puede corroborar con el diagrama de barras que es un porcentaje alto en comparación con los errores de apenas 1.3%.

### A45

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	52	8,7	8,7	8,7
	1	548	91,3	91,3	100,0
	Total	600	100,0	100,0	

Tabla 8: Porcentaje de aciertos obtenidos en la variable ángulo a 45 grados

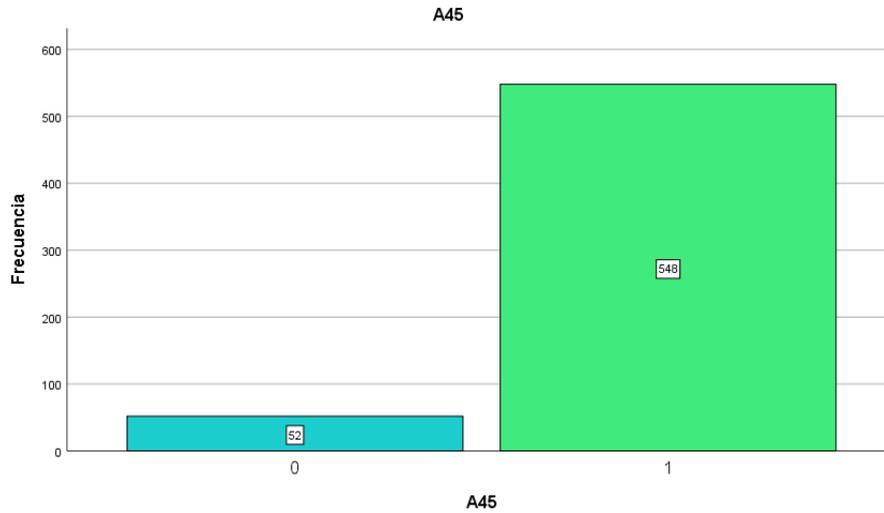


Figura 40: Diagrama de barras del porcentaje de aciertos obtenidos en la variable ángulo a 45 grados

Con una configuración de ángulo en 45 grados se obtuvo un 91.3% de aciertos. Se puede notar que es menor con respecto a la variable con 0 grados.

#### 4.1.3.3 Análisis del porcentaje de acierto con respecto a la variable iluminación

A continuación, se analizan los resultados según los diferentes valores de la variable independiente iluminación:

### Baja

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	32	5,3	5,3	5,3
	1	568	94,7	94,7	100,0
	Total	600	100,0	100,0	

Tabla 9: Porcentaje de aciertos obtenidos en la variable iluminación en iluminación baja

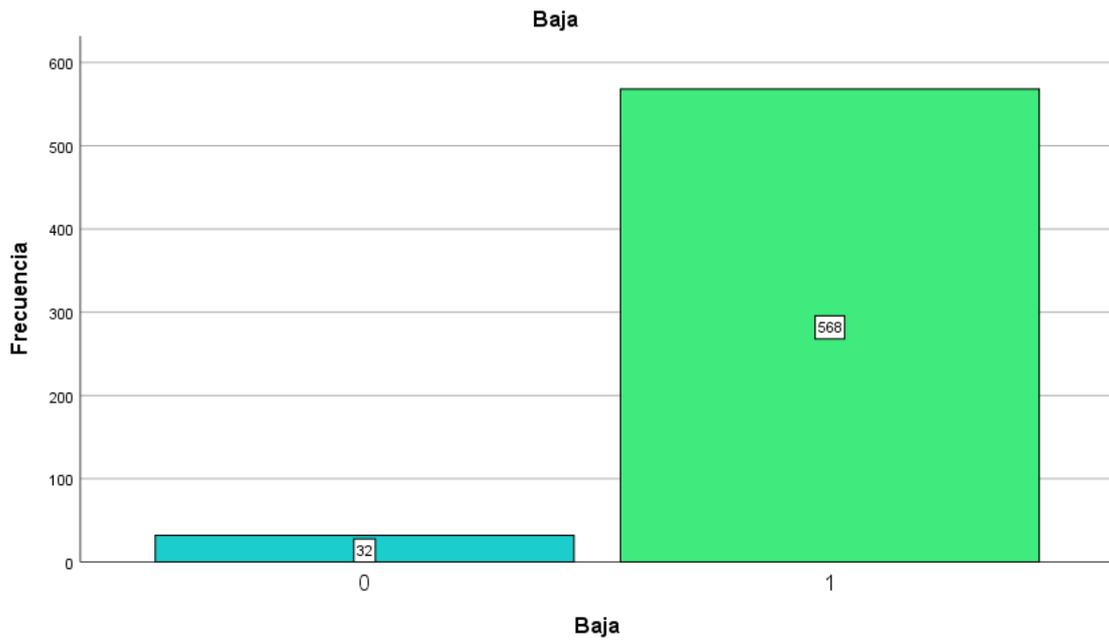


Figura 41: Diagrama de barras del porcentaje de aciertos obtenidos en la variable iluminación en iluminación baja

En el total de experimentaciones realizadas con una configuración de iluminación baja, se obtuvo un 94.7% de aciertos en las predicciones del sistema. Se nota que este valor todavía sigue estando en un rango aceptable de aciertos.

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	11	1,8	1,8	1,8
	1	589	98,2	98,2	100,0
	Total	600	100,0	100,0	

Tabla 10: Porcentaje de aciertos obtenidos en la variable iluminación en iluminación recomendada

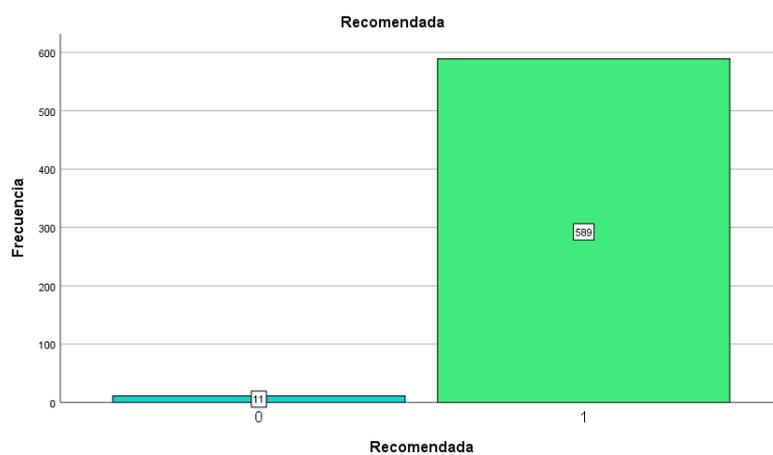


Figura 42: Diagrama de barras del porcentaje de aciertos obtenidos en la variable iluminación en iluminación recomendada

En la configuración de iluminación recomendada, se obtuvo un 98.2% de aciertos. Con apenas un 1.3% de errores, se nota que aquí hay más aciertos que en la configuración de iluminación baja.

#### 4.1.4 Estudio de relación entre porcentaje de aciertos y la variable distancias

A continuación, se realiza el estudio correspondiente a la hipótesis de la investigación.

##### 4.1.4.1 Estudio de normalidad de la variable Porcentaje de Aciertos a varias distancias

Se obtuvieron los siguientes descriptivos al analizar los porcentajes de acierto en cada distancia:

		Descriptivos			
Porcentaje de Aciertos	Distancia		Estadístico	Error estándar	
	05metros	Media	97,00	,997	
		95% de intervalo de confianza para la media	Límite inferior Límite superior	95,02 98,98	
		Media recortada al 5%	98,75		
		Mediana	100,00		
		Varianza	79,494		
		Desviación estándar	8,916		
		Mínimo	60		
		Máximo	100		
		Rango	40		
		Rango intercuartil	0		
	Asimetría	-3,321	,269		
	Curtosis	10,582	,532		
	075metros	Media	94,75	1,530	
		95% de intervalo de confianza para la media	Límite inferior Límite superior	91,70 97,80	
		Media recortada al 5%	97,50		
		Mediana	100,00		
		Varianza	187,278		
		Desviación estándar	13,685		
		Mínimo	40		
		Máximo	100		
Rango		60			
Rango intercuartil		0			
Asimetría	-3,381	,269			
Curtosis	11,063	,532			
1metro	Media	95,13	1,180		
	95% de intervalo de confianza para la media	Límite inferior Límite superior	92,78 97,47		
	Media recortada al 5%	96,94			
	Mediana	100,00			
	Varianza	111,377			
	Desviación estándar	10,554			
	Mínimo	50			
	Máximo	100			
	Rango	50			
	Rango intercuartil	10			
Asimetría	-2,685	,269			
Curtosis	7,133	,532			

Tabla 11: Descriptivos de la variable distancias en las distancias 0.5m, 0.75m y 1m

Con los descriptivos se puede notar que a la distancia de 0.5 metros se tiene una media del 97% de aciertos, con una varianza de 79.49% y una desviación estándar de 8.9% de aciertos. A la distancia de 0.75 metros se tiene una media del 94.75% de aciertos, con una varianza de 187.27% y una desviación estándar de 13.68% de aciertos. Con la distancia de 1 metro se tiene una media del 95.13% de aciertos, con una varianza de 111.37% y una desviación estándar de 10.55% de aciertos.

Se puede observar que en la distancia de 0.5 metros se obtiene una media más alta que en las demás configuraciones de la variable. En cambio, en la distancia de 0.75 metros se tiene una mayor varianza y desviación estándar, por lo intuye una mayor variabilidad en las predicciones a esta distancia. Para comprobar si estas variaciones son significativas, se precedió con el siguiente análisis:

### Pruebas de normalidad

Porcentaje de Aciertos	Distancia	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
		Estadístico	gl	Sig.	Estadístico	gl	Sig.
	05metros	,494	80	,000	,382	80	,000
	075metros	,412	80	,000	,423	80	,000
	1metro	,415	80	,000	,527	80	,000

Tabla 12: Prueba de normalidad Kolmogorov-Smirnov en la variable distancia

Con las pruebas de normalidad aplicadas a cada distancia se obtiene un p-valor menor a 0.05 por lo estas afirman que los datos no siguen una distribución normal. Esto se puede corroborar de forma gráfica con los siguientes histogramas:

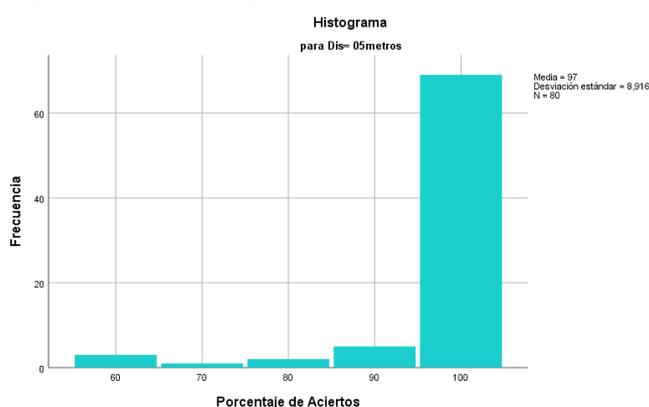


Figura 43: Histograma del porcentaje de aciertos a 0.5 metros

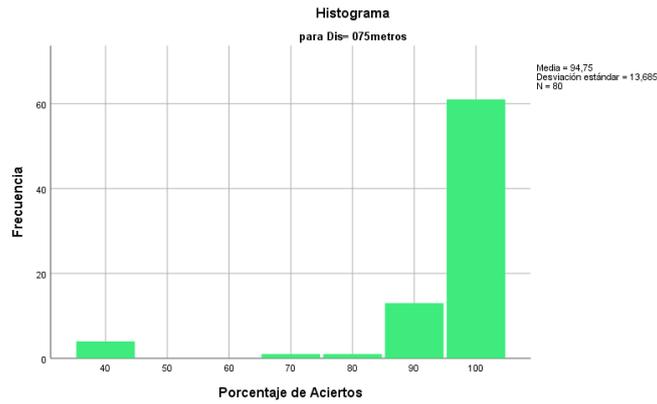


Figura 44: Histograma del porcentaje de aciertos a 0.75 metros

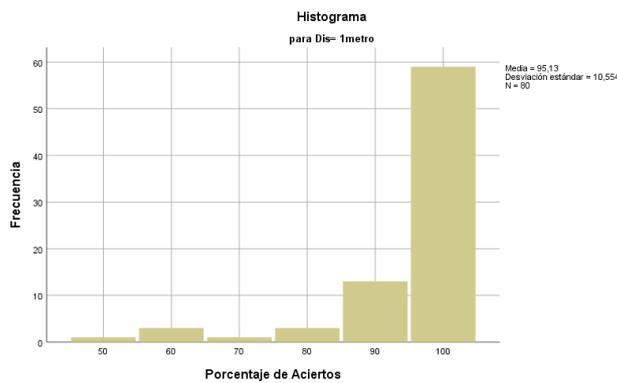


Figura 45: Histograma del porcentaje de aciertos a 1 metro

Se puede notar que en las tres distancias no se forma una campana de gauss, ya que la mayoría de datos están en el 100% de aciertos en los tres casos, por lo que se comprueba que no se sigue una distribución normal. Por lo tanto, para proceder con la hipótesis, y comprobar si hay una diferencia significativa en las distancias, se procede con el método no paramétrico de Kruskal-Wallis.

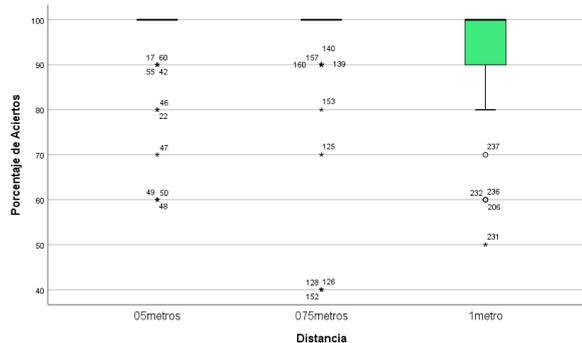


Figura 46: Diagrama de cajas de la variable distancia

En el diagrama de cajas de se puede notar que los datos, aparte de los valores atípicos, son tan cercanos a la mediana en las distancias de 0.5 metros y 0.75 metros que no se forma una caja en sí. En la distancia de 1 metro se tiene el mínimo en 80% y el primer cuartil a 90%, mientras que la mediana se encuentra en el 100% también.

#### 4.1.4.2 Prueba de Kruskal-Wallis

<b>Rangos</b>			
	Distancia	N	Rango promedio
Porcentaje de Aciertos	05metros	80	129,09
	075metros	80	117,68
	1metro	80	114,73
	Total	240	

Tabla 13: Rangos de la Prueba Kruskal-Wallis

Con los rangos promedio obtenidos de la prueba da una primera idea sobre la variabilidad en los porcentajes de acierto. Se puede notar que en los tres casos, se obtuvo un valores cercanos entre sí.

#### Estadísticos de prueba<sup>a,b</sup>

	Porcentaje de Aciertos
H de Kruskal-Wallis	3,752
gl	2
Sig. asintótica	,153

Tabla 14: Estadísticos de prueba de la Prueba Kruskal-Wallis

En el resultado de la prueba Kruskal-Wallis se obtuvo un p-valor de 0.153. Este es mayor a al nivel de significancia de 0.05, por lo que según esta prueba se sugiere que no hay una diferencia significativa entre los porcentajes de aciertos medidos a las distancias planteadas.

#### 4.1.4.3 Decisión de la hipótesis planteada

$$H_0: Me_{0.5m} = Me_{0.75m} = Me_{1m}$$

$$H_1: Me_{0.5m} \neq Me_{0.75m} \neq Me_{1m}$$

Con el p-valor de 0.153 obtenido la prueba no paramétrica de Kruskal-Wallis, al ser mayor que 0.05, no se rechaza la hipótesis nula. Por lo que se afirma en esta investigación que la mediana en los porcentajes de aciertos del sistema no varía significativamente en las tres distancias investigadas de 0.5 metros, 0.75 metros y 1 metro, con un 95% de confianza.

## 4.2 Discusión

En una primera investigación descriptiva del sistema se obtuvo porcentajes de acierto de 94.8%, 93% y 97.3% para las distancias de 0.5 metros, 0.75 metros y 1 metro respectivamente, por lo que se podría intuir que la mejor distancia para la implementación del sistema sería a 1 metro. Sin embargo, con la prueba no paramétrica de Kruskal-Wallis se pudo demostrar que el porcentaje de aciertos no varía significativamente según las distancias implementadas, con un 95% de confianza, por lo que no se puede afirmar que en alguna de estas tres distancias el sistema funciona mejor, pues los valores de porcentaje de aciertos variarían alrededor de los valores obtenidos.

En el caso de la variable ángulo se puede notar que existe un mayor porcentaje de aciertos en la configuración de 0 grados, con un 98.7% de aciertos. En el caso de la variable iluminación, se obtuvo un mayor porcentaje de aciertos en la iluminación recomendada, con

un 98.2% de aciertos. En este caso, según la investigación realizada, se puede atribuir estos valores a la calidad y variedad de datos de entrenamiento que estuvieron disponibles.

## CAPÍTULO V

### 5. CONCLUSIONES y RECOMENDACIONES

#### 5.1 Conclusiones

La investigación realizada sobre el estado del arte en sistemas biométricos con inteligencia artificial permitió la creación de un sistema de control de acceso que integra tecnologías de última generación, como el detector facial MTCNN y un diseño propio de una red neuronal convolucional profunda de 15 capas, que considera recomendaciones en la industria como métodos de aumento de datos y capas de dropout para lograr un buen nivel de abstracción en el aprendizaje, que pudo ser comprobado con las curvas de aprendizaje reportadas en la aplicación.

Se diseñó e implementó un prototipo en el área de redes de la Dirección de Tecnologías de la Información y Comunicación capaz de brindar una capa de seguridad a los equipos que se almacenan en sus instalaciones, el sistema permite la generación de reportes de entrada y salida del personal que permiten realizar un seguimiento en caso de pérdida de algún bien institucional.

La implementación de una página web desde la cual se pueda descargar remotamente los registros del sistema, construida con el lenguaje de programación php para el manejo de la información y html para la interfaz, se considera una decisión acertada pues esta brinda agilidad en su acceso a distancia que facilita a el director de la DTIC descargar esta información desde cualquier lugar.

Mediante el análisis de estadísticos descriptivos sobre los resultados en la experimentación de las variables independientes distancia, ángulo e iluminación se obtuvo que el porcentaje de aciertos varía entre el 93% y el 98.7% dependiendo de las configuraciones de las variables utilizadas. Estos resultados son alentadores, considerando que los sistemas biométricos que utilizan redes neuronales reportan una efectividad en sus predicciones entre 90% y 100%, según la literatura revisada[40], por lo que se puede confiar en la capa de seguridad que brinda el sistema al registro de esta área.

Con la prueba no paramétrica de Kruskal-Wallis se concluye que no hay una diferencia significativa en la mediana del porcentaje de aciertos entre las diferentes distancias consideradas para implementación del proyecto, a un 95% confianza, y habiéndose obtenido una mediana de 100% en los tres valores.

#### 5.2 Recomendaciones

Es importante verificar la lista de usuarios entrenados después de cada registro al sistema, pues para su correcto funcionamiento la red debe volver a entrenarse cuando hay nuevos usuarios.

Para la creación de la base de datos local para cada usuario en el sistema debe considerarse mantener un área despejada, para así evitar tener imágenes erróneas dentro de la carpeta usada en el entrenamiento y que este se realice erróneamente. De la misma manera, es importante, al momento de utilizar la función de reconocimiento, mantener el área de la frente sin obstáculos, pues se pudo observar dificultad en el detector facial MTCNN localizar rostros en estas condiciones durante la experimentación.

Se recomienda mantener un valor de iluminación dentro del rango sugerido para una oficina, 500 -1000 lux, para garantizar condiciones favorables de captura en los rostros. Este rango ha sido seleccionado para mantener la calidad de las imágenes y, por lo tanto, minimizar posibles desafíos relacionados con la variabilidad en las condiciones de iluminación.

Para garantizar un rendimiento óptimo, se recomienda ejecutar la aplicación en un computador con hardware moderno, enfocándose en una GPU con una cantidad de memoria mayor a la utilizada en este trabajo. Esto ya que durante la experimentación se observó un retraso en el video capturado en tiempo real, debido al esfuerzo computacional requerido para las operaciones de procesamiento que constantemente se realiza. Al seleccionar un hardware más avanzado, se espera mitigar este problema, mejorando la fluidez de la aplicación.

## 6. BIBLIOGRAFÍA

- [1] A. I. Cerezo Domínguez and J. L. Díez Ripollés, “La videovigilancia en las zonas públicas; su eficacia en la reducción de la delincuencia,” *Boletín Criminológico*, no. 95, 2010, doi: 10.24310/boletin-criminologico.2010.v16i0.8434.
- [2] D. Platero, “RECONOCIMIENTO DE IMÁGENES FACIALES ORIENTADO A CONTROLES DE ACCESO Y SISTEMAS DE SEGURIDAD.,” UNIVERSIDAD DISTRITAL “FRANCISCO JOSÉ DE CALDAS,” 2015.
- [3] J. R. Bastidas Gavilanes, “Registro De Asistencia De Alumnos Por Medio De Reconocimiento Facial Utilizando Visión Artificial,” p. 125, 2019, [Online]. Available: <https://repositorio.uta.edu.ec/jspui/handle/123456789/29179>
- [4] M. M. C. Mora and C. J. Sanz, “Sistemas Biométricos:,” *Asp. Soc. en la Secur. Ciudad.*, pp. 147–158, 2021, doi: 10.2307/j.ctv1s7chmp.17.
- [5] A. Gonzáles, F. Martínez, A. Pernia, M. Castejón, J. Orières, and E. Vergara, *Técnicas y Algoritmos Básicos de Visión Artificial*, vol. 4, no. 1. Universidad de la Rioja, 2006.
- [6] M. Vazquez, “Sistema de Reconocimiento Facial Mediante Técnicas de Visión Tridimensional,” 2014.
- [7] P. Pereyra and J. Donayo, “Reconocimiento Facial Mediante Imágenes Estereoscópicas Para Control de Ingreso,” UNIVERSIDAD DE BUENOS AIRES, 2010. [Online]. Available: [https://www.academia.edu/download/51999480/TESIS\\_LUIS.pdf](https://www.academia.edu/download/51999480/TESIS_LUIS.pdf)
- [8] A. Abeliuk and C. Gutiérrez, “Historia y Evolución de la Inteligencia Artificial,” *Rev. del Dep. Ciencias la Comput. la Univ. Chile*, vol. 21, pp. 14–21, 2021.
- [9] A. F. Garcés Núñez, “Sistema De Reconocimiento Facial Con Visión Artificial Para Apoyar Al Ecu-911 Con La Identificación De Personas En La Lista De Los Más Buscados,” *Repo.Uta.Edu.Ec*, p. 148, 2017, [Online]. Available: <http://repo.uta.edu.ec/bitstream/handle/123456789/5301/Mg.DCEv.Ed.1859.pdf?sequence=3>
- [10] D. Fernando, L. Delgado, O. Andrés, and L. Pabón, “Reconocimiento Facial Para La Automatización Del Registro De Asistencia a Clases,” 2022.
- [11] C. Valenzuela Urra, D. Reyes Lillo, and S. Oliveros Castro, “Introducción: Software libre y código abierto: experiencias innovadoras en bibliotecas y centros de información,” *Palabra Clave (La Plata)*, vol. 8, no. 1, p. e054, 2018, doi: 10.24215/18539912e054.
- [12] K. A. Majeed, Z. Abbas, M. Bakhtyar, and Z. Durrani, “Face Detectors Evaluation to Select the Fastest among DLIB , HAAR Cascade , and MTCNN,” vol. 1, no. 1, 2021.
- [13] D. Wang, H. Yu, D. Wang, and G. Li, “Face recognition system based on CNN,” *Proc. - 2020 Int. Conf. Comput. Inf. Big Data Appl. CIBDA 2020*, pp. 470–473, 2020, doi: 10.1109/CIBDA50819.2020.00111.
- [14] M. Mann and M. Smith, “AUTOMATED FACIAL RECOGNITION FOR CLASS ATTENDANCE,” *Int. J. Recent Dev. Sci. Technol.*, vol. 06, no. 06, pp. 168–173, 2022, [Online]. Available: [http://au.nec.com/en\\_AU/press/201509/nec-facial-recognition-increases-public-safety-](http://au.nec.com/en_AU/press/201509/nec-facial-recognition-increases-public-safety-)
- [15] M. M. Plá, “Biometría Facial con Deep Learning,” Universitat Oberta DE CATALUNYA, 2023.
- [16] F. Recognition, *Handbook of Face Recognition*. 2011. doi: 10.1007/978-0-85729-932-1.
- [17] H. Gonzalez and S. Velásquez, “Reconocimiento Facial Utilizando Viola-Jones Y Patrones Binarios Facial Recognition Using Viola-Jones and Binary Patterns,” *Nº*, vol. 23, pp. 57–63, 2019.

- [18] A. Adouani, W. M. Ben Henia, and Z. Lachiri, "Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences," *16th Int. Multi-Conference Syst. Signals Devices, SSD 2019*, pp. 266–271, 2019, doi: 10.1109/SSD.2019.8893214.
- [19] S. O. Adeshina, H. Ibrahim, S. S. Teoh, and S. C. Hoo, "Custom face classification model for classroom using haar-like and lbp features with their performance comparisons," *Electron.*, vol. 10, no. 2, pp. 1–15, 2021, doi: 10.3390/electronics10020102.
- [20] B. Niu, Z. Gao, and B. Guo, "Facial Expression Recognition with LBP and ORB Features," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–10, Jan. 2021, doi: 10.1155/2021/8828245.
- [21] C. Rahmad, R. A. Asmara, D. R. H. Putra, I. Dharma, H. Darmono, and I. Muhiqqin, "Comparison of Viola-Jones Haar Cascade Classifier and Histogram of Oriented Gradients (HOG) for face detection," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 732, no. 1, 2020, doi: 10.1088/1757-899X/732/1/012038.
- [22] M. S. Minu, K. Arun, A. Tiwari, and P. Rampuria, "Face recognition system based on haar cascade classifier," *Int. J. Adv. Sci. Technol.*, vol. 29, no. 5, pp. 3799–3805, 2020.
- [23] S. Sharma, "Face Mask Detection using MobileNetV2," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. 5, pp. 601–606, 2021, doi: 10.22214/ijraset.2021.34227.
- [24] H. Naufal, A. Sunarko, R. Hidayat, and R. Hartanto, "Comparative Analysis of Masked and Unmasked for Face Recognition Using VGG Face and MTCNN," Yogyakarta, 2023. doi: 10.4108/eai.5-10-2022.2327473.
- [25] V. D. Rangari, "Dominant Facial Emotion Recognition Using Mtcnn in Python," *Int. Res. J. Mod. Eng. Technol. Sci.*, no. 10, pp. 650–653, 2022, doi: 10.56726/irjmets30585.
- [26] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, 2016, doi: 10.1109/LSP.2016.2603342.
- [27] L. Rouhiainen, "Inteligencia artificial 101," *Alienta Editor.*, p. 352, 2018, [Online]. Available: [https://planetadelibrosar0.cdnstatics.com/libros\\_contenido\\_extra/40/39307\\_Inteligencia\\_artificial.pdf](https://planetadelibrosar0.cdnstatics.com/libros_contenido_extra/40/39307_Inteligencia_artificial.pdf)
- [28] C. Zhang and Y. Lu, "Study on artificial intelligence: The state of the art and future prospects," *J. Ind. Inf. Integr.*, vol. 23, no. April, p. 100224, 2021, doi: 10.1016/j.jii.2021.100224.
- [29] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [30] J. C. Ávalos, "Introducción al Deep Learning," 2022. [Online]. Available: <https://iaarbook.github.io/deeplearning/>
- [31] Y. Chen, L. Song, Y. Liu, L. Yang, and D. Li, "A review of the artificial neural network models for water quality prediction," *Appl. Sci.*, vol. 10, no. 17, 2020, doi: 10.3390/app10175776.
- [32] N. Kriegeskorte and T. Golan, "Neural network models and deep learning," *Curr. Biol.*, vol. 29, no. 7, pp. R231–R236, 2019, doi: 10.1016/j.cub.2019.02.034.
- [33] G. Portilla, "LAS REDES NEURONALES APLICADO EN EL APRENDIZAJE DE LA ROBÓTICA EDUCATIVA," UNIVERSIDAD TÉCNICA DEL NORTE, 2023.
- [34] F. Bueno, "Redes Neuronales: Entrenamiento y Comportamiento," Universidad Complutense Madrid, 2019. [Online]. Available: <https://eprints.ucm.es/id/eprint/64564/1/BUENOPASCUALFERNANDO.pdf>

- [35] A. E. Gamonal Chanco, “Diseño de un sistema por visión artificial para determinar la calidad de mandarinas,” *Repos. Inst. - UTP*, 2020, [Online]. Available: <http://repositorio.utp.edu.pe/handle/20.500.12867/3986>
- [36] H. Chanampe *et al.*, “Modelo de redes neuronales convolucionales profundas para la clasificación de lesiones en ecografías mamarias,” *XXI Work. Investig. en Ciencias la Comput.*, no. Cidi, pp. 971–974, 2019, [Online]. Available: <http://sedici.unlp.edu.ar/handle/10915/77381>
- [37] A. Cifuentes, E. Mendoza, A. Santrich, and S. Moreno, “Desarrollo de una red neuronal convolucional para reconocer patrones en imágenes,” *Investig. y Desarro. en TIC*, vol. 10, no. 2, pp. 7–17, 2019, [Online]. Available: <https://revistas.unisimon.edu.co/index.php/identific/article/view/4007>
- [38] S. Dong, P. Wang, and K. Abbas, “A survey on deep learning and its applications,” *Comput. Sci. Rev.*, vol. 40, p. 100379, 2021, doi: 10.1016/j.cosrev.2021.100379.
- [39] M. Grau, “Clasificador de modelos de movilidad mediante Deep Learning,” Universidad de Sevilla, 2020.
- [40] J. García, S. Hurtado, and M. Vidal, “Revisión de la Literatura: Método Automatizado De Reconocimiento Facial Basado En Algoritmos De Aprendizaje,” *Rev. Ibérica Sist. e Tecnol. Informação*, pp. 334–347, 2023.
- [41] J. Elias, M. Reyes, C. Stephano, and C. Rodríguez, “Sistema de reconocimiento facial para el control de accesos mediante Inteligencia Artificial Facial recognition system for access control through Artificial Intelligence,” vol. 4, no. 1, pp. 24–36, 2023.
- [42] E. Bash, “Real Time Streaming Protocol (RTSP) Status,” *PhD Propos.*, vol. 1, no. April, pp. 1–92, 1998.
- [43] A. Piñeda and G. Montes, “Ergonomía Ambiental: Iluminación y confort térmico en trabajadores de oficinas con pantalla de visualización de datos,” *Rev. Ing. Matemáticas y Ciencias la Inf.*, vol. 1, no. 2, pp. 49–71, 2014, [Online]. Available: <http://ojs.urepublicana.edu.co/index.php/ingenieria/article/view/228>

## 7. ANEXOS

### 7.1 Anexo 1: Interfaces de la aplicación

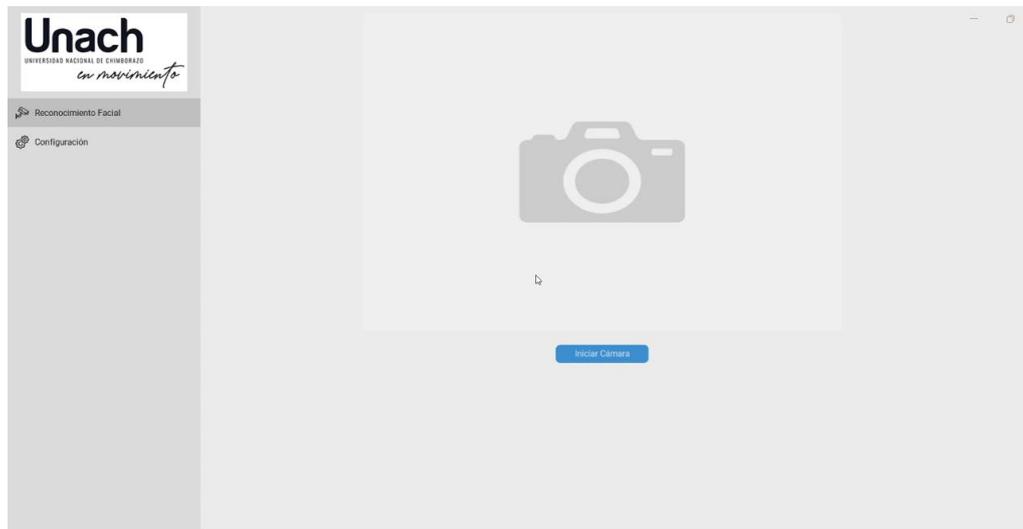


Figura 47: Interfaz principal de la aplicación

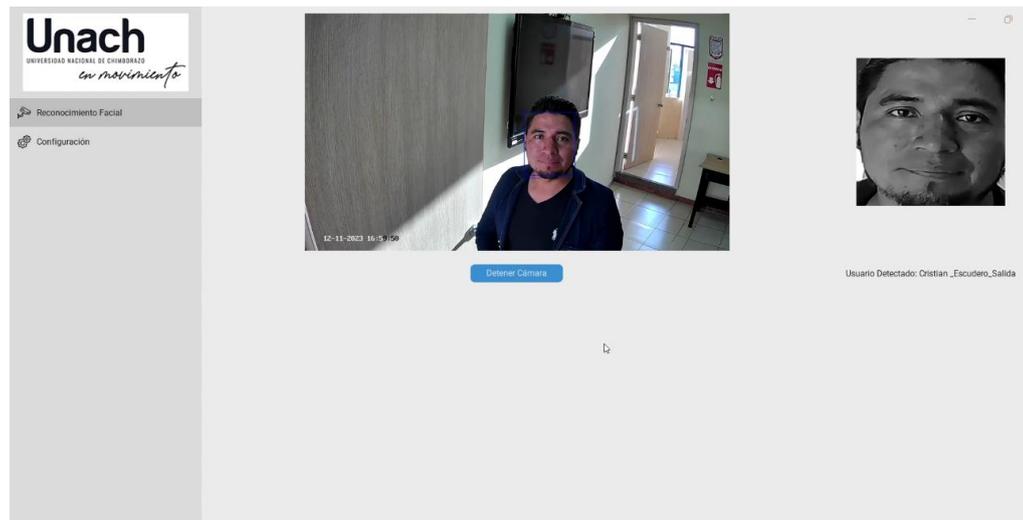


Figura 48: Interfaz de la función de reconocimiento

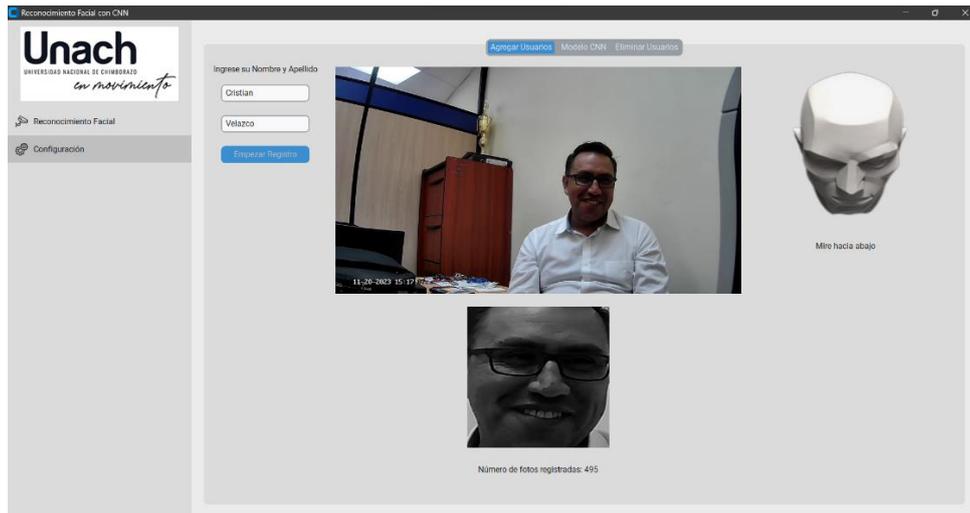


Figura 49: Interfaz del registro de usuarios

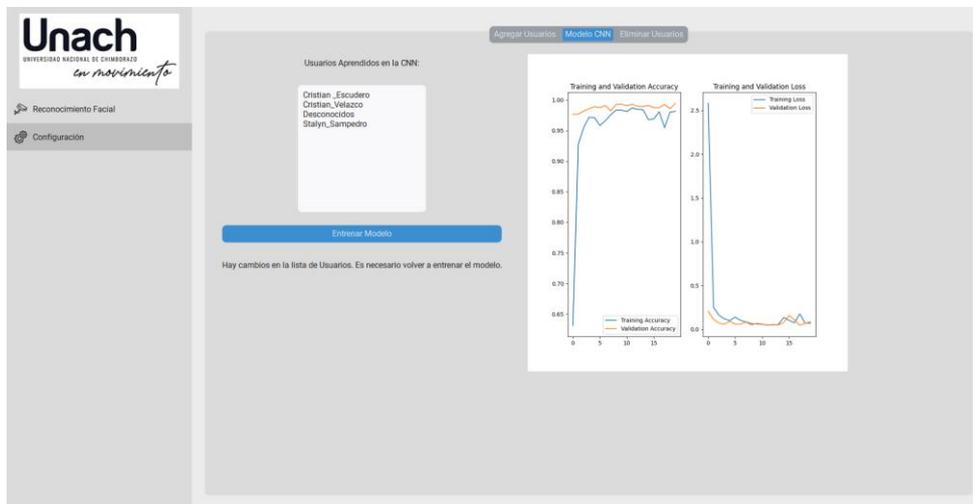


Figura 50: Interfaz del entrenamiento del modelo

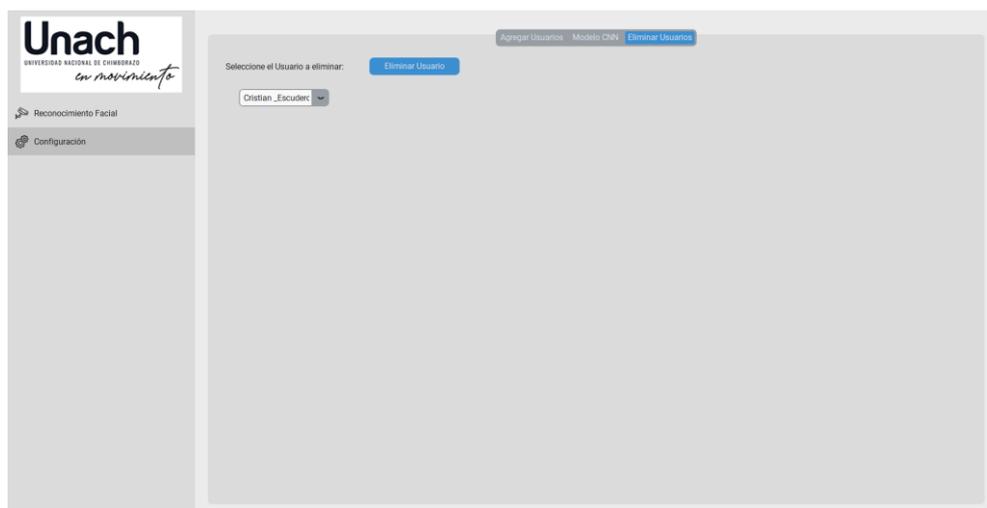


Figura 51: Interfaz de la eliminación de usuarios



Figura 52: Interfaz de la página web implementada

## 7.2 Anexo 2: Evidencias de la experimentación



Figura 53: Registro de trabajadores de la DTIC en el sistema



Figura 54: Ejecución de las pruebas experimentales con los trabajadores de la DTIC



Figura 55: Implementación sugerida para la cámara del sistema.

### 7.3 Anexo 3: Compromiso de emitido por la Dirección de Tecnologías de la Información y Comunicación



Dirección de Tecnologías de la  
Información y Comunicación  
VICERRECTORADO ADMINISTRATIVO

*en movimiento*

Oficio No. 01714-DTIC-UNACH-2022  
Riobamba, 16 de diciembre, 2022

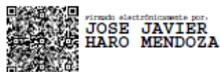
Ingeniera  
Deysi Inca Balseca  
**DIRECTORA DE LA CARRERA DE ELECTRÓNICA y TELECOMUNICACIONES**  
**FACULTAD DE INGENIERÍA**  
Presente

De mi consideración:

Con el debido comedimiento, tengo a bien poner en su conocimiento que la DTIC, se compromete a brindar las facilidades para que el señor Stalyn Enrique Sampedro Altamirano, estudiante de la carrera de Ingeniería en Telecomunicaciones realice la implementación del proyecto de tesis titulado: "Implementación de un sistema de control de acceso de personal utilizando inteligencia artificial para la Dirección de Tecnologías de la Información y Comunicación de la UNACH"

Particular que le comunico para los fines legales pertinentes.

Atentamente,



Ing. Javier Haro Mendoza  
**DIRECTOR DE TECNOLOGÍAS DE  
LA INFORMACIÓN y COMUNICACIÓN**

NFB.-.-.-

## 7.4 Anexo 4: Capturas de la aplicación Lux Meter

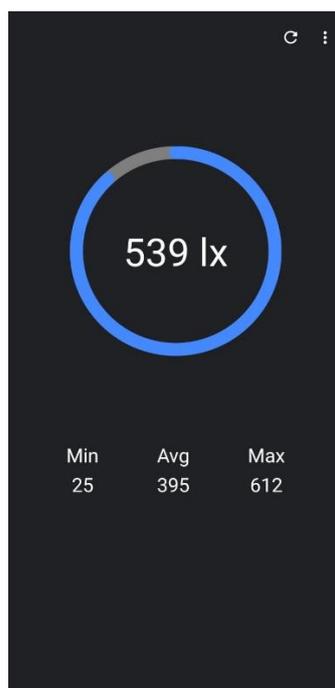


Figura 56: Iluminación para la experimentación en el rango recomendado para oficinas

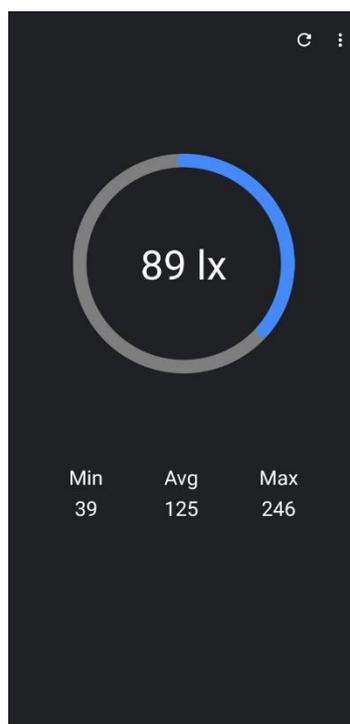


Figura 57: Iluminación para la experimentación en el rango bajo para oficinas

## 7.5 Anexo 5: Datos recogidos en la experimentación

A05	A75	A1
1,00	1,00	1,00



1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	0,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00



1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	0,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00

1,00	1,00	1,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	0,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00

1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00

1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00

1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
1,00	0,00	0,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
1,00	0,00	1,00
1,00	0,00	1,00
0,00	0,00	0,00
1,00	0,00	0,00
1,00	0,00	0,00
1,00	1,00	0,00
1,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00

1,00	0,00	1,00
1,00	1,00	1,00
1,00	1,00	0,00
1,00	0,00	1,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
0,00	0,00	1,00
0,00	1,00	1,00
0,00	0,00	1,00
1,00	0,00	1,00
0,00	1,00	1,00
1,00	0,00	1,00
1,00	1,00	1,00
1,00	0,00	1,00
0,00	0,00	1,00
1,00	0,00	1,00
0,00	0,00	1,00
1,00	0,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
1,00	1,00	1,00
0,00	1,00	1,00
0,00	1,00	1,00
0,00	1,00	1,00
1,00	1,00	1,00

Tabla 15: Base de datos de la variable distancias

Baja	Recomendada
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1





1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	1













1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
0	1
1	1
1	1
0	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1

1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
0	1
0	1
0	1
1	1
1	1
1	1
1	1
0	1
0	1
0	1
1	1
1	1
1	1
0	1
0	1
0	1
1	1
1	1

1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
0	1
1	1
1	1
0	1
0	1
0	1
1	1
1	1
1	1
0	1
1	1
1	1



1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1



1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1

1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	0
1	1
1	0
1	1
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	0
1	0
1	1
1	0
1	1
1	1
1	1
1	0
1	1
1	0
1	1
1	0
1	0
1	1



1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1



1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	0
1	0
0	1
1	1
1	0
1	0
1	0
1	0
1	0
1	0
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	0
1	1
1	1
1	0
1	0
1	1
1	0
1	0

1	1
1	0
1	0
1	1
1	0
1	1
1	0
1	0
1	0
0	0
1	0
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1



1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
0	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
0	1
0	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1
1	1





100	90	90
100	100	100
100	100	100
100	90	100
90	100	100
100	100	100
100	100	100
100	100	100
100	100	100
100	100	100
100	100	100
100	100	100
90	100	100
100	100	100
100	100	90
100	100	100
100	100	100
80	100	90
100	100	100
100	100	90
100	100	100
100	90	100
100	90	100
100	90	90
100	100	100
100	100	100
100	100	100
100	100	100
100	100	90
100	100	100
100	100	100
100	100	100
100	100	100
100	100	80
100	90	100
100	100	100
100	100	100
100	90	100
90	100	100
100	100	100
100	100	100
100	70	90
80	40	60
70	90	100
60	40	90
60	40	100
60	100	100
100	100	100

100	100	100
100	100	90
100	100	100
90	100	100
100	100	100
100	100	100
100	90	100
100	90	100
90	90	100
100	100	90
100	100	100
100	100	100
100	100	100
100	100	100
100	100	100
100	100	90
100	100	90
100	100	100
100	100	90
100	100	50
100	40	60
100	80	100
100	100	100
100	100	100
100	100	60
100	90	70
100	100	80
100	100	100
100	90	100

Tabla 18: Base de datos del porcentaje de aciertos en las distancias 0.5 metros, 0.75 metros y 1 metro