Desarrollo de un sistema de control de acceso de personal empleando reconocimiento facial respaldado con técnicas de aprendizaje profundo

Muñoz Vega, Edison Paúl

Departamento de Eléctrica y Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica, Automatización y Control

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica, Automatización y Control

Dr. Flores Calero, Marco Javier

29 de julio del 2021

Curiginal

Document Information

Analyzed document	Tesis-final_cap3.pdf (D110827675)
Submitted	7/28/2021 10:21:00 PM
Submitted by	
Submitter email	myflores@espe.edu.ec
Similarity	3%
Analysis address	myflores.espe@analysis.urkund.com

Sources included in the report

SA	**Universidad de las Fuerzas Armadas ESPE / Proyecto_de_grado_William_Ibarra.pdf** Document Proyecto_de_grado_William_Ibarra.pdf (D78418907) Submitted by: gfolmedo@espe.edu.ec Receiver: gfolmedo.espe@analysis.urkund.com	14
SA	**Universidad de las Fuerzas Armadas ESPE / Tesis - Sistema de reconocimiento facial y gafas espia - Alexander Lascano Pedro Pico.pdf** Document Tesis - Sistema de reconocimiento facial y gafas espia - Alexander Lascano Pedro Pico.pdf (D95335130) Submitted by: papico@espe.edu.ec Receiver: wgaguilar.espe@analysis.urkund.com	4
W	URL: http://190.169.30.62/bitstream/123456789/14700/1/TEG%20-%20De%20Sousa%2C%20Mora.pdf Fetched: 12/6/2020 2:12:28 PM	1

Dr. Flores Calero, Marco Javier

Profesor Titular Principal de la tesis

C.C: 0502198757

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

CERTIFICACIÓN

Certifico que el trabajo de titulación "**Desarrollo de un sistema de control de acceso de personal empleando reconocimiento facial respaldado con técnicas de aprendizaje profundo"** fue realizado por el señor **Muñoz Vega, Edison Paúl,** el cual ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito autorizar para que lo sustente públicamente.

Sangolquí, 29 de julio de 2021

Firma:

Dr. Flores Calero, Marco Javier

C.C: 0502198757

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

RESPONSABILIDAD DE AUTORÍA

Yo, **Muñoz Vega, Edison Paúl,** con cédula de ciudadanía n° 1721979688, declaro que el contenido, ideas y criterios del trabajo de titulación: **"Desarrollo de un sistema de control de acceso de personal empleando reconocimiento facial respaldado con técnicas de aprendizaje profundo"** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 29 de julio de 2021

Firma:

Muñoz Vega, Edison Paúl

C.C: 1721979688

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

AUTORIZACIÓN DE PUBLICACIÓN

Yo, **Muñoz Vega, Edison Paúl,** con cédula de ciudadanía n° 1721979688, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: "**Desarrollo de un sistema de control de acceso de personal empleando reconocimiento facial respaldado con técnicas de aprendizaje profundo"** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 29 de julio de2021

Firma:

Muñoz Vega, Edison Paúl

C.C: 1721979688

DEDICATORIA

A Dios y a mis padres con su amor, paciencia y esfuerzo me han permitido llegar a cumplir hoy un sueño más, gracias por todo. A mis amigos, por apoyarme cuando más los necesito, por extender su mano en momentos difíciles y por el cariño brindado cada día, de verdad mil gracias muchachos, siempre los llevo en mi corazón. Finalmente quiero dedicar este trabajo de tesis a mi hija Rebeca, mis padres me enseñaron como estar preparado para enfrentar la vida, tu hija mía me enseñaste como enfrentar la vida, sin estar preparado, Te amo.

Edison Paúl Muñoz Vega

Agradecimiento

Quiero expresar mi gratitud a Dios, quien con su bendición llena siempre mi vida y a toda mi familia por estar siempre presentes. Mi profundo agradecimiento a todas las autoridades y personal que hacen la Universidad de las Fuerzas Armadas ESPE, por tener su confianza en mí, abrirme las puertas y formarme a los largo de todos estos años dentro de su establecimiento educativo. De igual manera mis agradecimientos a mis padres, que con su esfuerzo supieron encaminarme siempre ya que con su paciencia y enseñanza lograron hoy ayudarme a cumplir este sueño. Finalmente quiero expresar mi más grande y sincero agradecimiento a todos mis amigos que siempre estuvieron ahí para apoyarme incondicionalmente, aportando en mi crecimiento personal y profesional.

Edison Paúl Muñoz Vega

Índice de Contenidos

Índice de Tablas

Índice de Figuras

Resumen

El presente proyecto de investigación consiste en el desarrollo de un sistema de detección y reconocimiento facial automatizado por medio de técnicas de aprendizaje profundo, mejorado como una aplicación para el registro de personal al ingresar a la zona o área de trabajo, a través de la visión por computadora, un sub-campo de aplicación de la Inteligencia artificial. Para detectar y alinear el rostro se utilizó algoritmos de aprendizaje profundo como MTCNN y un modelo desarrollado por el marco de trabajo MXNET, denominado RetinaFace pero al ser multiplataforma se lo usará en PyTorch, estos modelos nos permiten ajustar a los usuarios al sistema para realizar el control de identificación por medio de reconocimiento facial. Una vez detectado el rostro mediante los modelos antes mencionados, se obtiene las coincidencias más cercanas a los rostros ajustados al sistema realizando el reconocimiento y obteniendo su registro de ingreso/ salida de su área de trabajo, este proceso lo realiza Facenet, el cual es un modelo pre entrenado que permite identificar al usuario registrado en el sistema. Se agregó un algoritmo de anti-plagio, denominado algoritmo de la vida, el cual cumple la función de detectar la imitación de rostros y evitar la vulnerabilidad al momento de reconocer a una persona ajustada al sistema mediante una foto o un video. Se desarrolla una interfaz web con un servidor local denominado Flask, Cabe mencionar que todo el sistema de reconocimiento facial se lo realiza en tiempo real y con un par de fotos por cada usuario registrado, siendo esta característica fundamental ya que no se necesita un set grande de fotos por cada usuario que se registre al sistema.

PALABRAS CLAVE:

- **MTCNN**
- **MXNET**
- **RETINAFACE**
- **FACENET**
- **FLASK**

Abstract

This research project consists of the development of an automated facial detection and recognition system through deep learning techniques, improved as an application for the registration of personnel when entering the area or work area, through vision by computer, a subfield of application of Artificial Intelligence. To detect and align the face, deep learning algorithms such as MTCNN and a model developed by the MXNET framework, called RetinaFace, were used but being multiplatform it will be used in PyTorch, these models allow us to adjust users to the system to perform the Identification control through facial recognition. Once detected through the aforementioned models, the closest matches to the faces adjusted to the system are obtained by performing the recognition and obtaining their entry / exit record from their work area, this process is carried out by FaceNet, which is a model pre-trained that allows to identify the user registered in the system. An anti-plagiarism algorithm was added, called the algorithm of life, which fulfills the function of detecting the imitation of faces and avoiding vulnerability when recognizing a person adjusted to the system through a photo or video. A web interface is developed with a local server called Flask. It is worth mentioning that the entire facial recognition system is performed in real time and with a couple of photos for each registered user, this being a fundamental characteristic since a large set is not needed of photos for each user who registers to the system.

KEYWORDS:

- **MTCNN**
- **MXNET**
- **RETINAFACE**
- **FACENET**
- **FLASK**

Capítulo I

Introducción

Antecedentes

Los sistemas inteligentes se han ido fortaleciendo conforme pasan los años para brindar soluciones tecnológicas y de esta forma solventar requerimientos muy complejos para la humanidad. Estos sistemas han permitido dar la pauta para la creación de sofisticados prototipos tales como detección, reconocimiento y renderización de imágenes, detección de movimiento de objetos, seguimiento de trayectorias de objetivos. Existen hoy en día diversas aplicaciones en distintos campos tales como la agricultura, a través del control de calidad de los productos por medio de puntos de interés en imágenes de pruebas de calidad, en el campo de la medicina para detectar anomalías y enfermedades catastróficas a través de visión asistida por computador (Avilés Pincay & Barcia, 2016), en el campo de la seguridad biométrica para detectar y reconocer personas, ya sea mediante sus huellas, voz e incluso el rostro para así dar acceso a zonas restringidas. Hoy en día, que para la época que vivimos de pandemia, permiten detectar la temperatura de los individuos y si estos llevan puesto o no mascarilla.

Esta tecnología inteligente en cuanto a seguridad biométrica se refiere, tiene mucha notoriedad en cuanto a disminuir considerablemente detecciones erróneas, esto se refiere para dar acceso a distintas zonas donde solo el personal autorizado puede ingresar (Nayak, 2019).

En cuanto a los sistemas de reconocimiento facial, existen diversos desarrollos de ámbito empresarial, tales como HIKVISION (cámaras de reconocimiento facial y detección de temperatura único y múltiple en distintos entornos), ZkTeco (Cámara de reconocimiento facial), Dahua, con su sistema de reconocimiento facial, todas estos sistemas se encuentran equipadas con sofisticadas implementaciones tanto de hardware como de software, dichos sistemas son altamente optimizados y robustos, pero sin embargo en algunos casos al ser equipos cerrados en su desarrollo son imposibles de

realizar mejoras y esto hace que no sean alcanzables para todos los usuarios y empresas que desean adquirirlos.

Dentro de los problemas a resolver, se investigó a través de algoritmos libres para desarrollo (OpenSource), permitan diseñar un software de automatización y control que permita el control de acceso al área de trabajo de las personas de una manera mucho más rápida, eficiente en tiempo al momento de detectar y reconocer el rostro buscando siempre disminuir costos y optimizar recursos para que el producto llegue a los usuarios de pequeñas, medianas y grandes empresas (Pérez León & Rojas Arévalo, 2019).

Como antecedentes respecto a la problemática ya expuesta anteriormente, se han desarrollado trabajos de investigación de toda índole, comenzando por el procesamiento de imágenes, hasta algoritmos sofisticados de aprendizaje profundo. Las investigaciones inician con un algoritmo eigenfaces y redes neuronales (Chasiquiza Molina, 2008), en el cual se toma una serie de imágenes de muestras del rostro, obteniendo las características del mismo y posteriormente obtener un modelo que permita reconocer al usuario comparando con el modelo previamente ajustado. En varios trabajos de investigación utilizan tecnologías de uso libre que permiten un desarrollo a bajo costo como “Python” (Lenguaje de programación) y “OPEN-CV” (Mordvintsev & Abid , 2013), en combinación con algoritmos de extracción de características conocidos como puntos de interés donde incluyen técnicas más avanzadas para clasificar, detectar y reconocer objetos mediante clasificadores HAAR, (Zhao & Ram, 2004) con modelos que ya han sido entrenados, los cuales pueden ubicar fácilmente el rostro, o características más específicas como la nariz, boca y ojos, este sistema prototipo expuesto previamente fue adaptado para el acceso a un laboratorio de la EPN, (Arias Melendres, 2020) aplicado en una tarjeta desarrollo, raspberry pi, pero con la necesidad de una gran cantidad de muestras para validar los datos al momento de reconocer el rostro.

Una mejora del sistema de reconocimiento facial antes mencionado, se lo ha realizado con algoritmos de aprendizaje automático, aplicado en el trabajo de investigación denominado, “Visión artificial aplicada a la detección e identificación de

personas en tiempo real" (Acuña Escobar, 2019), en este proyecto se realiza una comparación de los distintos métodos aplicados al reconocimiento facial e ingresando brevemente al sub-campo de la inteligencia artificial como lo es el aprendizaje profundo, siendo los algoritmos del aprendizaje automático con TensorFlow, los más efectivos aplicados al trabajo de investigación antes mencionado. En el siguiente trabajo de investigación llamado, "Sistema de control de acceso mediante identificación y verificación facial fundamentado en algoritmos de aprendizaje automático y redes neuronales", (Ibarra Flores, 2020) compone algunos de los conceptos de aprendizaje automático, desarrollado con dos modelos específicamente tales como KNN (Vecinos más cercanos) y SVM (Máquinas de soporte vectorial), el sistema fue implementado en un tarjeta de desarrollo de NVIDIA, la cual posee una GPU (Tarjeta de video) de baja escala, concluyendo que el modelo SVM, es el que mejor respuesta obtiene al momento de la detección y reconocimiento de rostros.

Todas las investigaciones analizadas previamente tienen una conclusión en común, necesitan una gran cantidad de muestras en cuanto a rostros se refiere, para que el sistema no tenga falsos positivos al momento de detectar e identificar los rostros, y esto se debe a que el hardware, ya sea tarjetas de control o adquisición de datos en muchos casos son de bajos recursos en cuanto a tecnología se refiere y en cuanto al software, requieren además un gran cantidad de datos para aprender, entrenar para así extraer la mayor cantidad de características posibles y en sí obtener un modelo con una precisión elevada al momento de inferir los algoritmo para la detección e identificación de personas.

Posteriormente nos encontramos con el campo del aprendizaje profundo con el desarrollo de varios algoritmos que en conjunto con las redes neuronales artificiales (ANN) y redes neuronales convolucionales (CNN), en este sub-campo del aprendizaje automático se tiene modelos desarrollados por el firmware TensorFlow, los cuales presentan una elevada precisión y velocidad de detección cercana al 85% en varios modelos robustos y del 98% en modelos menos precisos pero brindan una rápida detección de rostros (Abadi, et al., 2015) modelos como YOLOV4 (Bochkovskiy A., 2020)

(solamente lo observas una vez por sus siglas traducidas al español), con su gran detección de múltiples objetos en tiempo real y que aumenta sus FPS (fotogramas por segundo) de 10% al 12% con respecto a su antecesor YOLOV3 (Nayak, 2019). YOLOV4 ha contribuido con algoritmos de detección e identificación de rostros por medio de su versatilidad al momento de entrenar objetos personalizados, tales como InsightFace (Guo , et al., 2018) y RetinaFace (Zhou, et al., 2019), creado por el marco de desarrollado MXNET (Tianqi , et al., 2015), que detecta el rostro y realiza la identificación del rostro del usuario registrado.

En diversas aplicaciones para detectar y alinear el rostro utilizan el algoritmo de MTCNN (Redes neuronales convolucionales cascada en multitarea) (Zhang, Zhang, Li, & Qiao, 2016), con algunas de sus técnicas de aprendizaje profundo como lo es Faster R-CNN (Ren, He, Girshick, & Sun, 2016), esto con el fin de determinar variables tales como, distintas posturas del rostro, iluminación y oclusiones. Para revelar algunas características del rostro y puntos de referencia faciales, varias investigaciones usan DLIB (Rosebrock, 2017), el cual utiliza un predictor de aprendizaje automático para determinar, con 68 puntos donde se localiza cada una de las partes del rostro tales como nariz, ojos, cejas y por su puesto la boca, otro modelo de detección de puntos faciales es MediaPipe Face Mesh (MediaPipe, 2020) con 468 puntos característicos del rostro creando un modelo del rostro en 3D, este modelo es bastante usado en dispositivos móviles.

Para el reconocimiento facial, existen varios algoritmos de aprendizaje profundo como, Face Recognition (Geitgey, 2020), FaceNet (Schroff, Kalenichenko, & Philbin, 2016), OpenFace (Amos, Bartosz , & Satyanaray, 2016), todos con la función más importante, reconocer el rostro del usuario con una sola foto al momento de cargar y ajustar al sistema. Algunos algoritmos además de los componentes anteriormente mencionados de detección y reconocimiento del rostro cuentan con modelos de detección de la edad, estado de ánimo, uno de ellos es DeepFace (Serengil & Ozpinar, 2020), por medio de este algoritmo se puede determinar si el estado de ánimo del usuario,

determinando si el usuario está feliz, enojado, sorprendido e incluso triste. Este algoritmo ha sido usado en la plataforma de Facebook.

Una vez investigados y a partir de medidas de precisión se han evaluado el desempeño de estos tres métodos (detección del rostro, puntos faciales, reconocimiento facial), obteniendo una mayor eficiencia, para la detección y parámetros de ajuste con MTCNN, MXNET con su modelo RetinaFace, a su vez en la determinación de puntos faciales, para el presente trabajo de investigación se ajusta la librería DLIB, en cuanto al reconocimiento del rostro se tomará como base FaceNet. Todo esto se lo implementará con recursos computacionales alcanzables para el desarrollo (Chauca Vera, 2020), a través de un pc o tarjeta de desarrollo que cuenta con un GPU de gama media con la tecnología necesaria para la creación del sistema.

Finalmente, para las interfaces del software de ámbito profesional han sido desarrolladas en lenguaje de etiqueta HTML5 y estilos CSS3 dando las animaciones junto con funciones similares a las de JavaScript propias del firmware de Flask, dichas interfaces son amigables al usuario, e intuitivas para los objetivos del sistema a implementar (Web Dev Simplified, 2020), (Renotte, 2020), (Josemon, Irshad , Sunil, & Sasikumar, 2017). Por lo tanto, el sistema debe ser capaz de realizar las tareas antes mencionadas y disminuir la detección de falsos positivos al momento de reconocer el rostro, así como el estado de ánimo, este último punto como desarrollo para trabajos futuros ya que puede ayudar mucho en cuanto a estadísticas para controlar el nivel de estrés del personal de ciertas áreas a lo largo del tiempo (Arias Melendres, 2020), todo esto buscando una implementación a bajo costo.

Justificación e Importancia

En los últimos años múltiples aplicaciones de visión e inteligencia artificial con aprendizaje profundo han tomado mucho protagonismo en el campo del reconocimiento facial, muchas de estas han dado soluciones para grandes empresas para controlar su personal al momento de ingresar a sus labores cotidianas, pero que si bien han aportado significativamente a la solución del problema, todavía se utilizan recursos de hardware

que aumentan los precios de adquisición de estos sistemas siendo en muchos casos más específicos, pequeñas y medianas empresas por costo muy difícil de implementar. Por lo tanto, la visión por computador en conjunto con el campo del aprendizaje profundo de crecimiento rápido, aumenta el desafío de lograr soluciones sostenibles y no tan costosas para diseñar un sistema que permita atacar el problema de un control adecuado de acceso del personal (Acuña Escobar, 2019) (Ibarra Flores, 2020).

No obstante, para el problema se encuentran desarrolladas varios trabajos de investigación con procesamiento de imágenes, filtrando minuciosamente rostros de un set grande de imágenes en distintos entornos, se tiene además soluciones empresariales de gran rendimiento, pero debido a costos muchos de estos equipos quedan fuera del alcance debido a que requieren en ciertos casos muchos datos para validar los rostros de muchos usuarios y en otros casos la adquisición que esto significa.

El trabajo de tesis propuesto desarrolla una solución fuertemente vinculada tanto al hardware como al software para implementar un sistema automático de control de acceso biométrico por medio del reconocimiento facial, el cual tiene como ideas principales ser rápido, eficiente y sobre todo preciso al momento de identificar al usuario o usuarios autorizados, conjuntamente se incluye la función de seguridad en cuanto a vulnerabilidades se refiere, en donde en muchos casos con un video o foto del usuario registrado en el sistema, se de libre acceso a esta zona y de esta forma cometer delitos, es por esta motivo que se incluyó un algoritmo de anti-plagio que permita al sistema evitar estas acciones.

Se realizará el diseño de una interfaz para la automatización y control de acceso del usuario el cual mediante la base de datos de rostros y Caras etiquetadas en la naturaleza (por sus siglas en inglés, "Labeled Faces in the Wild (LFW (Zhang, Zhang, Li, & Qiao, 2016) (Geitgey, 2020), ajustadas al sistema previamente, pueda identificarlo en el momento de entrar o salir de su área de trabajo, adicional a esto se crean registros

automáticos de la hora de ingreso y salida de cada usuario. Cabe mencionar que todo este proceso se realiza en tiempo real.

Alcance del Proyecto

El presente proyecto de investigación tiene como alcance el diseño de un sistema de reconocimiento facial aplicando técnicas de aprendizaje profundo para la automatización y control del acceso del personal, de un conjunto de características; generadas de un conjunto de rostros. El hardware en el cual se implementará será un equipo que posea una tarjeta gráfica de gama media, a través de una interfaz de usuario que será desarrollada en lenguaje de etiqueta, además permitirá la comunicación con los algoritmos de inteligencia computacional. El sistema será en tiempo real y de respuesta de reconocimiento a 3 segundos al momento de realizar el reconocimiento facial, todo esto se logrará con un par de rostros de cada usuario para ajustar el modelo y así realizar la detección e identificación del usuario, siendo este último punto un aspecto fundamental para el sistema ya que se puede agregar muchos usuarios y tener una precisión elevada al momento de identificarlos y con esto evitar confundirse entre cada usuario que tenga alguna similitud entre sí. Exponiendo brevemente, se iniciará con el algoritmo MTCNN y RetinaFace en la plataforma de PyTorch, basado en aprendizaje profundo, a través de cual nos permite detectar y alinear el rostro, con esto facilitará obtener las características del mismo y así como también guardar los rostros de los usuarios para ajustar el modelo al momento de realizar el reconocimiento del usuario. Luego de realizar este paso se extraerá los puntos faciales del rostro por medio del algoritmo FaceMesh de MediaPipe, que nos permitirá identificar las partes principales del rostro como nariz, boca, ojos. Posteriormente se identificará el rostro por medio del algoritmo FaceNet, ya que al ser un modelo pre-entrenado con un set aproximado de 3.3 millones de rostros en distintos entornos y condiciones iluminación, postura, etc, tiene una efectividad de reconocimiento del 99.63% (Geitgey, 2020).

A esto se suma un algoritmo de anti-plagio o de la vida, para darle una seguridad y robustez al sistema, ya que se ha logrado detectar en muchos casos que existen

vulnerabilidades tales como al colocar una foto o video de un usuario registrado, concediéndole acceso, y es por este motivo que se incluye esta función. Todo esto unido con el pequeño marco de desarrollo de uso libre denominado Flask que nos permite crear aplicaciones web, el cual ya viene un servidor web incluido, además es compatible con Python.

En último lugar se diseña la interfaz web para el sistema de control de acceso biométrico en el lenguaje de etiqueta HTML5 con los estilos CSS3, OpenCV y animaciones con funciones propias de Flask que tienen mucha similitud a JavaScript en cuanto a comunicación con el lenguaje de Python y otros aplicativos se refiere.

En la interfaz consta la información del monitoreo de los usuarios en donde se mostrará las personas autorizadas, registradas y en caso de no ser un usuario registrado publicar una asignación de desconocido. El sistema tendrá un menú desplegable que permitirá ajustar una cámara IP. Adicional se crea una ventana en la interfaz con lenguaje de etiqueta donde permite crear, quitar y ajustar los parámetros del modelo para nuevos usuarios, este programa será paralelo al sistema de identificación facial actualizando los pesos de la red neuronal de forma automática sin detener el sistema, posteriormente se creara información donde consta la hora, día, mes y año, así como el momento de ingreso y salida del usuario, estos datos serán exportadas en un archivo de Excel automáticamente por el sistema, sin utilizar una base de datos pero creando una gran cantidad de información en el cual observará parámetros como los expuestos previamente que servirán para sistemas de contabilidad los cuales determinarán las horas trabajadas por cada uno de los usuarios.

En la etapa final del proyecto se dará lugar a la realización de pruebas de funcionamiento que busca analizar la precisión y velocidad de respuesta del sistema, además de realizar un estudio determinando que al utilizar el sistema desarrollado, el usuario puede elegir el umbral de decisión y encontrar un equilibrio entre FAR (tasa de aceptación falsa) y TAR (tasa de aceptación verdadera).

Objetivos

Objetivo General

Desarrollar e implementar un sistema de control de acceso de personal por medio de reconocimiento facial en el cual se considere algoritmos de aprendizaje profundo funcionando en tiempo real.

Objetivos Específicos

- Realizar un estudio del estado del arte sobre algoritmos de aprendizaje profundo para la detección, reconocimiento facial, características únicas al momento de detectar el rostro y rasgos faciales del rostro.
- Diseñar una interfaz de usuario intuitiva y de fácil manejo enfocada en controlar el acceso de los usuarios vía reconocimiento facial.
- Diseñar un sistema biométrico seguro y robusto que disminuya la probabilidad de detecciones erróneas y además se observa en la interfaz de usuario como respuesta, si el usuario está registrado o es desconocido.

Capítulo II

Fundamentación Teórica

Introducción a los sistemas biométricos

Para iniciar con el estudio de los sistemas biométricos, se parte del concepto de biometría, el cual menciona que es el estudio de métodos automáticos para el reconocimiento único de humanos basados en uno o más rasgos físicos propios (Cedeño Navarrete & Párraga Vera, 2017). Teniendo claro este concepto se tiene que los sistemas informáticos son la aplicación de métodos matemáticos y estadísticos sobre el rasgo físico o de conducta de una persona los cuales permiten verificar su identidad o para identificar a las personas.

Las huellas dactilares, el rostro y hasta la forma como se mueve al caminar un individuo, son todas características únicas de cada persona, hoy en día estos datos se utilizan muchísimo para aplicaciones donde se requiera cancelar algún servicio, o para registrarse en alguna plataforma web con el rostro.

Funcionamiento y rendimiento

Los sistemas biométricos se basan en verificar científica y digitalmente en parámetros característicos vivientes únicos e irrepetibles para cada persona, de tal manera que estos se constituyen como únicos para identificar correcta y positivamente a una persona, evitando los métodos tradicionales tales como firmas, contraseñas, pines y códigos vulnerables a ser copiados y manipulados al ser sustraídos, falsificados o descifrados con fines delictivos (Quevedo Gonzalez, 2017).

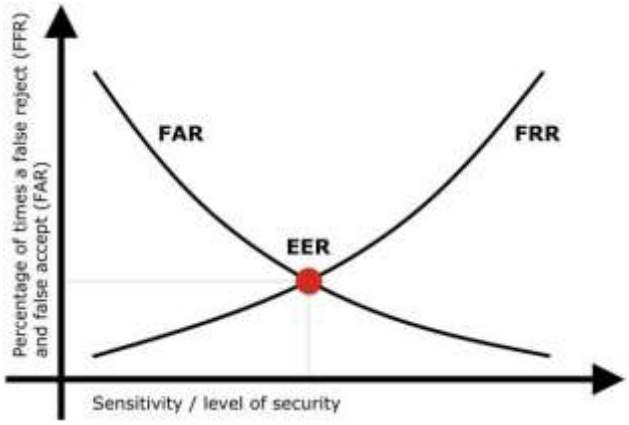
En teoría, cuando ingresa la información del individuo registrado al sistema biométrico, en su totalidad las características del mismo, concuerdan, entonces cuando un individuo no está registrado en la base de datos del sistema biométrico, está no empareja con los datos y no le concede el acceso.

Las tecnologías modernas han ido mejorando conforme los avances tecnológicos de hardware y software han ido evolucionando, obteniendo tasas de error en promedio desde los 60% hasta los 99.9%.

El rendimiento de una medida biométrica está sujeto a pruebas llamadas falso positivo (por sus siglas en inglés False Acceptance Rate, FAR), junto con la tasa de falso negativo (por sus siglas en inglés False Rejection Rate, FRR) (Yuahn, 2017).

Figura 1

Rendimiento de los sistemas biométricos.



Nota: La figura representa los índices para medir la efectividad de un sistema biométrico y verificación. Tomado del sitio web: *FAR and FRR: security level versus user convenience,* (recogtech, 2021).

En los sistemas biométricos implementados en las empresas el FAR y FRR tal como se observa en la Figura 1, son parámetros que se transforman modificando ciertas características, existen diversas medidas adicionales a considerar como la tasa en la cual se acepta o rechaza errores los cuales son iguales, la tasa de error igual (por sus siglas en inglés, Equal Error Rate, EER) y la tasa de error de cruce (por sus siglas en inglés, Cross-over Error Rate, CER), cuanto más bajo son estas dos medidas, se considera el sistema biométrico más exacto (Alvarado Zambrano, Landeta Rodríguez, Sánchez Jiménez, & Castro Arreaga , 2010).

Comparativa de los sistemas biométricos en la industria

A continuación se muestra una Tabla 1, comparativa de los diversos sistemas biométricos que se encuentran en la industria:

Tabla 1

Comparativa de los sistema biométricos actuales

Comparativa de sistemas biométricos						
Parámetros	**Iris**	**Geometría mano**	**Huellas dactilares**	**Escritura y firma**	**Voz**	**Rostro**
Fiabilidad	Muy alta	Alta	Muy alta	Media	Alta	Muy Alta
Fácil uso	Media	Alta	Alta	Alta	Alta	Muy Alta
Estabilidad	Alta	Media	Alta	Baja	Media	Alta
Aceptación	Media	Alta	Alta	Muy alta	Media	Alta
Prevención de ataques	Alta	Alta	Alta	Media	Baja	Alta

Nota: Esta tabla muestra el grado de aceptación que tiene en la industria los sistemas biométricos. Tomado de *Análisis diseño e implementación de un sistema de inmótica para el edificio administrativo de la facultad técnica para el desarrollo de la universidad católica de Santiago de Guayaquil:* (Alvarado Zambrano, Landeta Rodríguez, Sánchez Jiménez, & Castro Arreaga , 2010).

Ventajas de los sistemas biométricos

La identificación con datos biométricos avanza muy rápidamente ya que brinda facilidad y practicidad en la vida diaria ya que los métodos engorrosos de escribir contraseñas, números alfanuméricos es muy tedioso, y en muchos casos los individuos tienden a olvidar, colocar una huella dactilar, un escaneo del iris, en un sensor de un dispositivo es más fácil que escribir una contraseña, además muchas de estas son objetos de ataques además que es un método anticuado pero aún son fáciles de implementar y por eso son tan extendidos.

En cuanto a los sistemas de reconocimiento facial se refiere, es una cuestión de costos se utilice con mayor precisión se puede capturar un rostro, si existen los suficientes sensores así como las técnicas y métodos adecuados para obtener las características faciales resulta más seguro que una contraseña. El futuro está en la autenticación multifactorial de al menos dos características, y que por razones de utilidad prevalecerá los procedimientos que no causen tanta molestia al usuario.

Seguridad y fiabilidad de los sistemas biométricos

Los proveedores que utilizan los sistemas biométricos deben brindar la seguridad de que los datos se almacenen de forma segura y encriptada directamente en el dispositivo no como se lo ha venido realizando en grandes servidores en una nube, los cuales son vulnerables ataques cibernéticos (Quevedo Gonzalez, 2017).

Los sistemas biométricos tiene mejoras en cuanto a la seguridad se refiere, pero se tiene aún problemas, en Israel investigaciones recientes se determinó que lograron piratear una base de datos de 23 GB con cerca de 27 millones de datos, huellas dactilares y registros faciales (BBC Mundo Tecnología, 2017). Es ahí que surge el problema, ya que el elemento biométrico no puede borrarse o modificarse.

En pruebas de laboratorio los denominados hackers ya han descifrado muchos métodos de encriptación biométrica, el escáner dactilar del iPhone por ejemplo puede ser pirateado con una huella dactilar de un vaso de cristal, así mismo el escáner de iris de Samsung se puede vulnerar con una foto del ojo del usuario y un lente de contacto, en sistemas biométricos más avanzados, lograron piratear con una mano de cera un escáner de venas humana, así también hacker de origen chino, con simples pegatinas en unos lentes han logrado sortear sistemas de autenticación facial (Lopez, 2018). Es importante mencionar que son escenarios especiales de prueba, la calidad de los sensores tiene un papel decisivo en contrarrestar estos aspectos, Un Smartphone es más fácil de piratear que un acceso a un área de seguridad.

Por lo tanto las contraseñas biométricas no son 100% infalibles, aun de esta manera siguen siendo una forma mucho más segura las actividades digitales de los individuos. A continuación se muestra una tabla de factibilidad de los sistemas biométricos.

Sistemas biométricos en la industria

Los sistemas biométricos, tienen potencial para identificar a las personas con alta certeza, la industria ofrece distintas tecnologías cada una con un diferente segmento de mercado, se muestra en la Tabla 2 los distintos aplicativos existentes en cuanto sistemas biométricos se refiere:

Tabla 2

Diversos sistemas biométricos en la industria

Tecnología	Aplicación	Mercados aplicados
HikVision-ZkTeco	Reconocimiento facial, identificación de personas	De uso civil, militar y empresarial
Huella Dactilares	Registros civil	Gobiernos, regiones y bancos
Identificación Facial	Acceso a instalaciones Identificación de individuos	Instituciones financieras, dispositivos móviles
Reconocimiento de voz	Acceso a instalaciones Acceso a sistemas	Viajes, búsqueda de información
Escritura y firma	Validación de datos	Instituciones gubernamentales

Nota: Esta tabla muestra, las diversas tecnologías de sistemas biométricos que existen en la industria además el área comercial de aplicación.

En la actualidad la industria biométrica, han ido perfeccionando este campo, siendo muchas las empresas que desarrollan sistemas biométricos muy sofisticados de uso profesional, donde en ciudades como Londres y Beijing cuentan con redes de sistemas biométricos usados para la seguridad ciudadana.

Inteligencia artificial

La inteligencia artificial es una combinación de algoritmos propuestos para desarrollar una variedad de sistemas, máquinas y robots que pueden parecerse a los humanos, siendo de las más fuertes en cuanto a crecimiento y expectativa salarial para los profesionales que se tiene hoy en día (Giraldo de la Caridad & Silvia Margarita , 2017).

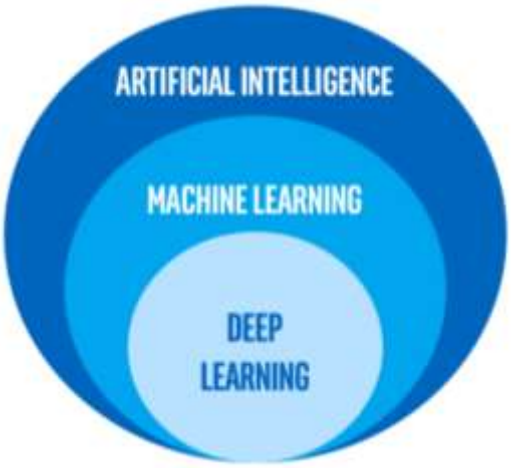
Según opiniones y conceptos anteriores, existen diferentes tipos de IA, estos se detallan a continuación:

- Robótica: Aplicado a tareas y habilidades que se le puede programar a un robot usando el IA mediante la movilidad del mismo
- Speech: Cuando se necesita reconocer la voz, obteniendo sonidos, decodificando estos datos y así extraer palabras.
- NLP: Aplicando técnicas de IA para entender, interpretar y procesar un lenguaje
- Ciencia de datos: Sub-campo de la IA, que permite obtener modelos y predicciones estadísticas para extraer significado y conocimiento de los datos.
- Visión por computadora: Otorgando la habilidad a un sistema con IA de observar y a través de este medio obtener imágenes del mundo real.
- RPA: Automatización robótica de procesos, este concepto permite imitar tareas digitales repetitivas de los seres humanos.

Se puede además observa en Figura 2, el vínculo que conforman todas las disciplinas de la IA, unidas en conjunto para dar soluciones a diversas aplicaciones sofisticadas, en breve se dará más detalles de estos sub-campos de aplicaciones así como una explicación de la visión por computadora, rama de mucha importancia para el desarrollo del presente trabajo de investigación.

Figura 2

Vínculo entre la IA, machine learning y deep learning



Nota: El gráfico representa los tipos de aprendizaje que enmarca la inteligencia artificial. Tomado de *The Difference Between Artificial Intelligence, Machine Learning and Deep Learning*, (Robins , 2020)

Aprendizaje automático

Es una tecnología que permite realizar tareas automáticas con algunas operaciones con el objetivo de evitar en lo posible la necesidad de que intervengan los seres humanos. Esto da como resultado una factibilidad a la hora de analizar una gran cantidad de información de una manera más adecuada y efectiva.

Figura 3

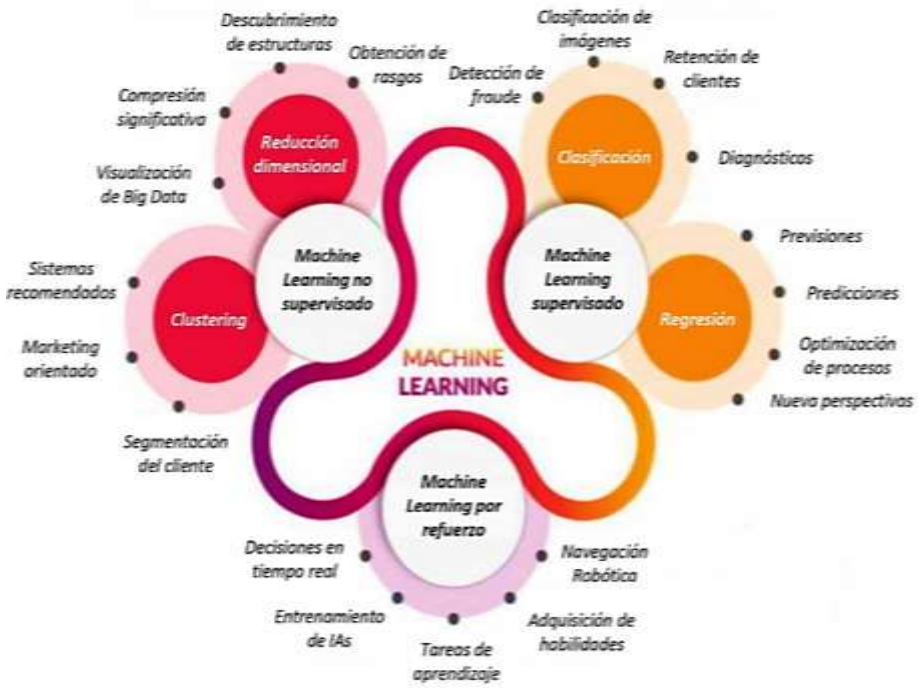
Esquema general del aprendizaje automático

Nota: La figura muestra cómo se comporta un sistema general de aprendizaje automático

La relaciones y características obtenidas proporcionan un modelo pre-entrenado, como se muestra en la Figura 3 y estas reglas son las utilizadas para extraer nuevas predicciones en los posteriores datos ingresados al sistema.

Figura 4

Aplicaciones del aprendizaje automático



Nota: El gráfico representa los distintos paradigmas del aprendizaje automático y su clasificación. Tomado del sitio web interempresas.net | *Los conceptos de Machine Learning y Deep Learning en la industria,* (interempresas, 2021)

Paradigmas del aprendizaje automático

Se clasifica en varios tipos, tal como se muestra en la Figura 4 uno de los más aplicados, es según el tipo de estrategia y ayudas que recibe el sistema de IA sobre una información disponible:

- *Aprendizaje Supervisado*: Instrucción que se delega en encontrar las interrelaciones o patrones entre una variable de ingreso y una salida. Predice el

futuro a partir de datos históricos. Es aplicado para resolver inconvenientes de categorización y regresión.

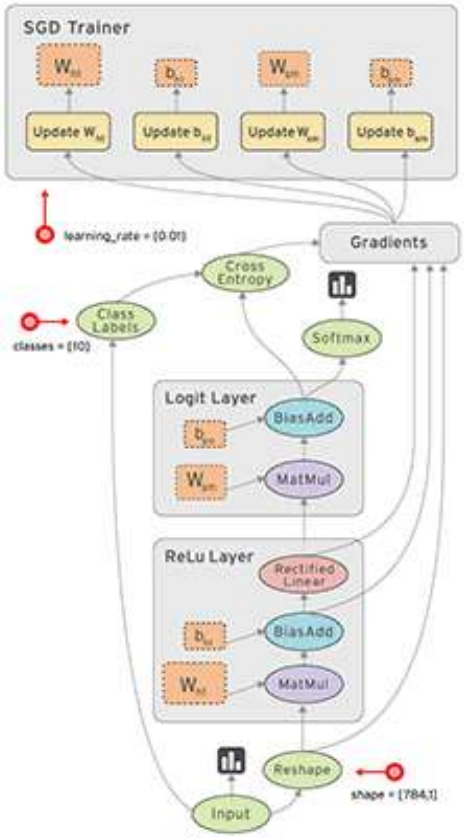
- *Aprendizaje No supervisado*: Son diseñados para desarrollar nuevos conocimientos por medio del hallazgo de regularidades en los datos. Aquí se encuentra el problema de clusterización, donde el algoritmo descubre patrones de semejanza entre los datos.
- *Aprendizaje Reforzado*: Los sistemas aprenden a partir de la experiencia, basado en la prueba y error dando pauta a reglas de premio para crear un sistema más óptimo o castigo en caso contrario. Es un aprendizaje interesante en los sistemas de IA, ya que no requiere el ingreso de un set enorme de información.

Framework para el aprendizaje automático -Tensorflow

Desarrollado y optimizado por Google (Abadi, et al., 2015), es una plataforma de código abierto en su mayoría de algoritmos y librerías para el desarrollo de aplicativos tanto de aprendizaje automático como aprendizaje profundo. Basado en la manipulación, almacenamiento y transferencia de tensores (generalización del concepto de matrices), tal como se muestra en Figura 5.

Figura 5

Estructura de TensorFlow



Nota: En la figura se muestra la forma en como realiza TensorFlow el procesamiento matemático de una Red neuronal Artificial. Tomado de *Learn TensorFlow 4: Convolutional Neural Network (CNN's),* (TensorFlow Org., 2015)

Es muy usado en el desarrollo de modelos de aprendizaje automático, inferidos en GPU's para sus modelos más robustos ya sea de forma local o en la nube. Su vasta gama de librerías y modelos pre-entrenados permiten aplicarlos en diversas plataformas así como también en dispositivos móviles.

Tensorflow completa en su marco de desarrollo una API de aprendizaje profundo de alto nivel para crear, compilar y entrenar modelos, conocida como *Keras*, que por medio de una interfaz amigable, modular y configurable, proporciona al usuario la facilidad de una rápida creación de prototipos (Warden & Situnayake, 2020) (Chauca Vera, 2020).

Al utilizar esta API unido a Python como lenguaje principal, permitirá definir y entrenar redes neuronales. Posee también una integración con distintos lenguajes asi como el edge computing para dispositivos móviles, el IOT, servidores periféricos locales e integrados a través de plataforma conocida como *Tensorflow Lite* (TensorFlow Org., 2015).

Neurona artificial

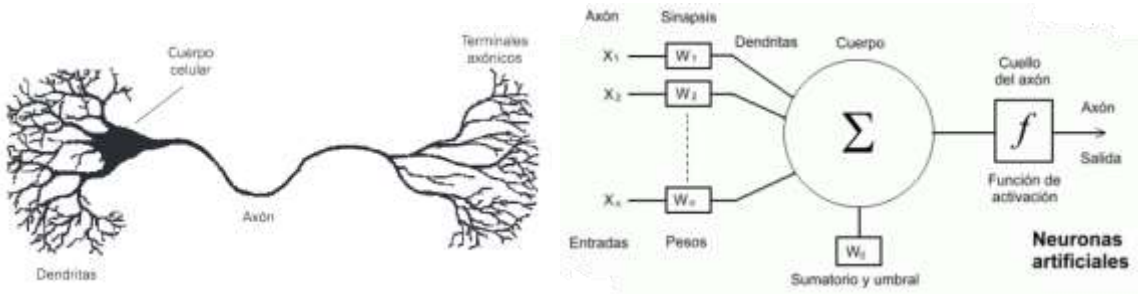
Uno de los grandes desarrollos de la IA son las redes neuronales artificiales, antes de determinar este concepto, se explica que es una neurona artificial. Una neurona artificial, es una unidad que intenta copiar el comportamiento de una neurona biológica que posee el cerebro humano, en ella se intenta crear interconexiones para crear conocimiento por medio de una red neuronal.

Estructura de una neurona artificial

Existen diversas investigaciones en cuanto a neuronas artificiales se refiere, donde una de la primeras fue de McCullont-Pitts en 1943, que asemeja al concepto anteriormente explicado, tal como se muestra en la Figura 6, luego con fines de entendimiento se analiza el perceptrón de Frank Rosebank de 1958 (Haykin, 1994). El manifiesta que la neurona artificial se compone de una sumatoria matemática y cada una de sus entradas son representadas como entradas de datos multiplicadas por un algún valor conocido como, peso sináptico, el cual le asigna una prioridad a cada una de las entradas para posteriormente ser procesadas en la sumatoria, una vez realizado este proceso son sometidas a una función de activación y que por medio de esta proporciona la activación de la neurona a través de su salida. Este concepto tuvo aplicaciones de separabilidad lineal con una sola neurona.

Figura 6

Izquierda: Red neuronal biológica; Derecha: Representación del modelo matemático de una neurona.



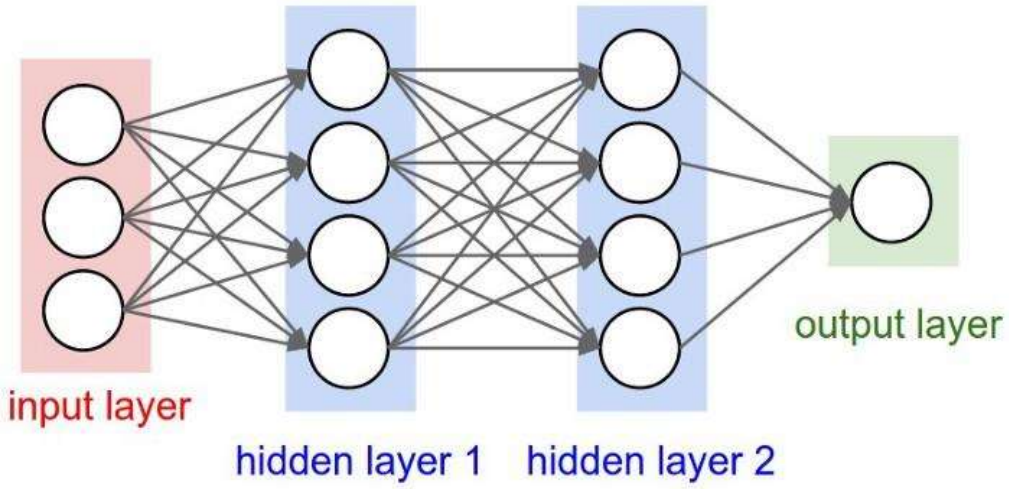
Nota: La imagen muestra la representación de una neurona biológica y una neurona artificial representada matemáticamente. Tomada de *Redes Neuronales: una visión superficial*, (Caparrini, 2019)

Redes neuronales artificiales (ANN)

Una red neuronal artificial como se muestra en la Figura 7, es un modelo infundido en el funcionamiento del cerebro humano, se conforma por un grupo de nodos conocidos como neuronas artificiales que permanecen conectadas y transmiten señales entre sí. Estas señales se transmiten a partir de la entrada, tomando propiedades del sistema el cual se encuentre desarrollando por medio del aprendizaje, donde se va a tener una funcionalidad que predice los resultados hasta crear una salida. La unión de neuronas forman capas neuronales todas ellas conectadas, donde su valor de salida ingresa a las siguientes capas de la red, hasta llegar a la salida, donde mostrará la predicción para cada entrada dada (Nielsen, 2015). Todas las neuronas en una capa trabajan sin dependencia y no comparten información alguna.

Figura 7

Red neuronal conectada



Nota: En el gráfico se observa la estructura de una red neuronal conectada completamente. Tomada de *Redes neuronales,* (Freire & Silva, 2019)

Hiperparametros.

Dentro de una neurona y redes neuronales en general se tiene algunos parámetros que componen para que el modelo tenga un óptimo desempeño a continuación se detalla estos puntos (Rodríguez, 2018) :

- **Sobreajuste (Overfitting):** Este concepto se toma cuando la red neuronal aprende de los datos de entrenamiento pero tiene bajo desempeño sobre los datos de validación o en datos que no ha visto jamás. Se debe evitar a toda costa este parámetro.
- **Bajo ajuste (Underfitting):** Aparece cuando la red no puede identificar u obtener resultados correctos ya que los patrones de las muestras no son del todo correctas, sus resultados por lo general son malos.
- **Datos de entrenamiento:** Llamadas también muestras, son los datos que se le entrega a red neuronal para que pueda extraer las características necesarias y por consiguiente poder aprender. Por lo general este set de datos es muy grande,

y se divide en dos partes donde el 80 a 90% de los datos son para el entrenamiento y el restante es para validación.

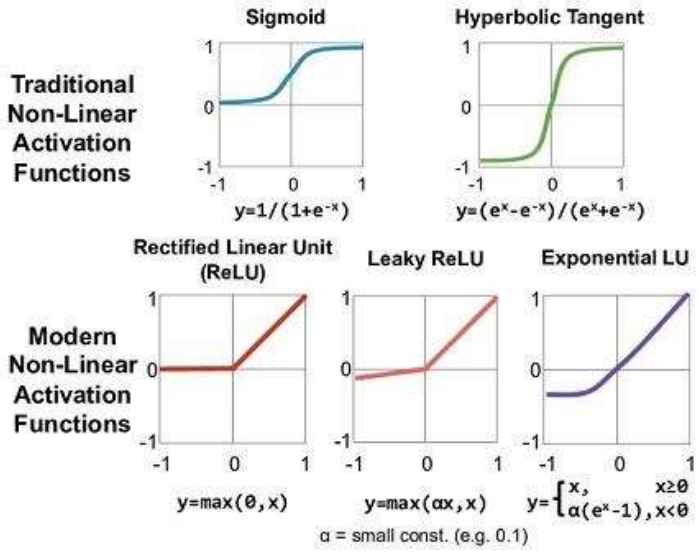
- **Datos de validación:** Son datos distintos con los cuales se valida que tan buen desempeño presenta la red neuronal.
- **Retropropagación (Back-Propagation):** Es un algoritmo denominado por muchos autores el cual permite la optimización de la función de pérdida mediante la actualización de los pesos y el sesgo para que pueda mejorar las predicciones de la red neuronal**.**
- **Propagación adelante (Forward-Propagation):** Es la forma como las redes neuronales crean las predicciones.
- **Optimizador:** Es la función que permite a la red tener un buen desempeño, existen diversos algoritmos de optimización como el gradiente descendiente o la función de Adam.
- **Taza de aprendizaje:** (Learning Rate, por sus siglas en inglés,), es el parámetro que proporciona a la red que tan extenso será su entrenamiento. Si este valor es muy pequeño la actualización de los pesos no cambiarán correctamente y además puede tardar mucho más tiempo en aprender, en cambio si la tasa de aprendizaje es muy alta, la actualización de los pesos puede llegar a superar el punto ideal del mínimo global y jamás en encontrarlo, si bien es más rápido el aprendizaje no tendrá un buen desempeño la red neuronal.
- **Función de pérdida:** Permite conocer el desempeño de la red neuronal, si se tiene un resultado alto quiere decir que la red es baja en cuanto a su trabajo y viceversa. Unido al back-propagation, sirve para optimizar o minimizar el algoritmo, dependiendo el problema se puede usar las funciones matemáticas como entropía cruzada o error mínimo cuadrático
- **Época:** Es el número de iteraciones que se ejecutan el algoritmo tanto de back-propagation y forward propagation. En cada ciclo todos los datos pasan por la red

neuronal para que estudie y extraiga las características actualizando de manera consecutiva los pesos.

- **Tamaño de lote: (**Batch Size. traducido del inglés), Son los datos que ingresan a la red en cada iteración de un ciclo, este comportamiento permite actualizar los pesos muchas veces, además cuando se tiene set de datos enormes, se necesitan muchos recursos computacionales, donde si este parámetro varía en lotes más pequeños permite que la red neuronal aprende más rápido y su entrenamiento no tome tanto tiempo.
- **Funciones de activación:** Son funciones matemáticas las cuales muestran una respuesta a la salida conforme a la suma ponderada ingresada en la entrada, es decir normalizan la los valores para activar o desactivar la salida en función de las entradas. Estas funciones de activaciones pueden ser de dos clases, funciones lineales y funciones no lineales tal como se muestra en Figura 8.

Figura 8

Representación gráfica y matemática de las funciones de activación.



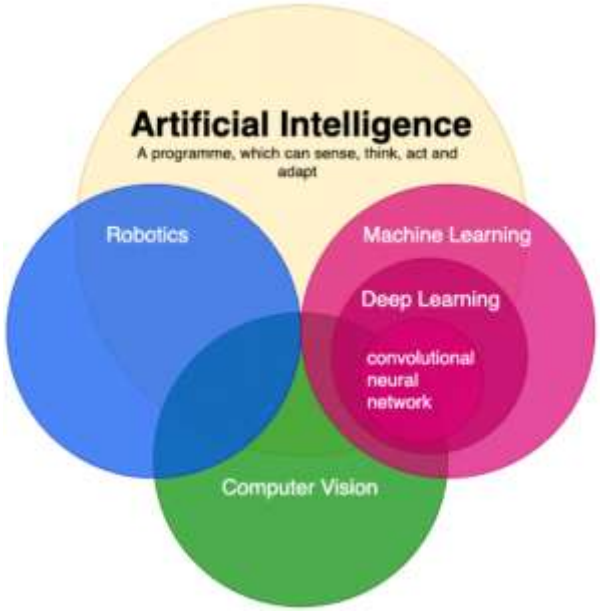
Nota: El gráfico muestra las funciones de activación usadas en las RN. Tomado de *Catálogo de componentes de redes neuronales (II): funciones de activación,* (Gavilan, 2020)

Visión por computadora

Uno de los aplicativos antes de continuar profundizando en la IA, es la visión por computadora, este concepto como tal se muestra en la Figura 9, su aplicación se encuentra entre el aprendizaje automático y profundo.

Figura 9

Visión artificial en el campo de la Inteligencia artificial



Nota: El gráfico muestra en qué área de la inteligencia artificial y otros aplicativos se encuentran los aplicativos de la visión artificial. Tomado de *Tele Stroke System for Stroke Detection*, (Rishabh, 2020)

Existe confusión en el término visión artificial y visión por computadora, en sí el concepto de la visión artificial se menciona que es la tecnología que se utiliza para detectar errores en una línea de producción o productos que deben categorizarse. Se utiliza principalmente en procesos industriales (Sucar & Gómez, 2014) en cambio la visión por computadora, también conocida, visión de máquina e imágenes es una rama de la IA, vinculada a procesos, técnicas tanto de software como de hardware que permiten extraer imágenes del mundo real por medio de un computador. Estas características se

encuentran desde métodos por procesamiento de imágenes hasta varios modelos pre-entrenados tanto de aprendizaje automático como de aprendizaje profundo.

Si bien la visión por computadora se puede usar sola sin ser parte de un sistema grande, la visión por computadora es parte de un sistema (Sucar & Gómez, 2014).

Sistemas de visión por computador

Basado en el punto anterior, se resume este concepto de sistema de visión por computador en mencionar que es un sistema independiente y autónomo, que realiza alguna de las tareas del sistema de visión humano (Alegre, Pajares, & de la Escalera, 2016). Este sistema de visión por computadora, es capaz de extraer características tridimensionales a partir de un set de imágenes o un modelo pre-entrenado.

Este concepto de visión artificial, en si como tal no es un tarea sencilla de emular para un computador, y esto es debido a que a diferencia del ojo humano el cual recibe las imágenes de los objetos en 3 dimensiones (3D), una máquina, o pc, recibe en dos dimensiones (2D). Esta simplificación de una dimensión reduce de gran manera la cantidad de información, creando errores en el sistema inteligente y aunque existen a día de hoy métodos y técnicas los cuales permiten robustecer el sistema, (uno de ellos es la visión estéreo), sigue siendo este punto una difícil tarea para el sistema de visión por computador a neutralizar (Fernández Castilla, 2013).

Representación de imágenes digitales

La información de una imagen se puede representar de diversas formas, estas formas son:

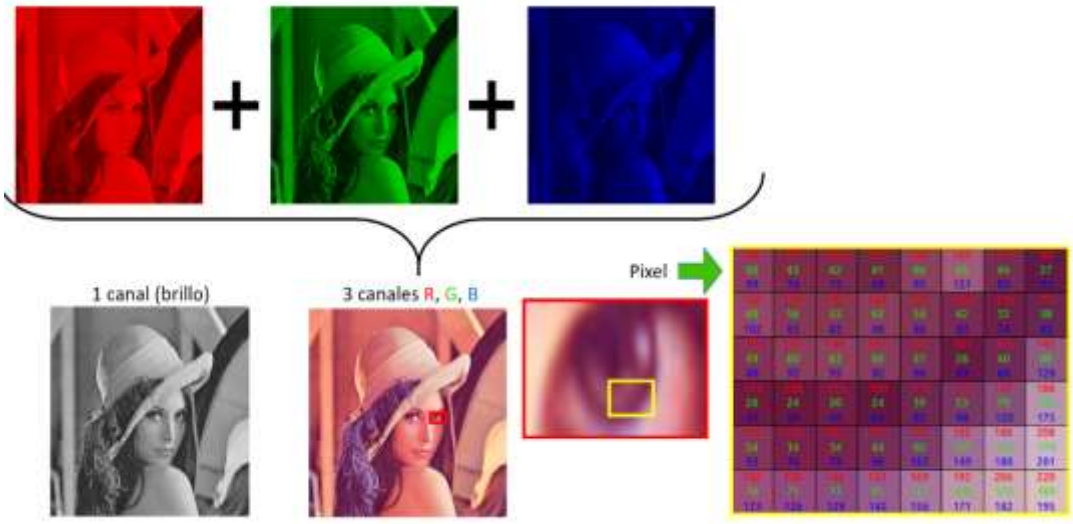
- *Imágenes de intensidad*: Surgen a través de la noción de luminosidad. Estas imágenes en cada uno de sus píxeles son representados por la intensidad luminosa o brillo en los cuales ha detectado el sensor de luz en el transcurso de la toma de muestras de estas imágenes.

- *Imágenes térmicas*: Se encargan de adquirir muestras por medio de sensores infrarrojos en los cuales se puede conseguir la temperatura que posee cada cuerpo de la escena obtenida.
- *Imágenes de rango*: Son conocidas como imágenes de profundidad o alcance, y es una forma muy común para determinar las coordenadas en tres dimensiones de uno o varios objetos en escena.

Luego de esta breve explicación, independientemente de la forma en la que esté representada una imagen, regularmente se lo realiza mediante una matriz numérica bidimensional MxN, siendo esta una imagen digital la cual el computador puede comprender y procesar tal como se muestra en la Figura 10.

Figura 10

Digitalización de una imagen



Nota: El gráfico muestra la obtención de las características para crear una matriz que ingresa al controlador o pc, como inicio para alguna aplicación de visión artificial. Tomado de *Procesamiento digital de imágenes,* (Morales Caporal, 2020).

Etapas de un proceso de visión artificial

Los pasos en un proceso de visión artificial generalmente se dividen en dos grupos. En un primer grupo están los pasos que son métodos de bajo nivel, y en un

segundo son los que realizan análisis de nivel de escena de procesamiento de imágenes de alto nivel. El objetivo de los pasos de nivel bajo es obtener lo más básico de la imagen, como bordes, regiones y otros atributos simples. En el caso de un procesamiento de alto nivel, los extractos se recopilan en el nivel inferior y se construye una descripción de la escena.

Para fines del proyecto se ha determinado seguir estos pasos del caso de alto nivel a continuación se detalla las etapas involucradas tal como se muestra en la Figura 11.

Figura 11

Etapas de análisis de un sistema de visión por computadora.

Nota: La imagen muestra cómo está compuesta las etapas de un sistema de visión por computadora. Tomado de *Etapas de un sistema de visión - Visión artificial,* (Solución Ingenieril, 2021)

- *Escenario analizar*: Es el área que se desea capturar, donde se encuentra la información que busca analizar y procesar.
- *Adquisición y digitalización de la imagen*: En esta etapa se captura una proyección en dos dimensiones de la luz reflejada por los objetos de la escena, pasando a un formato digital comprensible para la pc.
- *Preprocesamiento:* Se realizan tareas de eliminación de ruido y/o realce de la imagen.
- *Segmentación*: detección de bordes y regiones. Permite separar los diferentes elementos de la escena.

- *Extracción de características:* Se obtiene una representación formal de los elementos segmentados en la etapa anterior.
- *Reconocimiento y localización*: Mediante técnicas, como la triangulación, se localiza al objeto en el espacio 3D.
- *Interpretación*: A partir de la información obtenida en las etapas previas y del conocimiento acerca del entorno se interpreta la escena y se de una aplicación de acuerdo a la información analizada.
- *Aplicación*: Es el objeto final que se controlará de acuerdo a la información que se procesó.

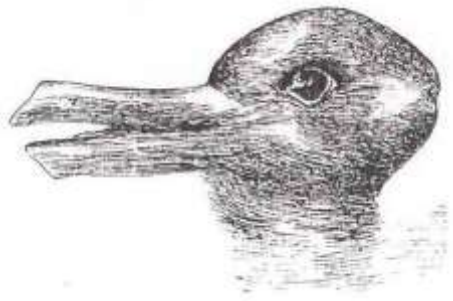
Debilidades de la visión por computadora

Un sistema de visión por computadora cuenta con algunas debilidades que se detallan a continuación:

- Ambigüedad en la definición de un concepto: Para entender mejor este punto se expone el siguiente ejemplo, se tiene un set de datos de sillas, y por distintos métodos, técnicas el sistema de visión por computadora se logra determinar que es una silla, pero qué ocurre si existe un objeto similar confundiendo al sistema de visión tal como se muestra Figura 12.

Figura 12

Imagen Ambigua.

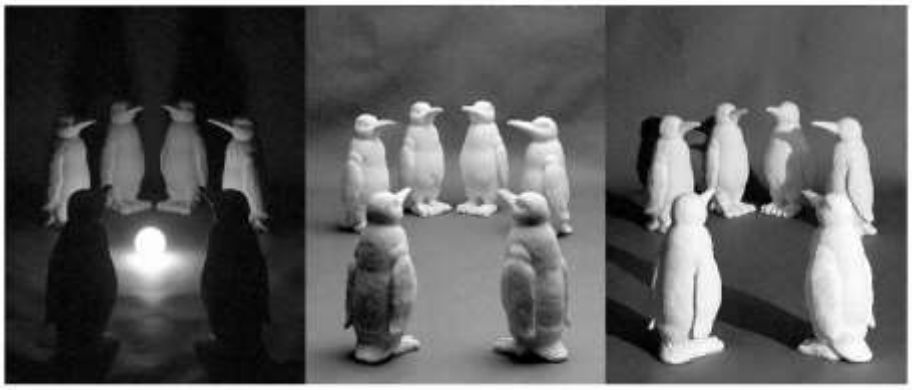


Nota: La imagen muestra una ambigüedad, ya que no se puede diferenciar si es un pato o un conejo. Tomado de *The mind's eye. Popular Science Monthly,* (Jastrow, 1899)

- *Cambios de iluminación*: Un inconveniente muy crítico dependiendo la aplicación de IA, donde se logra obtener las características para realizar una acción de control por visión por computadora en un entorno de luz A, pero cuando se cambia a otro entorno de luz B, el sistema no trabaja de la misma manera tal como se observa en la Figura 13.

Figura 13

Cambios de luz en una imagen



Nota: La imagen muestra las variaciones de luz que puede tener una imagen en un mismo entorno. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada,* (Fernández Castilla, 2013)

- *Cambios de escala*: Cuando se han extraído características de una imagen con una dimensión y escala definida, pero al momento de pasar a la fase de inferencia, el sistema de visión no funciona de la misma manera a pesar de ser una imagen con las mismas características pero con distinta escala y dimensión, tal como se muestra en la Figura 14.

Figura 14

Cambios de escala en una imagen

Nota: La imagen muestra los cambios de escala que puede tener una imagen. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada,* (Fernández Castilla, 2013)

- *Oclusión*: Es la percepción visual de un objeto estando atrás o en frente de otro objeto (Wimmer & Doherty, 2011) tal como se observa en la Figura 15 Esto hace que en muchas aplicaciones, crea falsos negativos, donde si bien el objeto a detectar se encuentra en la escena, este se encuentra rodeado de un objeto al cual no se le desea realizar ningún análisis de visión por computadora.

Figura 15

Fenómeno De la oclusión en una imagen

Nota: La imagen muestra como la oclusión impide distinguir los objetos en una imagen. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada,* (Fernández Castilla, 2013)

- *Movimiento*: Donde la escena se encuentra en constante movimiento y para el sistema de visión es muy complejo detectar el objeto ya que estas imágenes para analizar pueden ingresar distorsionadas tal como se observa en la Figura 16.

Figura 16

Movimiento en una imagen

Nota: En la imagen se aprecia como el objeto a procesar está en movimiento, un problema de los sistemas de visión por computadora. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada,* (Fernández Castilla, 2013)

- Pérdida de información: Este punto ocurre significativamente en el procesamiento de imágenes, ya que el programador en muchos casos no logra optimizar de manera adecuada los diferentes métodos y técnicas para extraer las características necesarias de la imagen procesada.

Muchos de estos aspectos y debilidades del sistema de visión por computadora se han contrarrestado de manera significativa con el sub campo de la IA, como es el aprendizaje profundo.

Aprendizaje Profundo

El aprendizaje profundo es una forma de aprendizaje automático que permite a las computadoras aprender de la experiencia y comprender el mundo en términos de una jerarquía de conceptos. Es un subconjunto de IA que imita el funcionamiento del cerebro humano en el procesamiento de datos y la creación de patrones para su uso en la toma de decisiones. El auge como tal del aprendizaje profundo surge en base al incremento de mejoras de las computadoras, ver en Figura 17 y hardware embebido, aumentado en

23% la disminución del error en aplicaciones de clasificación (Krizhevsky, Sutskever, & Hinton, 2012).

Figura 17

Concurso de clasificación de imágenes imageNET

Nota: La imagen muestra los resultados de los concursos de clasificación de imágenes hasta el 2017. Tomado de *ImageNet Large Scale Visual Recognition Challenge,* (Russakovsky, et al., 2017)

Por lo tanto, el aprendizaje profundo es un sector de métodos de aprendizaje automático basados en redes neuronales artificiales con aprendizaje de representación. Debido a que la computadora recopila conocimiento de la experiencia, no se necesita ningún ser humano para operar la computadora y especificar el conocimiento que necesita la computadora. La jerarquía de conceptos permite que la computadora aprenda de manera autónoma conceptos complicados construyéndose a partir de conceptos más simples. Por lo tanto, un gráfico de estas jerarquías tendría muchas capas de profundidad.

En términos simples, el aprendizaje profundo es una tecnología de software utilizada por los programadores para enseñar a las computadoras a forjar lo que los humanos han estado haciendo desde el principio de los tiempos: aprender con el ejemplo, procesar y filtrar información compleja con los cinco sentidos para producir un resultado final.

Redes neuronales convolucionales (CNN)

Las redes neuronales convolucionales, son una variante de las redes neuronales artificiales (ANN), son una diferenciación del perceptrón multicapa, muy prácticas para aplicación de visión artificial, tales como clasificación, detección y segmentación de imágenes.

Están formadas por múltiples capas de filtros convolucionales de varias dimensiones, a menudo luego de cada capa se incluye una función que permite un mapeo causal no-lineal, se puede decir que sus capas son filtros que se utilizan para pre-procesar una imagen buscando patrones y características propias de una imagen, por lo general estos filtros son numerosos y tienen la capacidad de reconocer características como bordes, esquinas, figuras geométricas como círculos o cuadrados.

Existen diversos filtros muy complejos para aplicaciones más robustas como clasificar animales, tipos de autos, etc. Por lo general este filtraje se encuentra en las últimas capas de la red.

Los patrones y características que generalmente busca son:

- Objetos
- Texturas
- Curvas
- Colores
- Bordes
- formas

Se inicia con una fase de extracción de características, compuesta de neuronas convolucionales, después hay una disminución por muestreo para en definitiva permite obtener neuronas de perceptrón más fáciles para realizar la clasificación final sobre los parámetros extraídos (Obando Cisneros, 2019).

En un modelo bien entrenado la red neuronal debe caracterizarse por tener un sesgo bajo (el modelo genera predicciones correctas con los ejemplos entrenados) y una

varianza baja (el modelo es capaz de generalizar su conocimiento). En la Figura 18 se muestra la arquitectura básica de una red neuronal convolucional:

Figura 18

Arquitectura de red neuronal convolucional

Nota: La imagen muestra la arquitectura básica de una red neuronal convolucional, unida a una capa de clasificación. Tomado de *Arquitectura de red neuronal convolucional,* (Calvo, 2017)

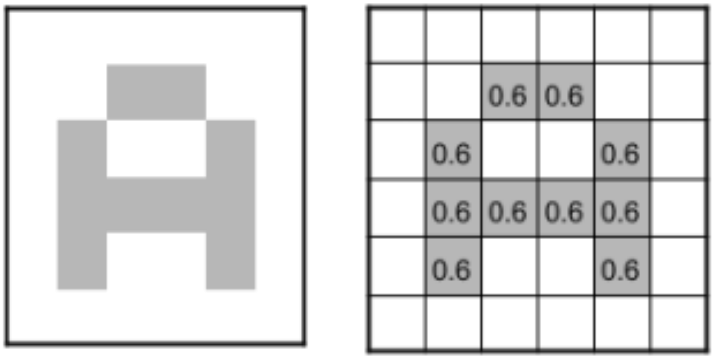
- **Capa de entrada**: Los píxeles de la imagen. Los cuales son, alto, ancho y profundidad. Pueden ser de 1 o 3 canales, esto es el color de la imagen.
- **Capa De Convolución:** Cumple la función de analizar la imagen con una ventana deslizante, se enfoca en una pequeña parte de dicha imagen para obtener un mapa de características. Este paso se logra mediante una serie de filtros que son los encargados de extraer y resaltar los diversos patrones y rasgos en el ejemplo se utiliza 32 filtros, se puede decidir la cantidad que se necesite dependiendo la aplicación siendo este aspecto el volumen de salida.
- **Capa ReLu**: Se emplea una función de activación en cada elemento de la matriz, conservando los datos positivos e ignorando los datos negativos.
- **Pool ó Subsampling**: Se realizará una reducción en las dimensiones altas y anchas de la imagen, pero manteniendo las características más altas.
- **Capa Tradicional o Clasificación**: Es la encargada de conectar con la última capa y concluye con el número de neuronas que se desea clasificar.

Aprendizaje de una red neuronal convolucional

Se inicia con una imagen con dimensiones de 28x28 la cual es digitalizada, para que la pc pueda entender, tal como se muestra en la Figura 19.

Figura 19

Izquierda: Imagen en escala de grises a procesar; Derecha: Imagen convertida a una matriz de píxeles



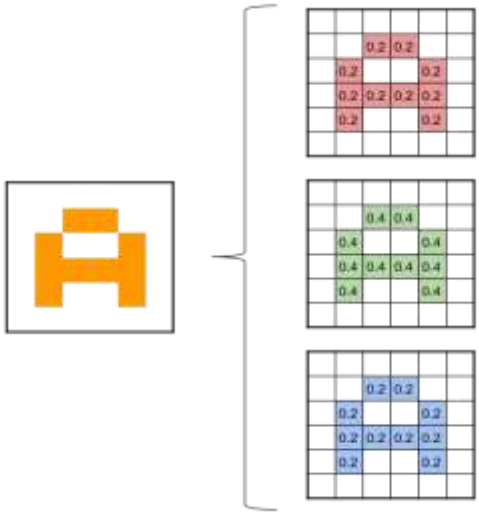
Nota: El gráfico de la derecha muestra la imagen que se le realiza un procesamiento, y la imagen de la izquierda es la imagen en matriz de píxeles de 0 a 255 pero normalizada de 0 a 1. Tomado de *Redes Neuronales Convolucionales,* (Barrios, 2021)

Posteriormente se toma la matriz de la imagen procesada para la capa de entrada de la red si esta imagen es de 28x28, esto quiere decir que existen aproximadamente 784 neuronas, en el caso de que la imagen está en escala de grises, pero si es de color, se debe multiplicar por 3 ya que es la escala RGB, Siendo las neuronas 2352, esto compone la capa de entrada de la red neuronal convolucional, tal como se observa en Figura 20.

.

Figura 20

Izquierda: Imagen de color a procesar; Derecha: Imagen convertida en 3 matrices de pixeles por su escala en RGB



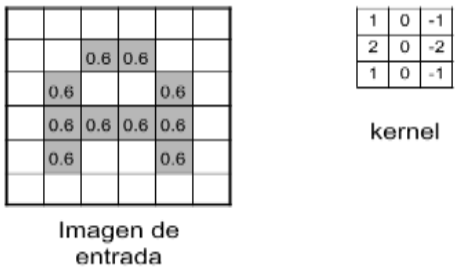
Nota: La figura de la derecha muestra la imagen de color, que se le realiza un procesamiento, y la imagen de la izquierda es la imagen dividida en matrices de pixeles de 0 a 255 pero normalizada de 0 a 1. Tomado de *Redes Neuronales Convolucionales,* (Barrios, 2021).

Antes de ingresar los datos a la red, es preferible para este caso normalizar los datos, ya que la escala RGB entrega valores en un rango de 0 a 255, por lo tanto se divide para 255, tal como se mostró en la Figura 19 y Figura 20.

A continuación se inicia las convoluciones, en este paso consiste en tomar un conjunto de píxeles próximos de la imagen que ingresa para de esta manera operar matemáticamente (producto punto) hacia una matriz más corta denominada kernel, tal como se muestra en la Figura 21. El kernel para fines de explicación, es de 3×3 píxeles de tamaño (puede ser de distinto tamaño según la aplicación) y con ese tamaño logra observar todas las neuronas que ingresan (tanto de arriba como de abajo, asi como a los lados de izquierda a derecha) y así logra crear matriz nueva de salida, que será una nueva capa denominada como neuronas ocultas.

Figura 21

Izquierda: Imagen de a procesar; Derecha: Kernel o filtro.

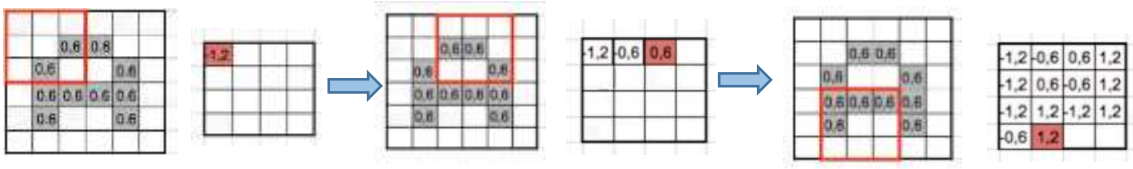


Nota: La figura de la derecha muestra la imagen que se le realiza un procesamiento, y la imagen de la izquierda es el kernel que va recorriendo la imagen extrayendo las características para la convolución. Tomado de *Redes Neuronales Convolucionales,* (Barrios, 2021)

Ahora una características de las redes neuronales convolucionales que explica por qué está tan arraigado a la sub-campo de IA, como lo es el aprendizaje profundo es que, a la imagen anterior, ver Figura 22, no solo se le aplica un kernel, sino que se le aplica muchos kernel's, que en definitiva toman el nombre de filtros, esto suponiendo que para esta primera capa oculta se obtienen 32 filtros siendo un total de 25088 neuronas. Siendo este resultado elevado, ahora se deduce por simple inspección cual fuera el número si la imagen fuera de color y con dimensiones de 224x224.

Figura 22

Recorrido del kernel en la imagen extrayendo características

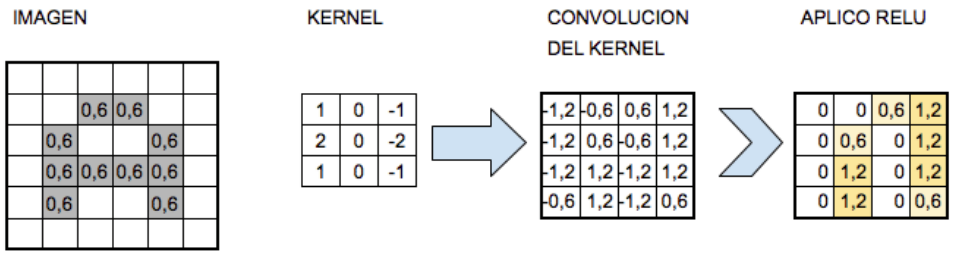


Nota: La figura muestra como el kernel va recorriendo la imagen extrayendo las características para la convolución. Tomado de *Redes Neuronales Convolucionales,* (Barrios, 2021).

Conforme la matriz kernel se va desplazando, va obteniéndose una imagen nueva que fue filtrada por esta matriz kernel, tal como se observa en la Figura 23. En esta primera convolución y siguiendo lo que se mencionó anteriormente, es como si se obtuviera 32 imágenes filtradas nuevas. Al obtener estas nuevas imágenes se ha logrado captar ciertos rasgos y patrones de la imagen que ingreso. De esta forma más adelante se puede apoyar en poder diferenciar un objeto distinto de otro (por ejemplo. carro o persona),

Figura 23

Aplicación de la función ReLu



Nota: La figura muestra la aplicación de la función ReLu que separa los valores positivos de los negativos luego de aplicado el kernel. Tomado de *Redes Neuronales Convolucionales,* (Aprende Machine learning, 2021)

Ahora se utiliza una función de activación, una de las más comunes usada se denomina ReLu, que tiene por nombre unidad lineal rectifica (Traducida al español), permite transformar los valores introducidos, eliminando los valores negativos y solo tomando los valores positivos.

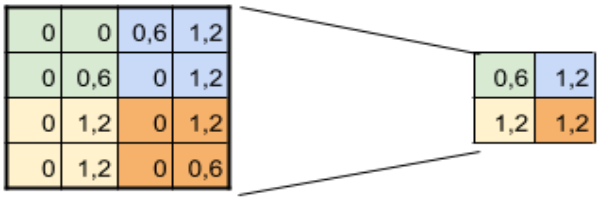
Muestreo

En el siguiente paso se toma una muestra de las neuronas más representativas este paso se debe realizar previo a una nueva convolución. Si se observa anteriormente en una imagen carente de color (blanco-negro) de 28x28 pixeles se consiguió una capa de entrada de 784 neuronas y posterior de esta capa inicial de convolucion se obtuvo una capa oculta de 25.088 neuronas (las cuales en realidad vendrían a ser las 32 mapas de

rasgos y patrones de 28×28 ((28 x 28) x 32), ahora si a partir de este punto se realiza otra convolución, la cantidad de neuronas de la siguiente capa requeriría un poder computacional significativo. Por ello y para disminuir la cantidad de la próxima capa de neuronas tal como se observa en la Figura 24, se realiza un muestreo preservando los patrones más importantes que obtuvo cada filtro.

Figura 24

Aplicación del muestreo o subsampling y Max-Pooling

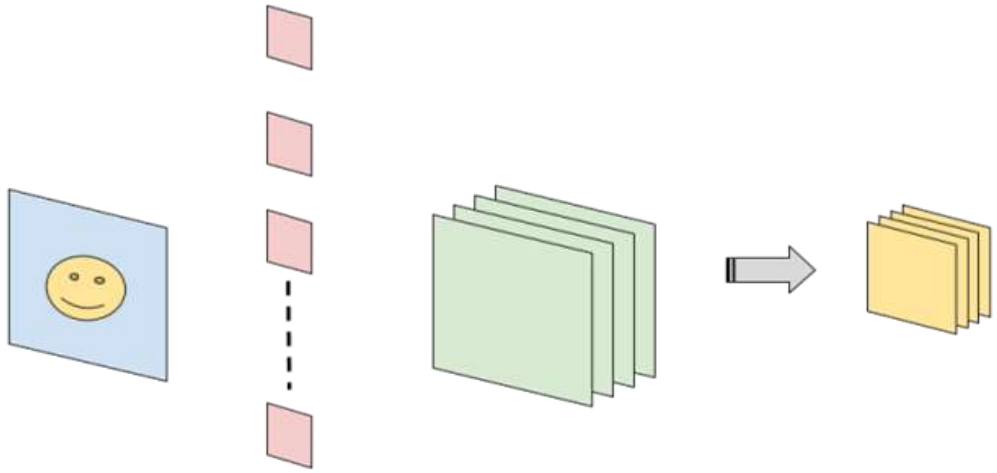


Nota: La figura muestra la aplicación de la técnica denominada muestreo o subsampling y Max-Pooling, donde toma los valores más relevantes reduciendo la salida a la mitad. Tomado de *Redes Neuronales Convolucionales,* (Aprende Machine learning, 2021)

Existen diversos muestreos como apoyo al punto anteriormente explicado, el más usado es el Max-Pooling, esta técnica utiliza una matriz con un tamaño de 2×2 y puede ser más grande la dimensión dependiendo la aplicación. Esta explicación menciona que se desplaza a través de las 32 imágenes de características conseguidas anteriormente con un tamaño de 28x28 píxeles recorridas tanto de arriba hacia abajo como de izquierda a derecha. Pero en lugar de tomar de cada píxel, lo que hace ahora es tomar de 2×2 (con 2 de ancho y 2 de alto siendo ahora de 4 píxeles) resguardando el valor más alto de entre esos 4 píxeles (razón por la cual toma el nombre de “Max”). Para este ejemplo, usando una matriz de 2×2, la imagen de salida es reducida “a la mitad” y quedará de 14×14 píxeles. Posterior a este proceso denominado sub-muestreo se queda con 32 imágenes con un tamaño de 14×14, dando como resultado de 25.088 neuronas a 6272, donde existe una disminución que en teoría, debe continuar guardando la información más relevante para detectar características requeridas.

Figura 25

Primera convolución



Nota: La figura muestra sintetizado como se realiza los pasos de la convolución para obtener las salidas con las características de las imágenes a procesar. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada,* (Fernández Castilla, 2013)

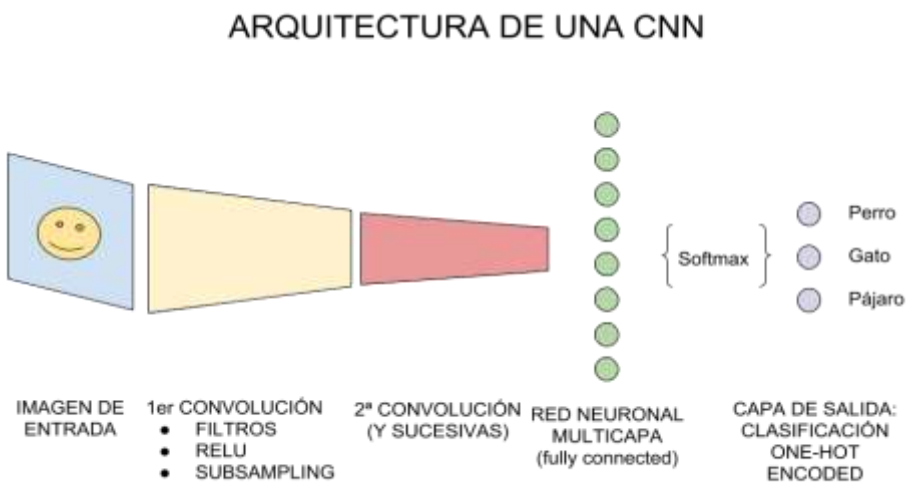
La Figura 25 representa la primera convolución: donde se tiene en la entrada, una serie de filtros, se genera un mapa de características y se procede a realizar el sub-muestreo. Por consiguiente, en el caso de imágenes de 1 único color se tendría los siguientes parámetros:

Conforme se desarrolla y se agregan más capas convoluciones, los mapas de características tendrá la capacidad de identificar formas más complejas, y el conjunto total de capas de convoluciones ya reconocerá al objeto entrenado-ajustado al sistema.

Para finalizar, se incluye la capa final oculta a la que le realizó subsampling, tal como se observa en la Figura 26, y a través de este método se aplana, cuyo significado es que abandona su aspecto tridimensional pasando a ser una capa de neuronas tradicionales, y con ello se aplana (y conecta) una nueva capa oculta de neuronas tipo feedforward.

Figura 26

Conexión a una red tradicional



Nota: La figura muestra simplificado como está compuesta las redes neuronales convolucionales así como sus diferentes secciones que la componen. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada,* (Aprende Machine learning, 2021)

Por tal razón esta capa oculta tradicional creada, se aplica una función llamada softmax, cuya función es de pasar la probabilidad (entre 0 y 1) en la salida de las neuronas. Entendiendo mejor con un ejemplo se tiene que si una salida con dos clases (perro y gato) y se tiene un vector con los siguientes parámetros de salida [0,2 0,8] nos revela que existe 20% de probabilidades de que un objeto sea un perro y 80% que pueda ser un gato.

La función softmax se conecta con la capa de salida final, esta posee el número de neuronas pertenecientes con las clases que se están clasificando. Dependiendo la aplicación se tendrá las salidas, para un par de ejemplos, si se clasifica perros y gatos, existirán 2 neuronas pero en cambio sí se clasifica autos, aviones o barcos habrán 3, y así sucesivamente dependiendo las clases que pueda tener el sistema.

Las salidas, cuando el modelo es entrenado tienen la forma denominada one-hot-encoding, estas salidas para los dos ejemplos mencionados anteriormente será [1,0] y [0,1] para carros y personas y [1, 0, 0]; [0, 1, 0]; [0, 0, 1] para el ejemplo de autos, aviones y barcos.

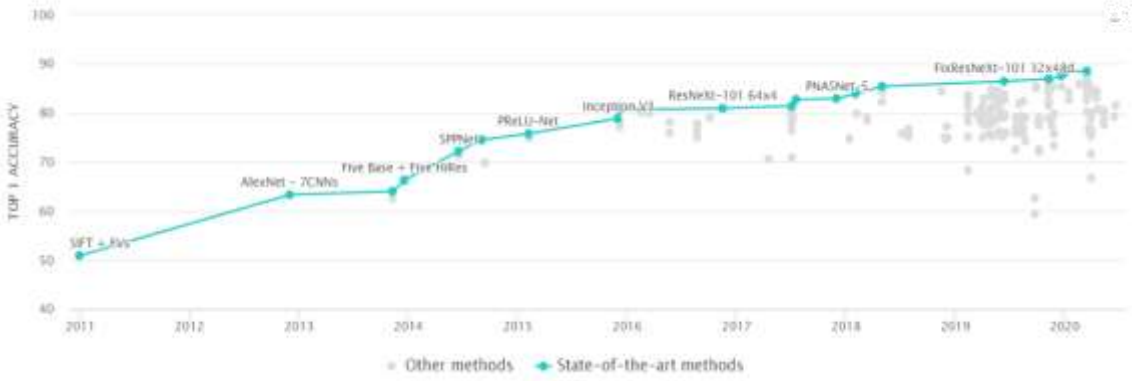
Arquitecturas de las redes neuronales convolucionales.

Con objetivos de desarrollo del presente trabajo de investigación, se han determinado los métodos y arquitecturas en dos de las tantas aplicaciones que se pueden realizar con el aprendizaje profundo, las cuales son las siguientes:

- **Clasificación de imágenes**: Con técnicas de filtrado de imágenes y procesamiento de imágenes antes usado, clasificar un objeto de otro era muy complejo, pero con modelos pre-entrenados de redes neuronales convolucionales esta tarea se volvió algo más sencillo. Entre las arquitecturas más conocidas se tiene *AlexNet, VGG-18, VGG-19, GoogleNet, MobileNetV2, Resnet,* se detalla en la Figura 27 el orden de precisión así como su ranking.

Figura 27

Ranking de arquitecturas de redes neuronales avanzadas para clasificación de imágenes

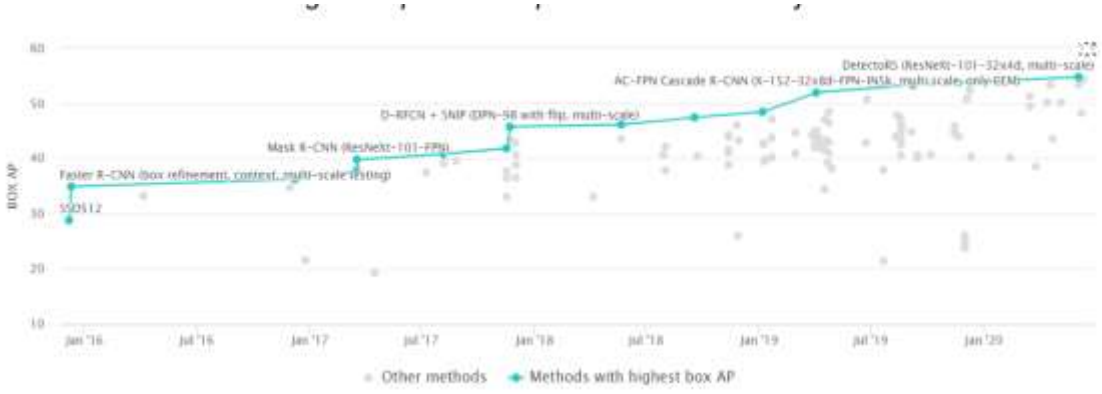


Nota: El gráfico muestra un ranking de arquitecturas CNN, entrenadas para aplicaciones de clasificación con el dataset de ImageNet. Tomado de *Object Detection on COCO test dev,* (Paper With Code, 2021)

- **Detección de objetos**: En combinación con las técnicas anteriores, mezclan las arquitecturas para descriminar un objeto de otro de una forma muchísimo más eficiente, se muestra en la Figura 28, el orden ascendente su precisión sobre un entrenamiento de datos de COCO.

Figura 28

Ranking de arquitecturas de redes neuronales avanzadas para detección de objetos



Nota: El gráfico muestra un ranking de arquitecturas CNN, entrenadas para aplicaciones de detección de objetos. Tomado de *Object Detection on COCO test-dev,* (Paper With Code, 2021)

Esos dos aplicativos del aprendizaje profundo son vitales para el desarrollo del presente sistema de reconocimiento facial, ya que con esto se puede clasificar los rostros y de esta manera poder detectarlos y en definitiva identificarlos.

Diferencias de la IA y sus campos de acción

Resumiendo, la IA, son todas las técnicas que otorgan a las computadoras replicar a los humanos usando hardware y software de baja escala para el desarrollo de proyectos, mientras el aprendizaje automático (machine learning por sus siglas en inglés), es el subconjunto de IA, que utiliza métodos estadísticos para dar acceso a que las máquinas mejoren las experiencias con respecto a ese aprendizaje adquirido, utilizando grandes cantidades de datos para aprender y extraer de ellos las mayores características posibles pero utilizando hardware y software de mediana-alta escala.

Finalmente el aprendizaje profundo (deep learning, por sus siglas en inglés), es un subconjunto de ML, siendo este tópico que sea factible el cálculo de la red neuronal multicapa que requieren tecnología de punta como tarjetas gráficas (GPU, Graphic Process Unit por sus siglas en inglés), por los complejos y extensos cálculos matemáticos que demandan para así obtener sistemas sofisticados con precisión y efectividad elevada, siendo este último campo el futuro de la automatización inteligente.

Métodos y algoritmos de detección facial y reconocimiento facial.

Detección Facial con el algoritmo de Viola-Jones

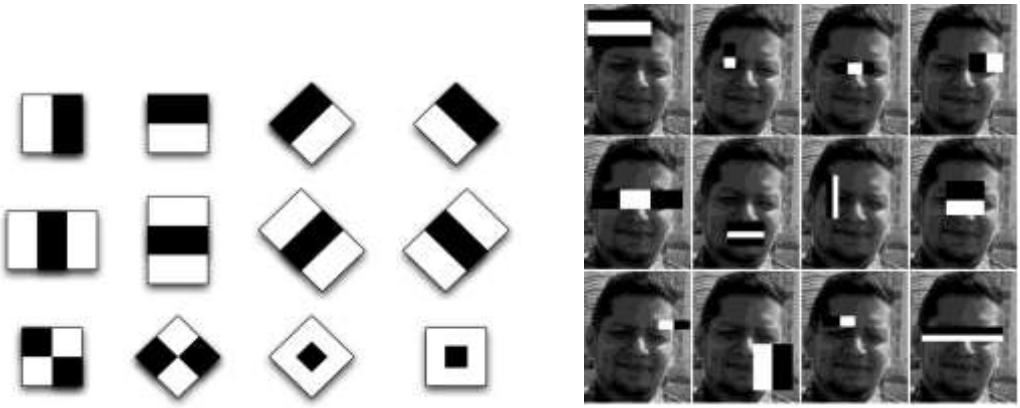
El algoritmo de Viola-Jones es un algoritmo de detección de rostros que no requiere altos recursos computacionales, otorga la facilidad de emplearse en tiempo real (Viola & Jones, 2001).

Su creación fue incentivada por el inconveniente de la detección de caras, sigue siendo ampliamente utilizado, en diversas detección de objetos en los cuales están caracterizados por patrones típicos de iluminación.

El concepto de este algoritmo se basa en la comparación entre las intensidades luminosas de regiones rectangulares de las imágenes, efectúa la clasificación mediante características en vez de píxel a píxel, tiene una tasa de detección de 99.7% (Mur Igualada, 2015). De este método se derivan algunas definiciones:

Figura 29

Izquierda: Algoritmo de viola-Jones; Derecha: Filtros Haar.



Nota: La figura muestra a la izquierda cómo está compuesto el algoritmo de Viola, en la derecha se muestra la aplicación del algoritmo con clasificadores Haar. Tomado de *Detección de Rostros basado en Filtros Haar + Adaboost,* (Pardo, 2018)

- **clasificadores HAAR:** Denominado como clasificador en cascada, muy usado para reconocimiento de objetos en procesamiento de imágenes, es entrenado por medio de cientos de muestras del objeto tanto verdaderas como falsas que pueda discriminar la forma del objeto que desea, tal como se observa en Figura 29.
- **Imagen integral:** A menudo usada para una expeditiva evaluación de los parámetros.
- **Algoritmo AdaBoost:** Ligado al aprendizaje automático, permite aprender un comportamiento del conjunto de datos y así mejorar su rendimiento.

Ventajas

- Alta velocidad al momento de procesar los datos
- No depende de localización ni escala de la imagen.
- Trabaja con múltiples filtros y así disminuye el porcentaje de falsos positivos.

Desventajas

- Su alta eficiencia al momento de la detección solo lo realiza en imágenes de caras frontales.

- Debido a que está entrenado para determinar el rostro medio de formas rectangulares, no puede detectar caras que tengan un ángulo cercano a 45°, tanto vertical como horizontalmente.
- Muy sensible a cambios de iluminación y entornos distintos al entrenado.

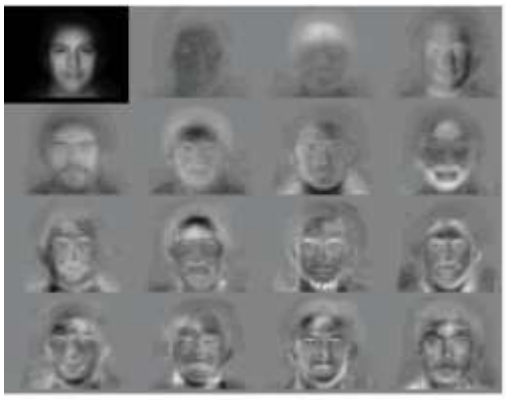
Detección Facial con PCA

Denominada como análisis de componentes principales, es un tipo de aprendizaje no supervisado, para reducir la dimensionalidad, esto quiere que capturado la mayoría de información simplificando y reduciendo a aspectos principales a considerar (Sirovich & Kirby, 1987).

Esta técnica adquiere las características particulares de cada una de las personas, donde las imágenes de los rostros deben tener el mismo tamaño, tanto para predecir como para entrenar el modelo, y de esta manera poder ubicar de forma general y obtener un patrón donde se encuentra cada parte del rostro, ya sea la nariz, la boca de cada individuo, este propósito se logra con la característica de reducción de dimensionalidad y tomando solo aquella información no correlativa, estos vectores toman el nombre de eigenfaces tal como se observa en la Figura 30.

Figura 30

Características de los vectores usando eigenfaces.



Nota: La figura muestra los vectores para obtener las características de la cara de cada individuo Tomado de *Low-dimensional procedure for the characterization of human faces*, (Sirovich & Kirby, 1987)

Por medio de las imágenes se desarrolla una memoria que despliega un conjunto de datos con solo una dimensión, obteniendo una imagen a detalle de los rasgos y características irrepetibles para cada persona. El PCA, requiere una imagen de rostro completo para obtener un rendimiento pleno en cuanto a la identificación de rostros en la proporción 1/1000 en sistemas biométricos (Biometrica, 2015).

Ventajas

- Puede representarse en un grupo de imágenes con una dimensión más reducida.
- Tiene muy buenos resultados de las imágenes de los rostros a reconocer cuando estas se encuentran a igual distancia, nivel de iluminación y posición que las imágenes que están en la base de datos.

Desventaja

- La gran cantidad de parámetros que influyen y perturban a los sistemas de identificación facial, producen además que no se pueda tener una fiabilidad en esta técnica, ya que características como cambios de peinados, barba, y edad afectan de sobremanera a los sistemas que aplican este método.
- Se necesitan sistemas de iluminación controlados para evitar cambios de entornos en cuanto a variación de posición e iluminación se refiere.

Detección Facial con MTCNN

La detección de rostros y su alineación en ambientes sin limitaciones son un reto debido a diversas posturas, cambios de iluminación y oclusiones. Basado en los enfoques de aprendizaje profundo se ha logrado alcanzar un rendimiento extraordinario en estas dos tareas (Adamczyk, 2021).

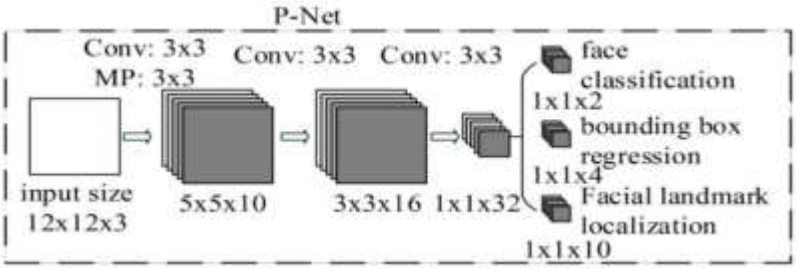
Esta técnica acoge una estructura en cascada con tres etapas de redes neuronales convolucionales profundas eficazmente diseñadas que pronostican la ubicación de la cara y el punto de referencia de una manera gruesa a fina.

Primero, modificar el tamaño de la imagen varias veces para detectar rostros de diferentes tamaños. A continuación, la red P (propuesta) escanea imágenes y realiza la

primera detección, tal como se muestra en la Figura 31. Esta salida de la red posee bajo umbral de detección y, por lo tanto, y tiene una tendencia a detectar varios falsos positivos, incluso después de NMS (supresión no máxima), lo cual no quiere decir que es un error, ya que la arquitectura de la red funciona así a propósito.

Figura 31

Red P

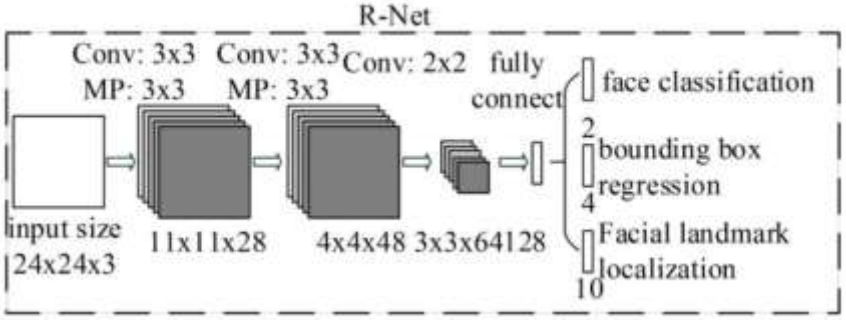


Nota: La figura muestra la red p de la arquitectura del modelo MTCNN. Tomado de *Architecture of MTCNN, Network-P*, (Zhang, Zhang, Li, & Qiao, 2016)

Estas regiones propuestas, que tienen falsos positivos, ingresan a la segunda red, la red R (Refine), tal como se muestra en la Figura 32, filtra las detecciones incluso las que tienen NM, obteniendo cuadros delimitadores bastante precisos.

Figura 32

Red-N

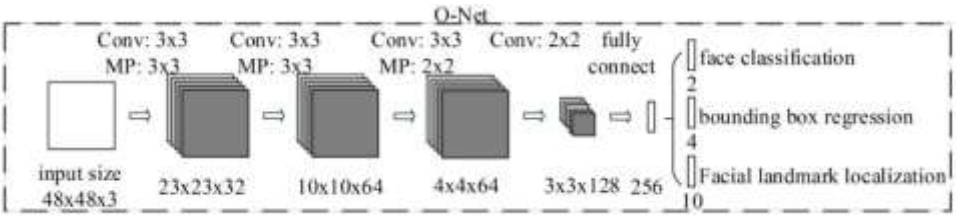


Nota: La figura muestra la red-n de la arquitectura del modelo MTCNN. Tomado de *Architecture of MTCNN, Network-N*, (Zhang, Zhang, Li, & Qiao, 2016)

En la etapa final, la red O, (Salida) realiza el refinamiento final de los cuadros delimitadores, ver Figura 33. De esta manera, no solo se detectan caras, sino que los cuadros delimitadores son muy correctos y precisos (Zhang, Zhang, Li, & Qiao, 2016).

Figura 33

Red-O



Nota: La figura muestra la red-o de la arquitectura del modelo MTCNN. Tomado de *Architecture of MTCNN, Network-O*, (Zhang, Zhang, Li, & Qiao, 2016)

En el proceso de entrenamiento, propone una nueva estrategia de minería de muestras duras en línea que puede mejorar el rendimiento automáticamente sin la selección manual de muestras. Esta técnica logra una precisión elevada sobre diversas aplicaciones vanguardistas para encontrar puntos de referencia faciales en la detección de rostros tales como FDDB y WIDER FACE, y también en la alineación de rostros como lo es AFLW, intentando conservar el rendimiento en tiempo real (Adamczyk, 2021)

Ventajas

- MTCNN es muy preciso y robusto.
- Detecta con mucha precisión rostros incluso con diferentes tamaños, cambios de iluminación y diferentes ángulos del rostro.
- También utiliza información de color, ya que las CNN obtienen imágenes RGB como entrada, algo que el algoritmo de viola-jones sería muy difíciles de implementar.

Desventaja

- Es un poco más lento que el detector Viola-Jones

- Necesita más recursos computacionales al ser un algoritmo de aprendizaje profundo.

Detección facial con MXNET-InsightFace

MXNet, es un marco de trabajo de aprendizaje profundo creado para brindar flexibilidad en los algoritmos desarrollados por esta plataforma. En este marco de desarrollo permite unir programación de diversos lenguajes para maximizar la productividad de las aplicaciones sobre todo en IA. MXNET es liviano y portátil, con escalamiento a diversas GPU's (Li, et al., 2015).

Con el concepto anterior tenemos ahora el modelo de MXNET, denominado InsightFace, el cual contiene varios aplicativos y algoritmos para el análisis facial con técnicas de aprendizaje profundo en 2D y 3D, siendo este de libre uso.

Dentro de su marco de aplicativos para el análisis del rostro tiene los siguientes algoritmos:

- Reconocimiento facial profundo: formas de entrenar, modelos pre-entrenados, 512-D características embebidas y combinación con diferentes marcos de desarrollo.
- Detección de rostros: con los modelos RetinaFace y además RetinaFaceAntiCov, el cual es un modelo pre-entrenado que le suma a las funciones de detección del rostro, si el usuario posee o no mascarilla.
- Alineación facial: En combinación con el modelo MTCNN, ha creado dos modelos como DenseUNet y CoordinateNet, simplificando la obtención de las características del rostro conforme a la inclinación y orientación del mismo.

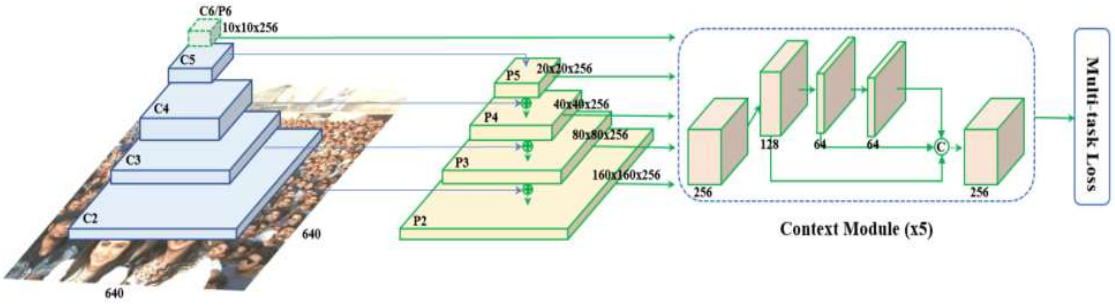
Detección facial RetinaFace

Es un modelo basado en los desarrollos de MXNET, el cual permite combinar con otros marcos de desarrollo para aplicaciones de IA, para el proyecto de investigación propuesto se ha desarrollado en el framework de Pytorch (Minivision, 2016). Para el

reconocimiento de una cara se deben considerar 4 etapas que son: detección, alineación, representación y verificación, es ahí donde RetinaFace tiene en las primeras etapas una elevada puntuación de confiabilidad (Serengil, Sefik Ilkin, 2021).

Figura 34

Arquitectura del modelo RetinaFace



Nota: La figura muestra el proceso de obtención de características del modelo de RetinaFace para detectar del rostro. Tomado de *Deep Face Detection with RetinaFace in Python,* (Serengil, Sefik Ilkin, 2021)

Estructura

Está basado en técnica de pirámides características, tal como se observa en Figura 34 donde crea copias de la imagen original a distintas escalas creando los niveles de una pirámide por luego pasar al clasificador o red y de esta manera detectar el contenido de las imágenes en diferentes planos (Zhuo, et al, 2019).

Utiliza una arquitectura de redes neuronales profundas llamada ResNet152, la cual se modifica según el marco de desarrollo se lo implemente (en Tensorflow se utiliza ResNet50) (Serengil, Sefik Ilkin, 2021).

Métodos de extracción de características por medio de puntos faciales

Face landmask 86

Dentro del contexto de los puntos de referencia faciales, el objetivo a detectar es la estructura facial como tal, es decir predecir dónde se encuentra la nariz, boca, ojos, etc.

Existen diversas técnicas que van desde procesamiento de imágenes, con clasificador en cascada HAAR antes estudiados, métodos de aprendizaje automático como SVM o árboles de decisión incluso el aprendizaje profundo. En si no es importante el método que se aplique, sino más bien encontrar el rostro y una vez deducido este paso predecir cada una de sus partes.

En base la técnica de aprendizaje automático, con la librería DLib de Python, se encuentra adaptada para extraer estas características faciales en dos sencillos pasos:

- Inicia con un conjunto de entrenamiento de puntos de referencia faciales etiquetados en las imágenes. Este conjunto de imágenes se las etiqueta de forma manual, dictando coordenadas específicas (x,y), de cada región que posee el rostro.
- Le da prioridad a la distancia entre pares de píxeles de entrada mediante su probabilidad.

Tabla 3

Modelo Dlib-68 rasgo faciales

Modelo	Características
Face_predictor-68.dat	Modelo entrenado con SVM+HOG de la librería Dlib, para determinar los puntos de referencia del rostro.

Nota: Esta tabla indica el nombre del modelo entrenado con SVM+HOG para 68 rasgo faciales

SVM

Los vectores de soporte (traducido del inglés Support Vector Machine), se utiliza tanto en predicciones del tipo de clasificación como de regresión, a continuación se da algunos aspectos relevantes de este modelo de aprendizaje automático:

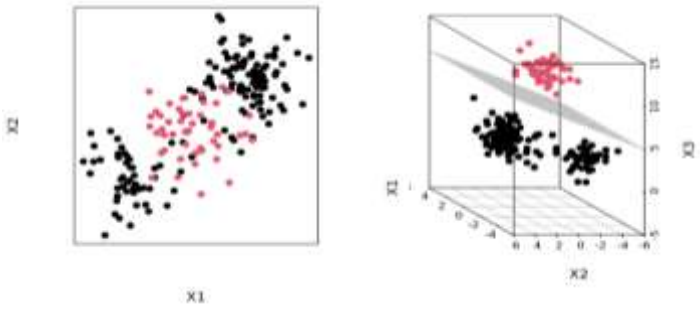
- Este algoritmo es un clasificador discriminatorio definido por un hiperplano de separación, en otras palabras, dados los datos de entrenamiento etiquetados, el algoritmo genera un hiperplano óptimo que clasifica los datos en diversos

espacios dimensionales esto quiere decir que este hiperplano divide un plano en distintas partes donde cada clase se encuentra alojada.

- Cuando los datos son lineales, las máquinas de soporte separa estos datos de creando un margen de distancia amplio para que cada clase y de esta forma discriminar de una forma más óptima, tal como se muestra en la Figura 35.

Figura 35

SVM para clasificación de datos lineales, a la izquierda los datos lineales a clasificar; a la derecha los datos del tipo lineal clasificados

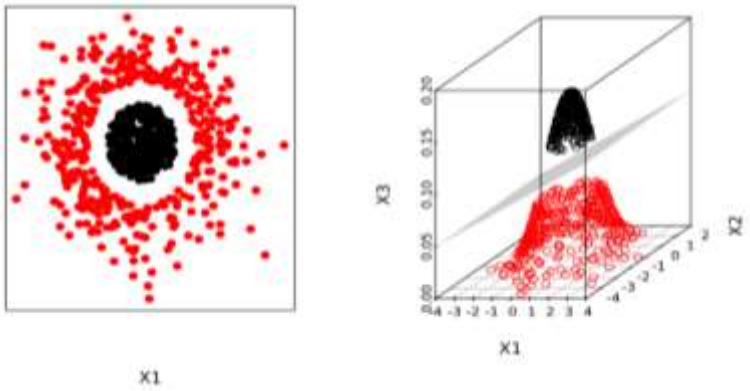


Nota: El gráfico representa como trabaja el modelo SVM en clasificación el cual separa los datos lineales con una nueva dimensión. Tomado de *Máquinas de Vector Soporte (Support Vector Machines, SVMs)*, (Amat Rodrigo, 2021)

Si los datos no son linealmente separables, se aplican kernel, estos kernels permiten mapear los datos de forma no lineal, para llevarlo a un espacio de alta dimensión donde los datos se vuelven linealmente separables, ver Figura 36.

Figura 36

SVM para clasificación de datos no lineales, a la izquierda los datos lineales no separables; a la derecha los datos del tipo no lineal clasificados



Nota: El gráfico representa la clasificación de datos no lineales con el modelo SVM. Tomado *Máquinas de Vector Soporte (Support Vector Machines, SVMs)*, (Amat Rodrigo, 2021)

Entonces ya explicado cómo funcionan las máquinas vectores de soporte, para el modelo de 68 rasgos faciales, se clasifican los datos en varias dimensiones para obtener correctamente las facciones del rostro. Tal como se muestra en la Figura 37. Se utiliza una base de datos de Ibug 300W (Sagonas, Antonakos, Tzimiropoulos, Pantic, & Zafeiriou, 2016), en ella se encuentra una serie de rostros alineados para poder extraer de manera ágil y rápida las facciones del rostro. Se puede obtener desde 194 hasta 468 puntos, dependiendo del aplicativo que se requiera.

Una vez tomada esta de datos se separan los datos aplicando el modelo SVM para así poder clasificar las clases y predecir a qué clase pertenecen los rasgos faciales de los ojos nariz, boca, cejas y de esta manera se obtiene un modelo donde localizar cada punto facial.

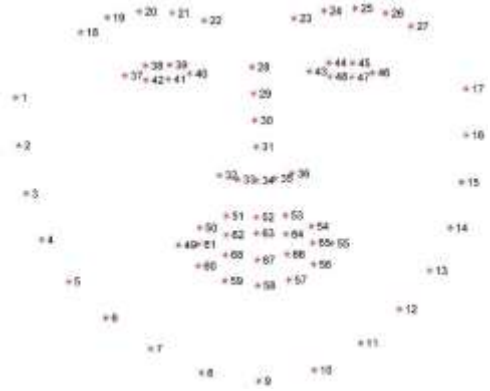
Ya proporcionados estos datos, procede a entrenarse un modelo que estima las posiciones de referencias faciales directamente con cada intensidad de los pixeles,

extrayendo características de cada imagen etiquetada, dando como resultado un predictor de puntos faciales de alta calidad y en tiempo real (Rosebrock, 2017).

Este predictor como tal, extrae 68 coordenadas faciales asignando cada zona referencial a un sector del rostro tal como se observa en la Figura 37.

Figura 37

Modelo de Dlib para 68 rasgos faciales



Nota: La figura se observa las ubicaciones por números de las partes de rostro. Tomado de *Facial landmarks with dlib, OpenCV, and Python,* (Rosebrock, 2017)

La siguiente Tabla 4, se muestra la localización de los rangos de los vectores de las facciones del rostro.

Tabla 4

Diversos sistemas biométricos en la industria

Puntos de referencia	Distribución
Mandíbula	0-16
Ceja izquierda	17-21
Ceja derecha	22-26
Nariz	27-35
Ojo derecho	36-41
Ojo izquierdo	42-47
Boca	48-68

Media Pipe-468 face landmask

Media Pipe, es un marco de desarrollo (traducido del inglés, framework) que ofrece soluciones con aprendizaje automático y profundo en aplicaciones de visión por computador y en tiempo real. Es multiplataforma y perfeccionado que no requiere un hardware de gran escala para correr sus modelos (MediaPipe, 2020). Creado y desarrollado por Google, tiene en su repertorio un vasto set de aplicaciones de visión por computadora, en la siguiente tabla se detallan algunas de sus aplicaciones.

Tabla 5

Diversos sistemas biométricos en la industria

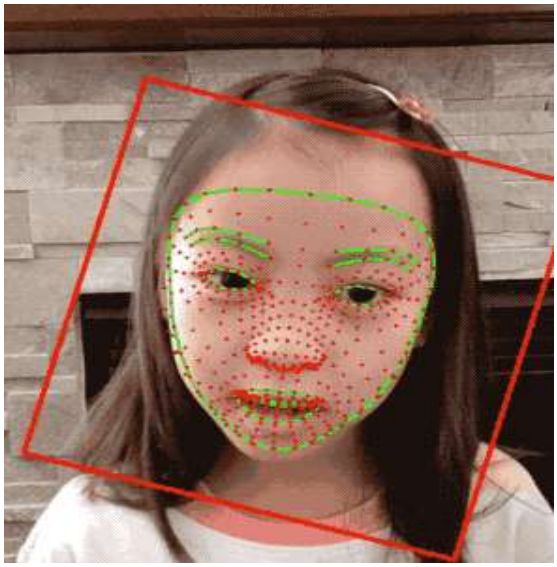
	Plataforma de desarrollo				
Aplicación	**Android**	**iOS**	**C++**	**Python**	**JS**
Detección de rostro	Si	Si	Si	Si	Si
Rasgos faciales 468	Si	Si	Si	Si	Si
Detección de iris	Si	Si	Si	-	-
Detección de objetos	Si	Si	Si	-	-
Estimación de pose	Si	Si	Si	Si	Si
Detección de manos	Si	Si	Si	Si	Si
Objetron: detección de objetos en 3D	Si	-	Si	Si	-

Nota: Esta tabla muestra, un resumen de los diversos aplicativos de visión por computadora que presenta MediaPipe de Google. Tomado de *MediaPipe: Live ML anywher,* (MediaPipe, 2020)

Face mask 468 (traducido del inglés, 468 Rasgos faciales), siguiendo el mismo concepto anterior del modelo de 68 puntos para extraer los rasgos faciales de MediaPipe, proporciona un modelo optimizado para los puntos de referencia de caras con 468 características faciales (Google, 2019), tal como se observa en Figura 38.

Figura 38

Modelo de Media Pipe con 468 rasgos faciales



Nota: La figura se observa los rasgos faciales por medio de Media Pipe. Tomado de *Media Pipe, Face Mesh,* (MediaPipe, 2020)

Funcionamiento

En la siguiente Tabla 6 se detalla cómo está estructurado el modelo:

Tabla 6

Modelo de detección de 468 rasgos faciales

Nombre Modelo	**Tipo de red**	**Arquitectura**	**Aplicación**
Media Pipe Face Mesh	CNN	MobileNetV2	468 Puntos faciales en 3D

Nota: Esta tabla muestra, un resumen de la estructura del modelo de 468 rasgos faciales en 3D. Tomado de *MediaPipe: Live ML anywher,* (MediaPipe, 2020)

El modelo genera las posiciones de los puntos en 3D, así como la probabilidad de que una cara esté presente y alineada razonablemente en la entrada. Una orientación alternativa común es inferir un mapa de calor 2D para cada punto de referencia, pero no

es apto para la predicción de profundidad ya que esto requiere altos costos computacionales para muchos puntos.

Este modelo ha sido mejorado tanto en precisión como solidez mediante el bootstrapping[1] y el refinamiento iterativo de las predicciones. Así pues se puede obtener predicciones de los rasgos faciales del rostro aun cuando esté presente expresiones como muecas, ángulos oblicuos y oclusiones.

Algoritmos de identificación facial con aprendizaje profundo

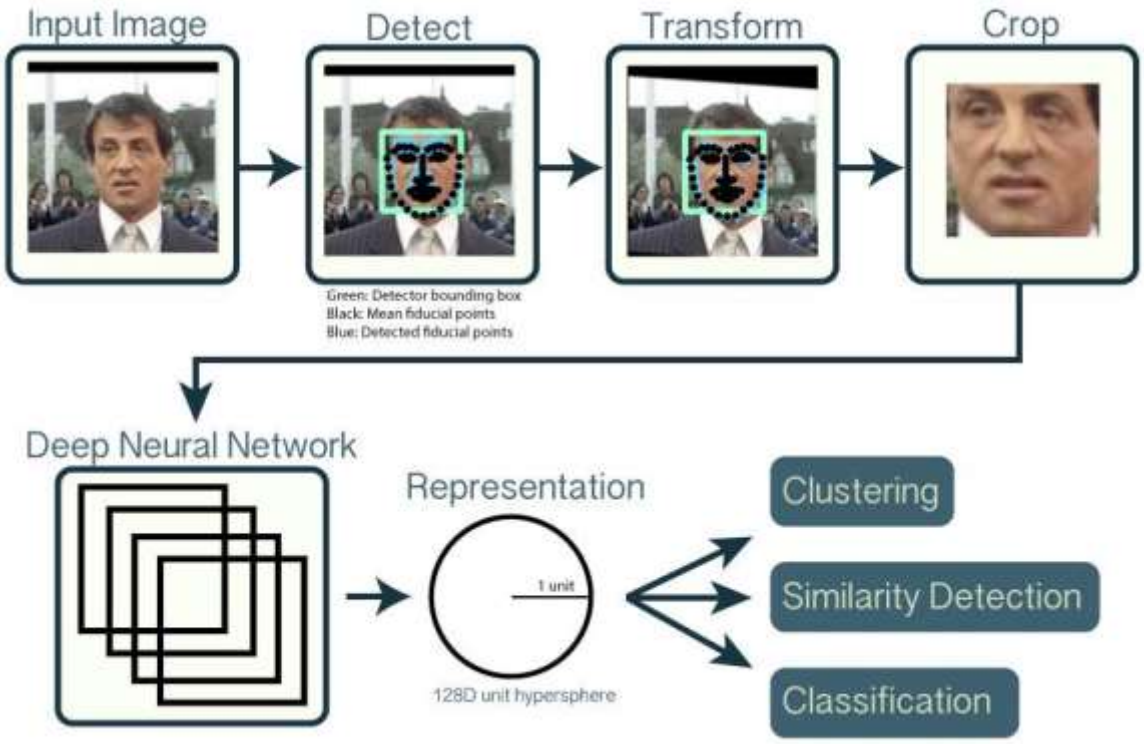
OpenFace

Open face, es un algoritmo de reconocimiento facial de uso libre, el cual contiene una red neuronal profunda permite además cargar algunos modelos externos para así tener un reconocimiento facial mucho más robusto. Tiene como características principales, las detección de puntos referencia faciales, la estimación de postura de la cabeza y estimar la mirada del individuo, tiene muy buena aprobación en cuanto su implementación en producción ya que tiene adaptabilidad a dispositivos móviles y equipos sin altos recursos de hardware (Amos, Bartosz , & Satyanaray, 2016).

[1] Bootstrapping: Usado para crear intervalos de confianza sobre parámetros de interés en los datos.

Figura 39

Arquitectura del modelo de detección de rostros de OpenFace



Nota: La figura muestra el proceso de detección de rostros del modelo de OpenFace. Tomado de *OpenFace: A general-purpose face recognition and library with mobile applications,* (Amos, Bartosz , & Satyanaray, 2016)

Descripción de funcionamiento general

La explicación de este modelo viene dada en la Figura 39 y además en base a los datos de entrenamiento del conjunto de datos LFW (Huang, Ramesh, Berg, & Learned-Miller., 2007) de uso libre se tienen las siguientes conclusiones:

- *Detección*: Verifica si existe un rostro con modelos anteriormente entrenados ya sea con algoritmos de baja escala o de aprendizaje profundo.
- *Alineación*: Ajusta el rostro para la red neuronal, según la información oficial del su sitio web (Amos, Bartosz , & Satyanaray, 2016), utiliza la estimación de pose

en tiempo real de la librería Dlib, para que los ojos y la boca se encuentre en la misma ubicación en la imagen.

- *Representación*: Crea una red neuronal profunda para constituir (o incrustar) la cara en una hiperesfera de unidad de 128 dimensiones. De esta manera se realiza una representación genérica del rostro de cualquier individuo.
- *Verificación*: A excepción de otras formas de obtener el rostro, la incrustación entrega el beneficio de que una mayor distancia entre las incrustaciones de dos caras significa que es probable que los rostros no sean de la misma persona. Esta característica es fundamental para dar facilidad en la detección de similitudes en el reconocimiento facial, en que la distancia euclidiana entre dos rostros no es significativa.

FaceNet

Facenet es un modelo de aprendizaje profundo de uso libre, inspirada en el modelo anterior, proporciona una arquitectura única en aplicaciones de reconocimiento facial, y verificación de persona, mediante redes neuronales convolucionales profundas, y por medio de otras técnicas estadísticas obtiene una precisión elevada con respecto a otros modelos de identificación facial (Schroff, Kalenichenko, & Philbin, 2016).

FaceNet proporciona una arquitectura única para realizar tareas como reconocimiento facial, verificación y agrupación. Utiliza redes convolucionales profundas junto con pérdida de triplete para lograr una precisión de vanguardia.

Descripción del funcionamiento general

Facenet, toma el mismo principio de Openface, realiza el mapeo de cada imagen del rostros dentro de un espacio euclidiano de modo que las distancias en ese espacio correspondan a la similitud de la cara, es decir si tomamos como ejemplo, la imagen de una persona “A”, se tendrá una distancia más cerca con respecto a las demás imágenes alojadas en la base de datos.

La principal diferencia entre FaceNet y otras técnicas es que estudia el mapeo de las imágenes y crea incrustaciones en lugar de usar cualquier otro proceso engorroso para tareas de verificación o reconocimiento facial. Ya una vez creadas estas incrustaciones, todas las demás aplicaciones como reconocimiento facial, se pueden realizar uniendo técnicas convencionales de dominio particular, tomando estas incrustaciones como un vector de características. Se puede aplicar técnicas como KNN, SVM o de agrupamiento para concentrar el rostro junto y solo se necesitaría definir un valor umbral de aceptación para la identificación facial (Kumar, 2019).

La importancia de Facenet es que no desarrolla un algoritmo nuevo, para realizar las funciones mencionadas previamente, sino que solo crea incrustaciones, las cuales se pueden adaptar para el reconocimiento facial.

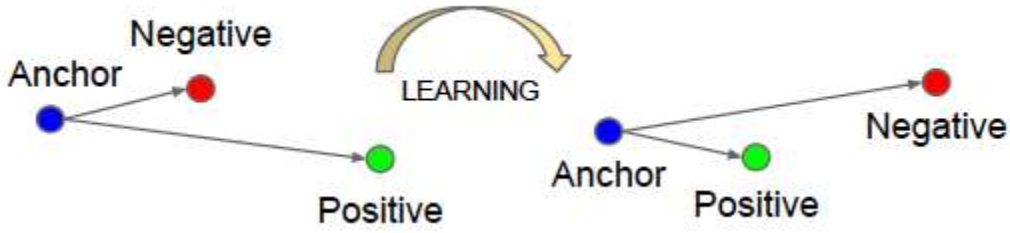
El modelo de Facenet posee modelos pre-entrenados con dos conjuntos de datos, el primer conjunto de datos denominado CASIA-WebFace, fue entrenado con cerca de 500 mil fotos sobre más de 10 mil identidades después de detectar el rostro obteniendo una precisión del 99.05%. El segundo modelo pre-entrenado fue con el conjunto de datos de VGGFace2, en cual se tiene cerca de 3.3 millones de rostros y cerca de 9000 clases, este modelo obtuvo un mejor rendimiento con una precisión del 99.65%.

Función de pérdida triple

Como se sabe, Facenet utiliza redes neuronales convolucionales (CNN), esta red está entrenada de manera que la distancia al cuadrado entre las incrustaciones correspondan a la similitud de los rostros. El concepto anterior se lo describe mediante el cálculo de la función de pérdida triple, para ello se necesita 3 imágenes, una imagen de base (llamada ancla), la imagen real y una imagen falsa.

Figura 40

Función de pérdida triple



Nota: La figura tiene la función de triple pérdida para encontrar la distancia más corta al rostro que se encuentra en la base de datos. Tomado de *FaceNet: A Unified Embedding for Face Recognition and Clustering,* (Schroff, Kalenichenko, & Philbin, 2016)

Básicamente, se concluye que se desea que las distancias entre la incrustación de nuestra imagen ancla y las incrustaciones de nuestras imágenes real sean menores en comparación con las distancias entre la incrustación de nuestra imagen de prueba y las incrustaciones de nuestras imágenes falsas (Kumar, 2019), tal como se observa en Figura 40.

Este modelo toma imágenes RGB de 160×160 y genera una incrustación de tamaño 128 para una imagen. Para esta implementación, se necesita un par de funciones adicionales (Data Science Blogathon, 2021). Estas funciones adicionales se denomina métricas de distancia y cada una de ellas tiene una mejor precisión dependiendo el modelo aplicado junto con los datos de entrenamiento con los que fue entrenado el modelo, obteniendo excelentes resultados con el modelo de FaceNet con respecto a otros modelos de identificación facial, estos aspectos se refleja en la siguiente Tabla 7.

Tabla 7

Métricas del modelo

	Coseno	Euclidiana	Euclidiana L2
VGGFace	Theshold:0.31	Theshold: 0.47	Theshold:0.79
	Accuracy: 89.28	Accuracy: 81.42	Accuracy: 89.28
	Presicion: 97.41	Presicion: 97.82	Presicion: 97.41
FaceNet	Theshold: 0.40	Theshold: 11.26	Theshold: 0.90
	Accuracy: 98.21	Accuracy: 98.57	Accuracy: 98.21
	Presicion: 100	Presicion: 100	Presicion: 100
OpenFace	Theshold: 0.11	Theshold: 0.47	Theshold: 0.47
	Accuracy: 57.85	Accuracy: 57.85	Accuracy: 57.85
	Presicion: 95.83	Presicion: 95.83	Presicion: 95.83
DeepFace	Theshold: 0.13	Theshold: 42.21	Theshold: 0.51
	Accuracy: 54.64	Accuracy: 52.50	Accuracy: 54.64
	Presicion: 100	Presicion: 100	Presicion: 100

Nota: En esta tabla se aprecia las métricas de la distancia, la precisión (Accuracy, traducido del inglés) y el valor umbral (Treshold, traducido del inglés) de diferentes modelos para la identificación facial. Tomado de *Detección y reconocimiento de rostros capaz de vencer a los humanos mediante FaceNet (traducido del inglés, Face Detection and Recognition capable of beating humans using FaceNet):* (Data Science Blogathon, 2021).

Todas estas métricas y valores de umbral vienen incluidas en el algoritmo de FaceNet, ya depende del uso y aplicación que se requiera dar.

Arquitectura de las CNN para Facenet

Existen diversas mejoras en cuanto al modelo de Facenet, pero su base fundamental se centra en dos CNN, la arquitectura Zeiler & Fergus y el modelo de Inception Resnet de estilo perteneciente a GoogleNet (Schroff, Kalenichenko, & Philbin, 2016).

- Zeiler & Fergus: Obtiene cerca de 140 MB de medidas característica por imagen, su optimización en las capas intermedias supera de manera convincente a arquitectura de modelos de clasificación de imágenes (Zeiler & Fergus, 2013).

- Inception ResNet V1: Como conclusión de esta red se tiene que utilizar múltiples filtros de distintos tamaños, de manera simultánea, donde a diferencia de redes tradicionales se elige un filtro con una dimensión fija. Estos filtros múltiples simultáneos, son concatenados en sus resultados obteniendo un valor cercano a 7.5 MB de parámetros por cada imagen.

En Facenet se utilizan optimizaciones para ejecutarse en dispositivos móviles, en tal razón tienen menos parámetros y filtros, para que su inferencia vaya más rápido (Kunal, 2020).

Algoritmos de anti-plagio para sistemas de reconocimiento facial

Cuando se desarrolla aplicaciones en donde se emplee sistemas de reconocimiento facial, es común que se exista posibles ataques de individuos que buscan realizar delitos con imágenes o videos de algún usuario registrado, como se explicó en los sistemas biométricos, estos sistemas no son 100% fiables por este motivo se necesitan dar un grado de seguridad para disminuir esos ataques.

Por lo tanto se necesita agregar al sistema de reconocimiento facial un algoritmo de detección de cuerpos vivos para que el usuario se encuentre registrado en el sistema y además esté de cuerpo presente y no mediante una imagen, foto o un video. Este nivel de seguridad está implementado en dispositivos sofisticados con recursos computacionales muy elevados, pero que no son tan alcanzables al usuario común, su implementación es relativamente muy simple pero no tan eficiente.

Limitaciones de hardware

Muchos dispositivos móviles y cámaras ya sean integradas o IP, no poseen sensores de alta gama y esto dificulta la extracción de características faciales más robustas para dar un nivel de seguridad elevado en los sistemas de reconocimiento facial, también sumado a esto, el costo de incrementar hardware es relativamente alto. Por este motivo se busca un modelo con IA para evitar esta vulnerabilidad al sistema y así entregar confianza al momento de implementarlo.

Modelo de detección de vida silenciosa

Como se explicó anteriormente la tecnológica de detección de cuerpos vivos sirve para encontrar si el rostro que aparece frente a la cámara es real o falsa como se aprecia en la Figura 41, en este punto la tecnología de cuerpos vivos, se ha dividido en dos partes: detección cooperativa de cuerpos vivos y detección de cuerpos vivos no cooperativa (traducido del inglés Silent-Face-Anti-Spoofing).

El algoritmo de verificación en vida silenciosa ejecuta de manera directa el control corporal en vivo una vez que el usuario se encuentra frente a la cámara, a diferencia del algoritmo de detección cooperativa de cuerpos vivos, que necesita que el usuario complete una serie de normas (estas pueden ser que encuentre en un lugar y a una distancia adecuada de la cámara) y después haga la verificación (Zeusess, 2019).

Esta es una de las técnicas que utilizan algunas compañías para autentificar a la persona y que ésta se encuentre frente a la cámara físicamente y no solo por un video o foto. Adicional a la técnica anterior se tiene la tecnología de las cámaras estéreo, donde mediante dos cámara y algoritmos de profundidad, mide la distancia de los rasgos faciales del usuario adaptado al sistema facial, donde obtiene cerca de 30 mil puntos faciales y de esta forma validar que el individuo se encuentra presente, siendo esta una visión en 3D del rostro, esta tecnología es aplicada en el Face ID de los IPhone (Mainenti, 2017), (Apple, 2017).

Figura 41

Detección de un rostro no vivo.

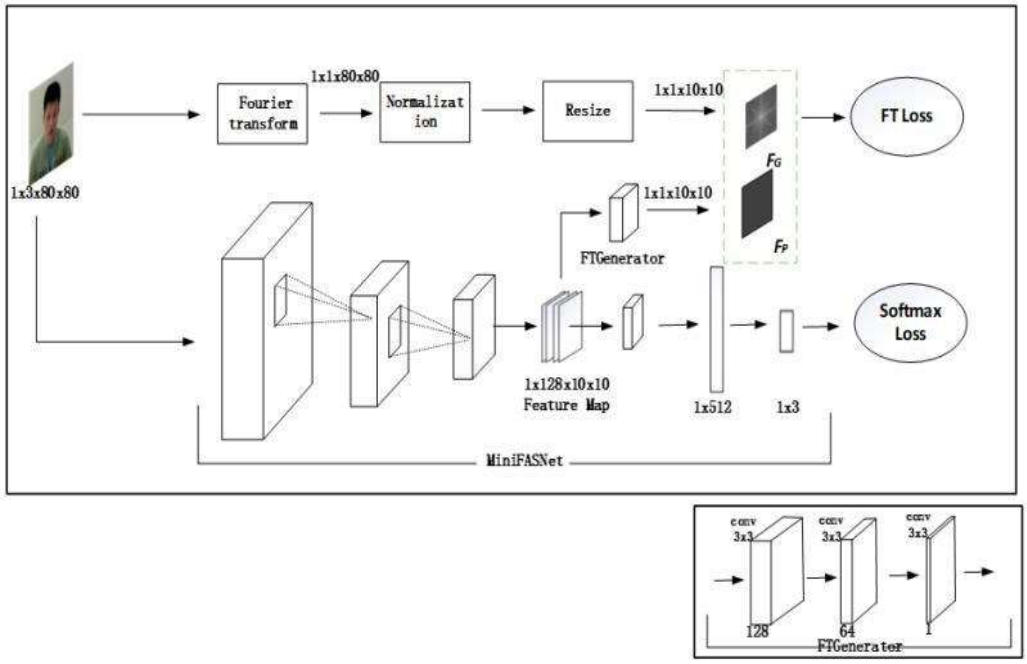
Nota: La figura proporciona la detección de un rostro dentro de una foto. Tomado de *Liveness Detection to prevent Spoofing Attack,* (iDenfy, 2021)

Funcionamiento

La arquitectura del modelo reside en la transformada de Fourier y un modelo de clasificación de redes neuronales convolucionales aplicativo en la detección de rostros, en el cual deduce la diferencia entre caras reales y caras falsas (Minivision, 2016). Se muestra su estructura general para la obtención de la predicción de si es un rostro vivo o no en la Figura 42

Figura 42

Arquitectura del modelo anti plagio



Nota: La figura muestra la arquitectura del modelo anti plagio para detectar rostros vivos. Tomado de *Silent-Face-Anti-Spoofing (Silent-Face-Anti-Spoofing,* (Minivision, 2016)

Transformada discreta de Fourier en 2D

La transformada discreta de Fourier permite tener la definición de una función matemática en otro dominio, es decir cambia del dominio del tiempo al dominio de la frecuencia, esto es debido a que muchas características se pueden analizar mejor en un dominio distinto a otro, ver Figura 43.

Figura 43

Ecuación de la transformada discreta de Fourier en 2D.

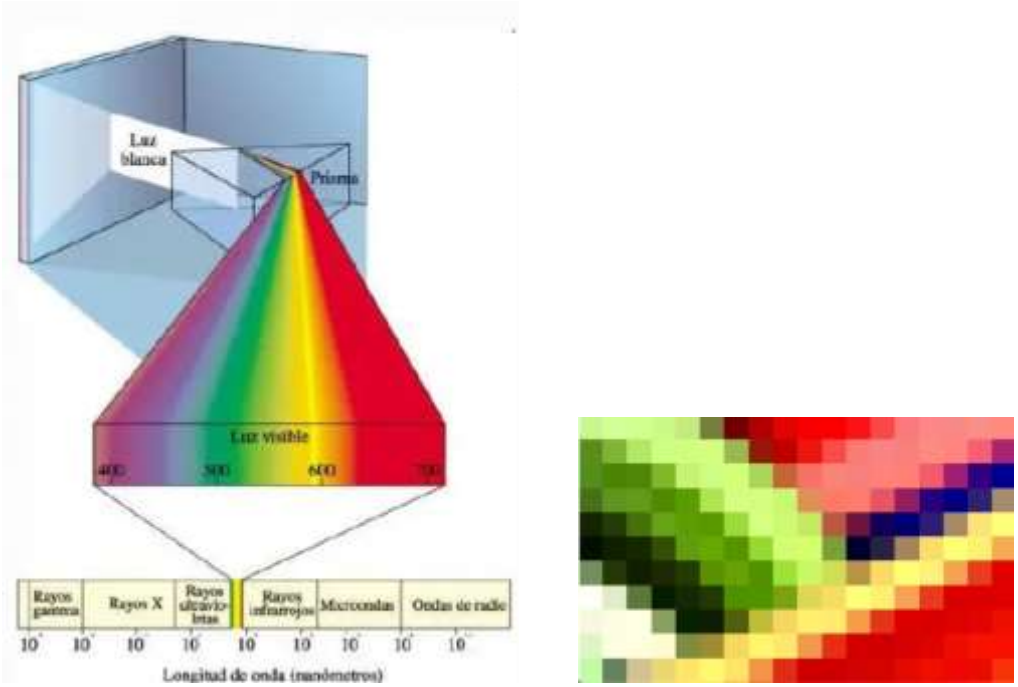
$$F(u,v) = \frac{1}{M \cdot N} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x,y) \cdot e^{-j2\pi \cdot (\frac{ux}{N} + \frac{vy}{M})}$$

Nota: La figura muestra la ecuación de la transformada discreta de Fourier en dos dimensiones.

Se necesita que la imagen para la aplicación se encuentre en el dominio del tiempo o del píxel, esta imagen deberá ser acotada o finita, esto se lo realiza a través de la medida de píxeles en columnas y filas, este paso se logra con la transformada discreta de Fourier en 2D. Esta transformada discreta de Fourier en 2D, transformará la función de la imagen $f(x,y)$ en función del pixel en columnas *Y*, y en filas en *X*, ver Figura 44 de la derecha hacia una función en el dominio de la frecuencia, esto entrega como resultado una versión de la imagen en distintas escalas espaciales, esta imagen contiene tanto bajas y altas frecuencias que al sumarse o superponerse dan como resultados la representación en ondas o frecuencia de la imagen, ver Figura 44 de la izquierda.

Figura 44

Izquierda: Rango de los colores RGB en ondas de frecuencia, Derecha: Imagen en píxeles



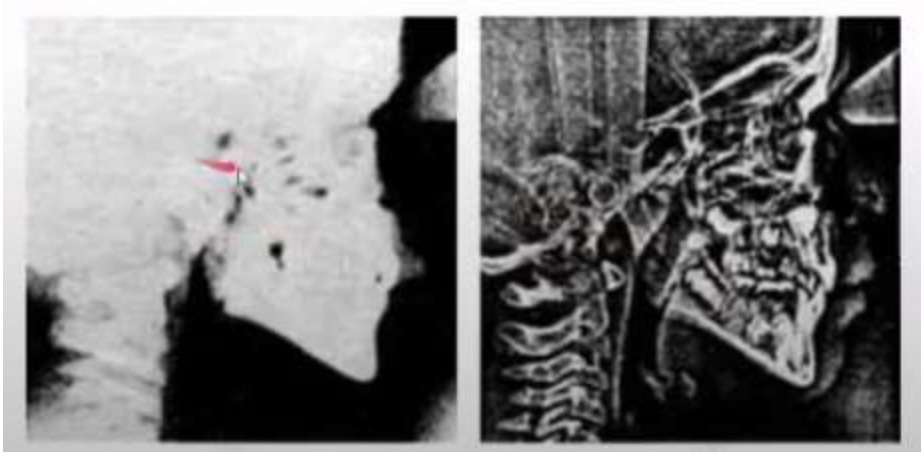
Nota: La figura muestra el rango de frecuencias de los colores primarios RGB y una imagen pixelada. Tomado de *métodos cuantitativos de colorimetría, reflectancia,* (Cerron Zarcco , 2015)

Cuando hablamos del pixel, se tiene que mencionar que este representa colores, estos se verán influenciados por el rango de la onda que llega desde la luz visible, esta luz al entrar en contacto con un medio como lo es la atmósfera, se verá refractada, cambiando la velocidad y dirección en términos de la incidencia de pasar de un medio a otro, esto como resultado entrega que ciertas tipos de ondas se refractan más rápido y otras más lento, dividiendo las ondas en ondas de rango corto y largo. Entonces permite mediante la aplicación del sistema de colores RGB, el cual es una síntesis aditiva de tres colores primarios como son el rojo, azul y verde, obtener la representación mediante una suma de colores del color final, es decir se sumará desde el 0 al 255, cada color de los primarios en distinta proporción dando como resultado el color final. Este color representado puede ser de onda corta o larga, y puede estar en el rango de los colores

primarios, y estos se verán refractados en un índice de refracción, tal como se observa en la siguiente **Figura 47**Figura 45.

Figura 45

Izquierda: Imagen en el dominio del tiempo, Derecha: Imagen en el dominio de la frecuencia



Nota: La figura muestra a la izquierda las características en el dominio del tiempo o píxel, en la derecha se tiene un aumento de características en el dominio de la frecuencia. Tomado de *Diseño, acondicionamiento y tratamiento de imágenes,* (Aguirre Balcázar & Bautista Toapanta, 2016)

Como se observa en la Figura 45, una vez representada la imagen en el dominio de la frecuencia, se aplica una serie de filtros y de esta manera se logra eliminar parte de la onda que se tenía inicialmente y de esta forma enfocarse en ciertos rasgos que se destaquen de la imagen, una vez obtenido esta característica, se necesita regresar del dominio de la frecuencia al dominio del tiempo o del pixel por medio de la transformada inversa de Fourier.

En conclusión, usando este método de la DFT en 2D y juntando con arquitecturas de CNN, este aspecto se observa en la Figura 42 en el rostro, se pretende encontrar un patrón de frecuencias dentro de un estándar que se refractan más o menos en ondas cortas o largas y delimitando en un rango por medio de filtros ya sean pasa-bajo y pasa-

altos, donde este parámetro se conoce índice de refracción. Todas estas diversas frecuencias ya filtradas se acumulan en un solo modelo que al juntar con la detección de rostro de la CNN, este indicará a su salida si se trata de un rostro real o falso. Este modelo desarrollado por en compañía de Minivision redujo la cantidad de parámetros mejorando el rendimiento y obteniendo baja pérdida de precisión (0.991 millones a 0.435 millones).

A su vez en la investigación muestra dos modelos de detección de vida, donde el algoritmo de ámbito de pago presenta una precisión del 99.7% y el de uso libre del 97.8%, suficiente para el desarrollo de la investigación.

Capítulo III

Guía de diseño del sistema de registro facial

Interfaces de usuario

Una interfaz de usuario, es un conjunto que abarca las distintas estructuras de control y caminos por el cual el usuario puede entablar comunicación con una máquina, computador o sistema.

Una interfaz tiene dos características esenciales los cuales son:

- Elevado grado de uso
- Ser intuitiva y amigable para cualquier usuario.

Entendido este concepto, podemos agregar un concepto, sobre una interfaz gráfica de usuario (GUI), se puede apreciar en la Figura 46, donde se concluye que es el contenido gráfico, a través del cual se visualiza información del proceso del sistema en una pantalla (Albornoz, Berón, & Montejano, 2017).

Figura 46

Interfaz gráfica de usuario

Nota: La figura muestra la interfaz gráfica de usuario.

Tipos de interfaz

Para el desarrollo de interfaces gráficas, existen 3 tipos según el propósito y diseño:

- Del tipo hardware: Encasilla un conjunto de elementos que dan la facilidad de ingresar, procesar datos con ratones, teclados y pantallas.
- Del tipo Software: Permite observar información sobre herramientas de control y procesos los cuales al usuario de manera fácil e intuitiva visualiza en una pantalla o dispositivo móvil.
- Mixto: Permite crear un vínculo entre el individuo y la máquina para así comprender las instrucciones que este ingresa traduciendo estos datos a lenguaje de máquina.

Interfaz web

Para un correcto diseño de una interfaz web se debe considerar los siguientes aspectos como, ¿Por qué?, ¿Para quién?, ¿dónde y con qué estructura?

Estos aspectos, sumado a las características esenciales deben brindar una interfaz web que en conclusión permita que el usuario se olvide que está interactuando con una máquina (Fernández Ruíz , Angós Ullate, & Salvador Oliván, 2006).

Guía de desarrollo de interfaz web

HTML5

Denominado como lenguaje marcado de hipertexto quinta versión, (Hypertext Markup, Language por sus siglas en inglés), es la quinta versión de un estándar para el desarrollo de interfaces web, páginas web, su estructura y contenido, Figura 47.

Figura 47

Logo HTML5

Nota: La imagen muestra el logo de HTML5. Tomado de *Developer-Mozilla: HTML5,* (Developer Mozilla, 2021)

CSS3

Hoja de estilos en cascada versión 3 (Cascading Style Sheets por sus siglas en inglés), es un tipo de lenguaje que define los estilos y presentación proporcionando a la interfaz web el atractivo visual.

Figura 48

Logo CSS3

Nota: La imagen muestra el logo de CSS3. Tomado de *Rolando Caldas: CSS3 básico,* (Rolando Caldas, 2021)

JavaScript

Se lo define como un lenguaje de programa secuencial, ligero el cual vincula la actualización de contenidos de forma dinámica, además orientado a objetos, es compilado justo a tiempo par de la hoja HTML5 y CSS3, muy usado en el desarrollo del contenido dinámico de la interfaz y páginas web (Developer Mozilla, 2021), ver Figura 49.

Figura 49

Logo HTML5

Nota: La imagen muestra el logo de JavaScript. Tomado de *Chris Williams, Logo JavaScript,* (Williams, 2011)

Flask

Denominado como micro-framework, escrito para el lenguaje de Python, diseñado para el desarrollo de aplicaciones web (Domingo Muñoz, 2021). Cuando se menciona la palabra micro no quiere decir que es un marco de desarrollo pequeño, o que tiene limitantes al momento de crear una interfaz web, sino que Flask toma las herramientas necesarias para la creación de una aplicación web pero si se necesita variantes adicionales (extensiones) se podrían añadir a Flask para darle mejor funcionalidad sin ningún problema, ver Figura 50.

Figura 50

Logo Flask

Nota: La imagen muestra el logo de Flask. Tomado de *Formadores IT: Flask,* (Formadores IT, 2021)

Ventajas de uso

- Para la creación de una App básica, rápida y versátil, que no demande tantas extensiones es más que suficiente.

- Proporciona un servidor web de desarrollo para probar las aplicaciones y ver los resultados obtenidos.
- Flask es de uso libre para aplicaciones sencillas, está amparado en la licencia BSD[2].

Desventajas

- No es recomendado para implementar en producción.

Retransmisión de video en directo

La retransmisión de video en directo (Streaming por sus siglas en inglés), es la tecnología que internet proporciona para la transmisión de video y audio por medio de la red todo esto se realiza sin la necesidad de descargar y guardar el video a una estación local (González & Cajamarca, 2014).

Formas de transmisión y protocolos comunicación de video

En cuanto a la transmisión del video por la red se refiere, existen 3 formas que son las siguientes:

- En directo: El video y audio transmitido son digitalizados en transmitidos en tiempo real a la red (EPN, 2021). Los usuarios pueden ver el evento o suceso al mismo tiempo que está ocurriendo.
- En diferido: El video y audio son grabados para ser mostrados al usuario cuando el evento está por concluir o término, No es en tiempo real.
- Bajo demanda: Los usuarios solicitan al proveedor que les autorice para poder observar el evento por medio de la web en cualquier momento, esta transmisión es almacenada en un servidor de streaming.

En cuanto a los protocolos de comunicación para el video tenemos los siguientes:

- TCP: Denominado protocolo de control de transmisión, permite la comunicación entre dispositivos verificando la transmisión de información entre el dispositivo

[2] Licencia BSD: Licencia de software libre tipo permisiva.

emisor y receptor, es orientado a conexión, debido a este concepto no es tan recomendado para transmitir video en vivo.

- UDP: Llamado protocolo de datagramas de usuario, permite la comunicación por medio de datagramas, sin esperar una sincronización entre el emisor y receptor, es recomendado para la transmisión de vídeo en vivo.
- WLAN: Es una comunicación inalámbrica, que usa ondas de radio, para comunicarse con dispositivos que tengan en su hardware esta tecnología. Por los estándares actuales como IEEE 802.11b y IEEE 802.11g, que admiten velocidades de 11Mbps a 2.4Ghz Y 54Mbps a 5Ghz respectivamente, la transmisión de video en vivo para circuitos cerrados de cámaras de video en vivo es recomendable su uso.
- RSTP: Denominado como protocolo de transmisión en tiempo real, es un protocolo no orientado a conexión que entrega una comunicación de uno o varios flujos sincronizados de datos, pueden ser de video o audio. Permite además controlar a distancia el video o acceso a un circuito cerrado de cámaras en vivo. Recomendado en la implementación de aplicaciones profesionales.

Formatos de codificación y compresión de video

Para que el video en una transmisión de video existe además de los protocolos mencionados previamente formatos de compresión de video que permiten una visualización en tiempo real y se muestra a continuación:

- MPEG-4: Existen varios estándares de uso pero en si es utilizado para circuitos cerrados de cámaras con un ancho de banda reducido y que necesiten enviar imágenes de alta calidad, limita el ancho de banda y la frecuencia de imagen.
- MP4: Compatible con muchas plataformas como YouTube y dispositivos móviles, es un formato utilizado por el usuario donde preserva la calidad de la imagen aun después de la compresión.
- H.264: Es una forma de codificación de vídeo avanzada, partiendo del criterio de MPEG-4, su funcionamiento se basa en no comprometer la calidad de la imagen

reduciéndolo hasta en un 50% con lo cual requiere menos ancho de banda y espacio de funcionamiento para circuitos cerrados de cámaras, muy recomendado para liberar el ancho de banda de la red en la transmisión de video.

- H.265+: Es una mejora de su antecesor, el formato de compresión de video H.264 de hasta el 80% en ancho de banda y espacio requerido en la transmisión de video de circuito cerrado de cámaras (HikVision, 2021)

OpenCV

Es una librería multiplataforma de uso libre muy utilizada en la visión artificial para tratar y analizar imágenes, ver Figura 51.

Figura 51

Logo OpenCV

Nota: La imagen muestra el logo de OpenCv. Tomado de *OpenCv: Web Official,* (OpenCV-Web Official, 2021)

Al ser multiplataforma permite trabajar con diferentes lenguajes de programación y sistemas operativos. Para el desarrollo de este proyecto de investigación se usa el lenguaje Python así como el sistema operativo Ubuntu 18.04 LTS, que además son de libre uso, proporcionan muchas librerías e información que permiten diseñar un sistema de identificación facial robusto.

Figura 52

Logo Python

Nota: La imagen muestra el logo de Python. Tomado de *Python: Web Oficial,* (Python, 2021)

En el siguiente capítulo conforme vaya avanzando se irá detallando cada uno de las diversas tecnologías usadas en el desarrollo del presente proyecto de investigación.

Capítulo IV

Diseño e implementación del sistema control de acceso por reconocimiento facial

Descripción

Para este capítulo se explicará cómo se encuentra estructurado y diseñado el sistema de control de acceso por medio de identificación facial con técnicas de aprendizaje profundo, visión por computador e inteligencia artificial.

El sistema cuenta con cuatro fases: algoritmos de detección, reconocimiento facial, módulo de autentificación con el algoritmo de vida y finalmente registro del personal todo esto unido a la interfaz web.

La primera fase inicia con la implementación de los algoritmos de aprendizaje profundo de detección facial así como el ajuste de los rostros del personal asociado al sistema con el algoritmo MTCNN y detección de rostro en tiempo real con el modelo RetinaFace. La segunda fase inicia con la aplicación del modelo para la detección de los rasgos faciales con MediaPipe unido a un modelo de aprendizaje profundo para identificación facial y así reconocer a los individuos ingresados en el sistema. La tercera fase es una validación de seguridad para detección de rostros reales o falsos, proporcionando robustez al sistema por medio de un algoritmo llamado anti-plagio o de vida. En la última fase se concentra en el diseño de la interfaz web y la comunicación con los algoritmos y archivo de tipo xlsx[3] que contiene la información del personal asociado al sistema de identificación facial, donde toma la información biométrica de los usuarios y si ha sido identificado procede a visualizar estos datos en la interfaz web en tiempo real.

De acuerdo como se avance en las fases de desarrollo se analizará y se explicará el diseño de cada módulo, detallando cómo se encuentra implementado, los recursos y algoritmos asociados al módulo. Tal como se muestra en el diagrama de bloques, Figura 53, del sistema de control de identificación facial.

[3]Xlsx: Formato de archivos para una hoja de cálculo

Figura 53

Diagrama de bloques del sistema de identificación facial

Nota: La imagen muestra el sistema global de identificación facial.

Adaptación del sistema

Para el desarrollo del sistema se han instalado varios recursos de hardware y software que se detallan a continuación en las Tabla 8 y Tabla 9.

Tabla 8

Características técnicas del software

Software	Características
S.O Linux	Distribución Ubuntu 18.04 LTS
Python	Versión 2.7 y 3.6
OpenCV	Librería para Python versión
Numpy	Librería de cálculo de matrices versión

Nota: Esta tabla muestra, un resumen del software que se utilizó en el desarrollo del sistema de identificación facial.

Tabla 9

Características técnicas del hardware

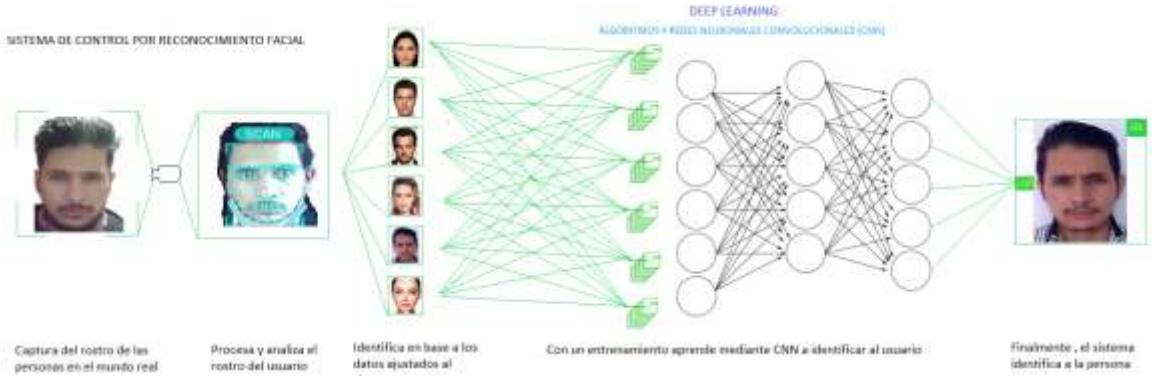
Hardware	Características
PC	HP, Pavilion Gaming Laptop
Procesador	Intel Core I5 de 8va Generación, 3.5 Ghz
GPU	Nvidia GTX-1050ti-4GB VRAM
RAM	8 Gb
Tarjeta de red	Ethernet 100/1000 Mbps-wifi

Nota: Esta tabla muestra, las características del hardware que se utilizó en el desarrollo del sistema de identificación facial.

De forma general, en la Figura 54, muestra como la imagen del usuario ingresa a la CNN, obteniendo la predicción del rostro más cercano a la coincidencia de todos los rostros ajustados al sistema para finalmente entregar los datos del rostro identificado.

Figura 54

Estructura de la red neuronal del sistema de identificación facial.



Nota: En la imagen se observa la estructura interna de la red neuronal convolucional del sistema de identificación facial.

Acceso y adquisición de imágenes

Para obtener los objetivos de desarrollo del proyecto se lo realiza con la cámara IP, los detalles de la cámara se muestran en la siguiente Tabla 10

Tabla 10

Características técnicas de la cámara

Hardware	Características
Cámara IP, HikVision	Tipo domo, resolución 960P color, 2 megapíxeles, 1920x1080, 45FPS, codificación H.264, +H.265

Nota: Esta tabla muestra, un resumen de las características técnicas de la cámara IP de la marca HikVision.

Para obtener una grabación de video se debe mantener una lectura de fotogramas constante por tal razón el objeto creado se encuentra contenido dentro de una estructura repetitiva (Ibarra Flores, 2020). Para ingresar al video se utiliza el programa VLC, el comando ingresado funciona para las cámaras de la marca HikVision, para cámaras de otras empresas se debe buscar el comando apropiado, se detalla en la siguiente Tabla 11:

Tabla 11

Estructura del comando para acceder a la cámara IP

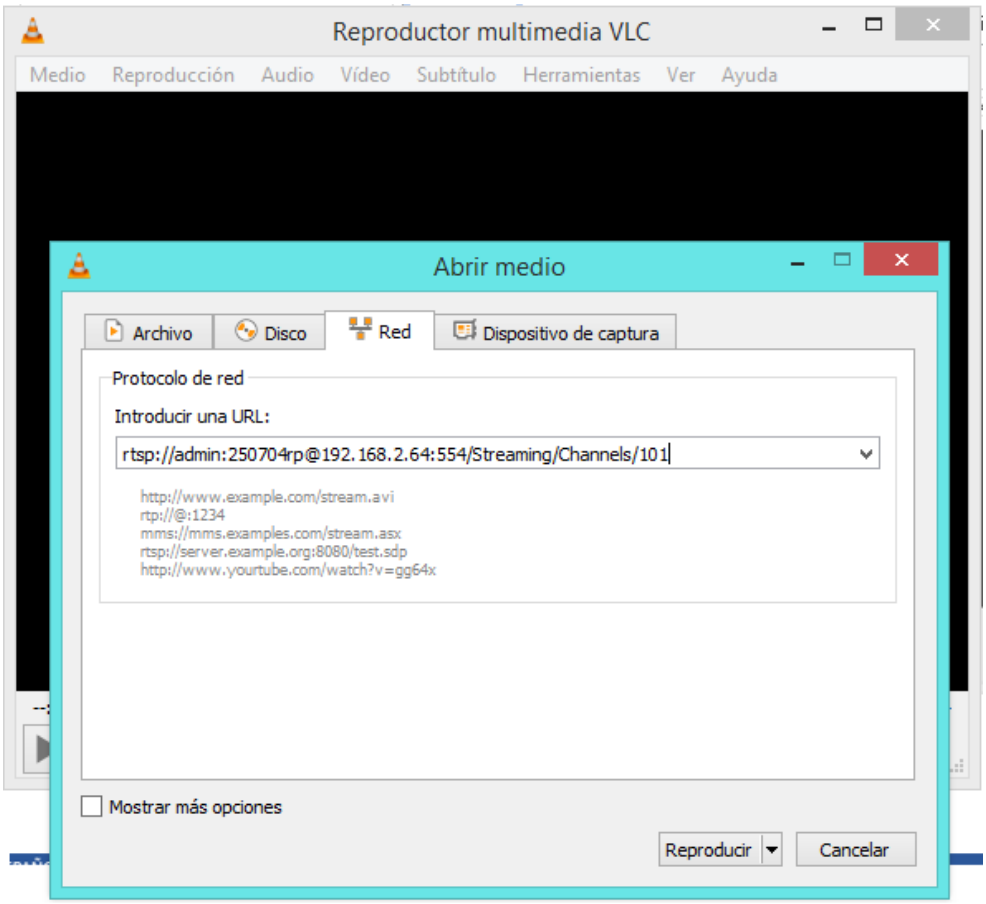
Comando	
rtsp://usuario:contraseña@IP:puerto/Streaming/Channels/101	
RSTP:	protocolo de transmisión de datos
Usuario:	nombre de la red asociada a la cámara IP
Contraseña:	clave de la red asociada a la cámara IP
IP	IP de la cámara
puerto:	puerto de transmisión de los datos
Streaming:	Asignación de video en vivo
Channels**:**	101, si la cámara tiene un solo canal de transmisión de video

Nota: Esta tabla muestra, el comando para acceder al video de la cámara IP de la marca HikVision.

Se abre el programa de VLC, y se presiona el comando de acceso rápido *Ctrl+N*, para medios de transmisión en streaming, una vez ahí se ingresa el comando con los datos correspondientes que se muestra en la tabla, ver Figura 55.

Figura 55

Programa VLC



Nota: En la figura muestra el ingreso del comando para obtener el video en vivo de la cámara IP.

De forma inmediata se muestra el área de video en vivo de la cámara, tal como se muestra en la Figura 56.

Figura 56

Video en vivo

Nota: En la figura muestra el video en vivo de la cámara IP a través del programa VLC.

En cuanto a velocidad de detección se refiere la cámara IP, tiene un canal de transmisión por medio de un cable y esto retrasa un poco la señal, pero con la aplicación de los formatos de compresión de vídeo, disminuye ese retraso con el cual permite desarrollar el presente proyecto de investigación. A continuación se detalla las librerías utilizadas en la siguiente Tabla 12:

Tabla 12

Parámetros de preprocesamiento de imágenes

Librería	Función
OpenCV	Cumple la función de tratar, y pre-procesar las imágenes para adaptarlas al sistema
Gstreaming	Librería de OpenCV para la transmisión de video en vivo sin cortes y retrasos prolongados

Nota: En la tabla se muestran las funciones para pre-procesar el video de la cámara IP.

Detección facial

Descripción

Para el presente proyecto de investigación se han tomado dos modelos de aprendizaje profundo, donde el primero ajusta las fotos de los usuarios que ingresaran al sistema de identificación facial mientras que el segundo modelo infiere y detecta el rostro en vivo. Se ha dividido en dos partes la detección facial ya que varios modelos de detección facial si bien son muy robustos en cuanto a esta aplicación, son casi imposibles de implementar para fines de investigación por inconvenientes de hardware y además entregan resultados inadecuados al momento de implementar en tiempo real. En la siguiente Tabla 13 se observa las librerías y funciones utilizadas.

Tabla 13

Descripción de los modelos empleados

Librería	Función
MTCNN	Red neuronal convolucional multitarea en cascada, Pre-entrenada, permite ajustar y alinear de forma general los rostros al sistema
AntiSpoofPredict	Esta librería contiene el modelo de RetinaFace que permite detectar el rostro y mostrarlo en pantalla en tiempo real
Pickle	Permite guardar todos los rostros detectados en un archivo base.
Keras	Biblioteca de para redes neuronales en Python, para ajustar y guardar los usuarios

Nota: En tabla se muestra un resumen de los modelos aplicados al sistema de reconocimiento facial-

El primer modelo MTCNN, toma los rostros de los individuos y los alinea si es necesario para generalizar la información biométrica de los usuarios ajustando al sistema, una vez realizado este paso con RetinaFace infiere y detecta el rostro en tiempo real. Luego de esto guarda la información de los usuarios en un archivo serializable de tipo Pickle junto con keras.

Ajuste de los usuarios al sistema con MTCNN

Entonces para el primer paso de ajuste de los usuarios al sistema se toma el modelo de aprendizaje profundo llamado MTCNN. Este modelo como se explicó en el capítulo 2, presenta algunas ventajas además de detectar el rostro, una de ellas que permite la alineación del rostro siendo esta condición al momento de ajustar al sistema primordial ya que generaliza los rostros a una postura uniforme y centrada siendo este aspecto fundamental para encontrar los rasgos faciales e identificar al usuario de manera más efectiva y precisa.

Figura 57

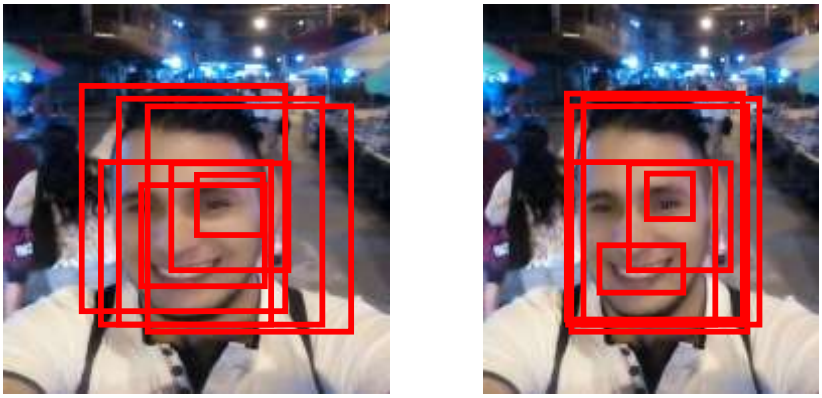
Imagen aplicando pirámide de imágenes

Nota: Adaptado de "*Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks*", (Zhang, Zhang, Li, & Qiao, 2016)

Su funcionamiento empieza con tres pasos, inicia con la toma de imágenes y escalando mediante la técnica de pirámide de imágenes a distintos tamaños, Figura 57 esta será la entrada a la primera parte de la red llamada de propuesta rápida (P-Net) .Esta red propuesta nos entrega los posibles rostros y su respectiva caja delimitadora.

Figura 58

Localización de las regiones de interés (ROI)

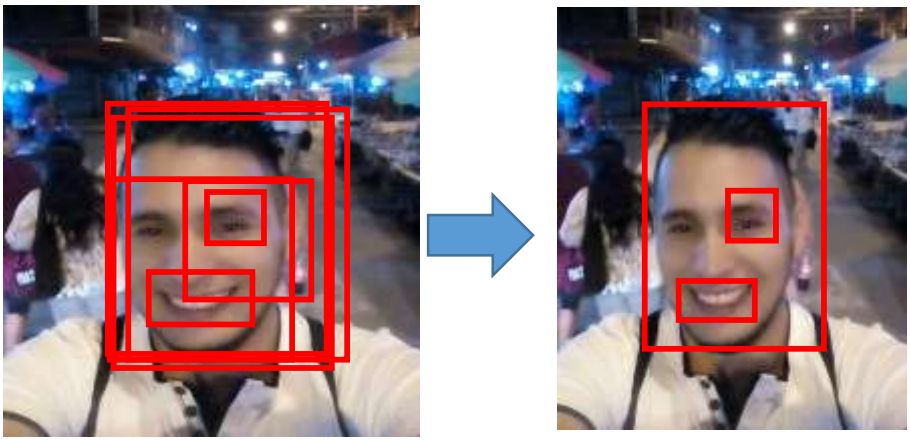


Nota: Adaptado de "*Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks*", (Zhang, Zhang, Li, & Qiao, 2016)

La siguiente etapa toma la salida de la etapa anterior como entrada a una CNN llamada Red de redefinición (R-Net), en la cual minimiza el número de posibles candidatos, ajusta las cajas delimitadoras a los posibles rostros encontrados, obteniendo una respuesta a la salida si es o no un rostro, ver Figura 58.

Figura 59

Búsqueda de rasgos faciales



Nota: Adaptado de "*Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks*", (Zhang, Zhang, Li, & Qiao, 2016)

Finalmente en la tercera etapa, la red de salida (O-Net), tiene como salida determinar las posiciones de los rasgos faciales importantes como son los ojos, nariz, boca de los datos ingresados, ver Figura 59.

Figura 60

Rostro detectado mediante MTCNN

Nota: Adaptado de "*Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks*", (Zhang, Zhang, Li, & Qiao, 2016)

Una vez realizado estos pasos, se ha determinado que efectivamente es un rostro y además su caja delimitadora, ver Figura 60

Alineación del rostro

Luego de haber obtenido el rostro lo que prosigue es verificar si la cara está alineado o no a una postura para generalizar a todos los rostros que ingresen al sistema y de esta manera la identificación facial no tenga inconveniente en deducir de manera eficaz y precisa.

Figura 61

Rostro con postura inclinada

Para alinear los rostros se toma en consideración la distancia euclidiana y los puntos de referencia facial previamente encontrados, se detectan la ubicación exacta de los ojos dentro del rostro, ver Figura 62

Figura 62

Izquierda: Fórmula de la distancia euclidiana. Derecha: detección del rostro

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (y_1 - x_i)^2}$$

Nota: La imagen muestra la fórmula de la distancia euclidiana que se aplicará para buscar alinear el rostro de la imagen de la derecha.

Ahora lo que procede es determinar cuál es el ojo izquierdo y derecho, para aplicar los cálculos de la distancia euclidiana, tal como se muestra en Figura 63.

Figura 63

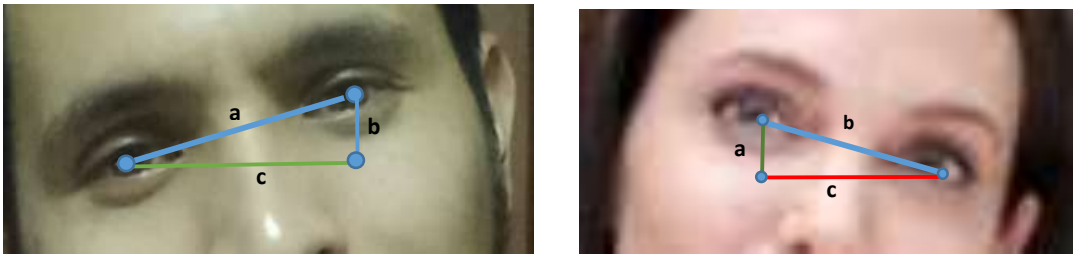
Detección de ojos en el rostro

Nota: La imagen muestra la detección de ojos como rasgo facial para buscar aplicar la alineación del rostro.

Los ojos forman una distancia que se puede calcular con la fórmula de la distancia euclidiana, como se aprecia en la Figura 64, pueden tener dos sentidos ya sea horario o anti-horario, pero se tiene el mismo fin, que es alinear el rostro a una postura general.

Figura 64

Izquierda: Inclinación del rostro sentido antihorario, derecha: inclinación del rostro sentido horario

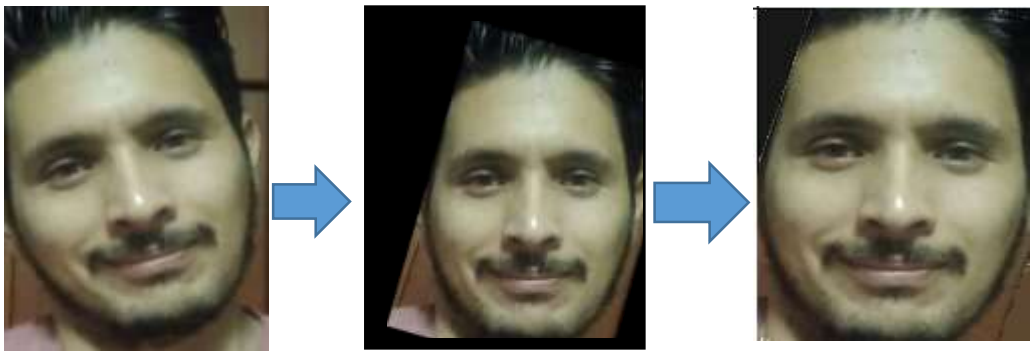


El paso anterior busca determinar el ángulo de rotación que permitirá rotar el rostro en sentido horario o anti horario a una postura recta, generalizando la foto a todos los usuarios que ingresen al sistema de reconocimiento facial.

Ya obtenidos estos datos, se procede a guardar los resultados de la información biométrica de cada usuario en un archivo de tipo Pickle que proporciona el lenguaje de Python.

Figura 65

Secuencia de alineación de rostros



Esta secuencia de ajuste de los usuarios se lo detalla con más detenimiento en la interfaz web.

RetinaFace

Es un modelo pre-entrenado de aprendizaje profundo proporcionado por Caffe para la detección de rostros donde también posee las características de alineación y corrección de postura del rostro. Se agrega este modelo ya que es mucho más ligero y fácil de implementar en tiempo real esto se debe a que no requiere tantos recursos computacionales, además una razón de peso, es debido que el algoritmo de vida el cual se explicará más adelante viene adaptado junto con el modelo de RetinaFace, facilitando su adaptación.

Detección del rostro con RetinaFace

Para visualizar la detección del rostro, se usa el modelo pre-entrenado estudiado previamente en el capítulo 2, con esto permitirá obtener la caja delimitadora del rostro (traducido del inglés, bounding boxes), y en efecto detectar el rostro. En la Figura 66, se muestra el proceso de detección del rostro con modelo de RetinaFace y a su vez la detección en tiempo real.

Figura 66

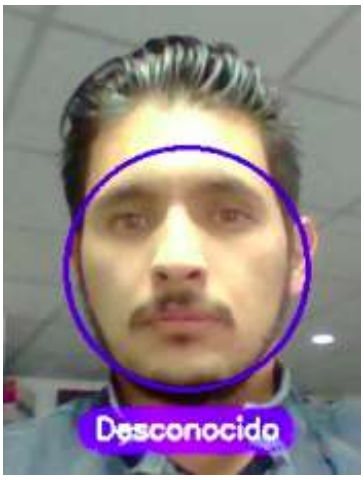
Diagrama de flujo para la detección del rostro

Nota: En la imagen se muestra el diagrama de flujo que utiliza el modelo de RetinaFace para predecir el rostro.

Una vez aplicado el diagrama de flujo de la figura anterior, se obtiene la siguiente Figura 67, que es la predicción de rostro en tiempo real con el modelo RetinaFace, por diseño se escogió dibujar un círculo en lugar de un cuadro delimitador en el rostro, siendo esta característica única al diseño del presente proyecto de investigación.

Figura 67

Detección del rostro



Nota: En la figura se muestra la detección en tiempo real del rostro con el modelo de RetinaFace

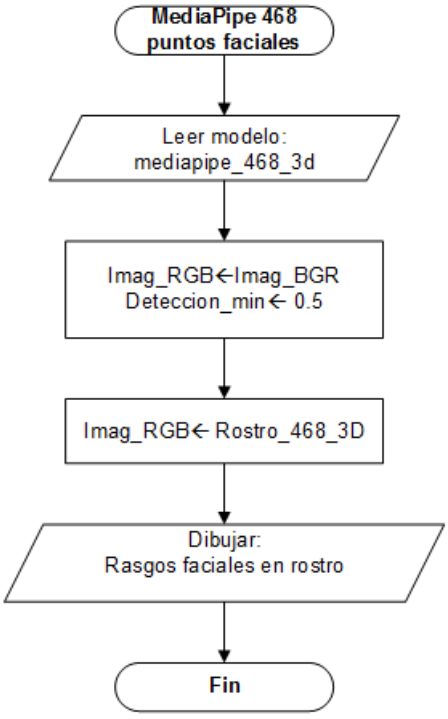
Módulo de extracción de rasgos e identificación facial

Extracción de rasgos faciales por Face-Mesh de MediaPipe

Luego de obtener el rostro, queda llamar al modelo de face mesh de 468 puntos faciales ya entrenado en 3D de Media Pipe tal como se muestra en el diagrama de la Figura 68.

Figura 68

Diagrama de flujo de Face_Mesh 468.



Nota: En la imagen se muestra el diagrama de componentes que utiliza el modelo de FaceMesh 468 de MediaPipe para obtener los rasgos.

Se observa en la Figura 69, la respuesta del diagrama de flujo con los 468 puntos faciales en 3D en el rostro del individuo.

Figura 69

Detección de 468 rasgos faciales con MediaPipe

Reconocimiento facial

Una vez ajustados los rostros del usuario y detectados con MTCNN y RetinaFace respectivamente, además de haber extraído los puntos faciales en un modelo que predecirá en el sistema, ahora continúa la identificación del individuo al cual pertenece el rostro ajustado al sistema donde se incluye también mostrar la información del usuario en la interfaz web, para lo cual se utiliza FaceNet.

Se detalla además a continuación las librerías que se utilizó para la identificación facial mediante la siguiente Tabla 14:

Tabla 14

Síntesis del modelo de identificación facial

Librería	Función
TensorFlow	Biblioteca de aprendizaje automático y profundo, facilita la importación del modelo facenet.h5 para identificación facial

FaceNet

Para la identificación facial se usó FaceNet del marco de desarrollo de TensorFlow, el cual se detalla en la siguiente Tabla 15.

Tabla 15

Parámetros del modelo de reconocimiento facial

Modelo	Arquitectura	Conjunto de datos entrenamiento	Precisión LFW[4]
FaceNet-TensorFlow	Inception ResNet	VGGFace	99.65%

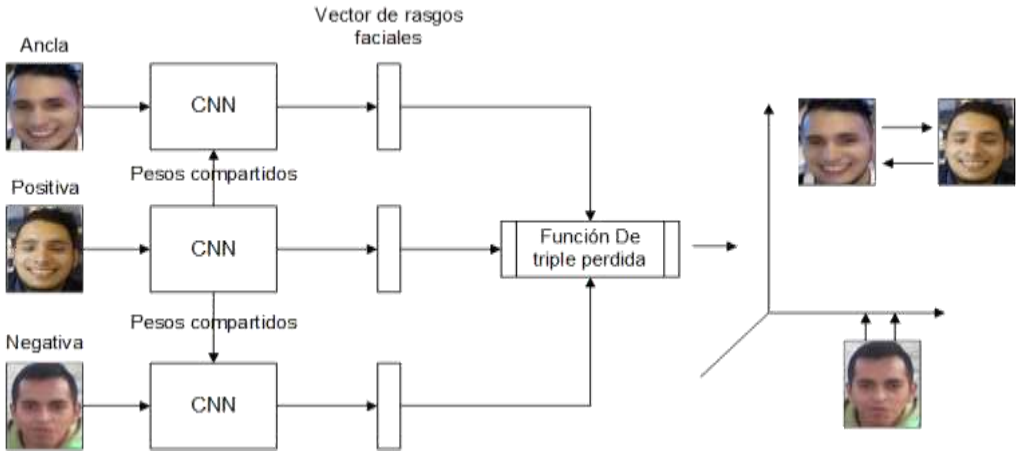
Nota: La tabla muestra un resumen de los parámetros que presenta el modelo Facenet, tomado de: *FaceNet: A Unified Embedding for Face Recognition and Clustering*, (Schroff, Kalenichenko, & Philbin, 2016)

Tal como se explica en la Figura 70, la idea de FaceNet, una vez obtenido el vector de rasgos faciales o incrustaciones de los rostros, se aplica la función de pérdida triple donde acerca dos imágenes de la misma persona y aleja los vectores que no corresponden a la misma imagen, la modificación de estos pesos compartidos se lo realiza a través de un valor de distancia guía que tiene en su estructura para agrupar vectores similares. En conclusión, si a dos vectores descriptores de caras se le aplica una métrica de distancia (distancia euclidiana, función coseno, tal como se explica en el capítulo 2 y Tabla 7) y entre ellos existe una distancia inferior o valor umbral de 0.4 (métrica de distancia del coseno), se puede considerar que pertenecen a la misma persona (Ibarra Flores, 2020). De lo contrario, son de diferentes personas.

[4] LFW: Conjunto de datos denominado, caras etiquetadas en la naturaleza, traducido del Inglés

Figura 70

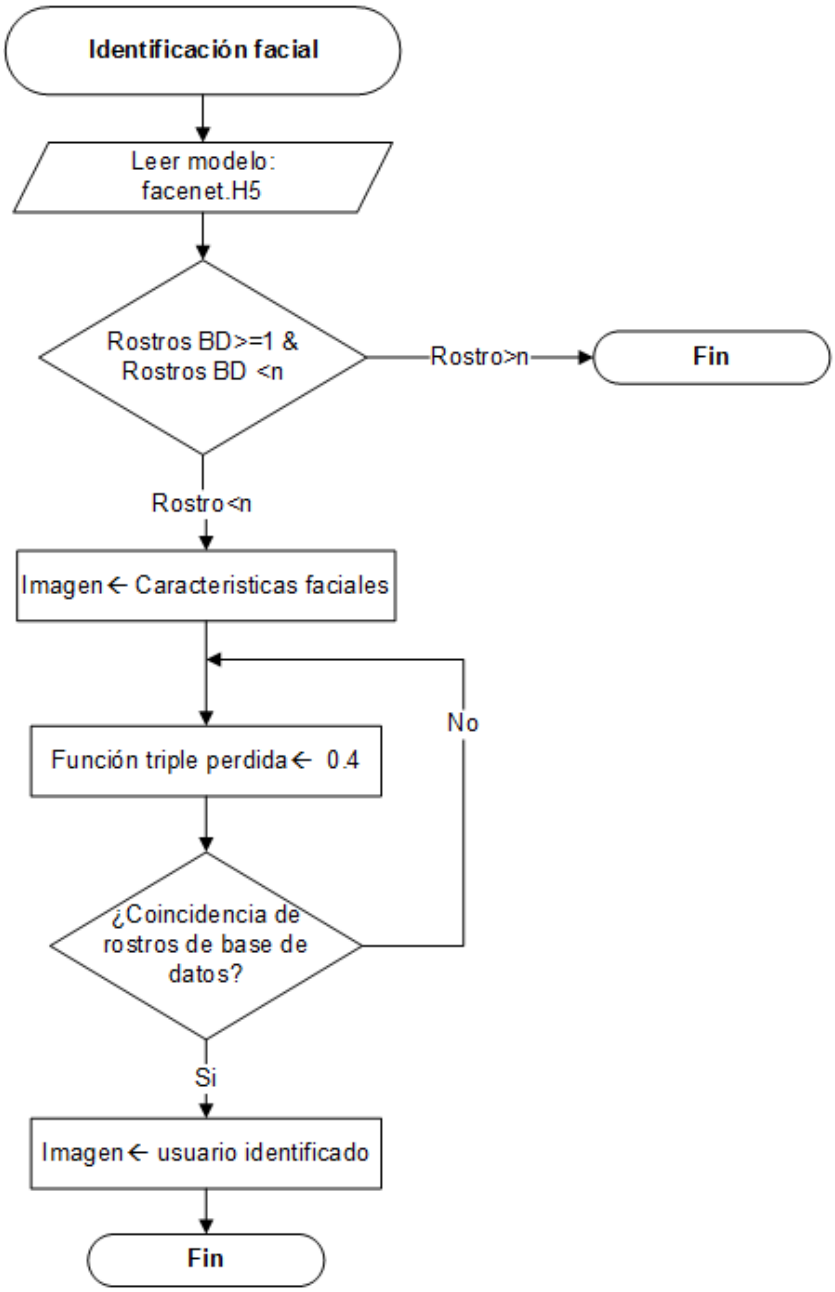
Función de triple pérdida aplicada al sistema de identificación facial



En el diagrama de flujo de la Figura 71 se puede apreciar la función de identificación facial, donde carga el modelo de FaceNet, y empieza analizar los rostros de la carpeta con los usuarios registrados, donde se encuentra coincidencia extrae los datos biométricos y unido la función de 468 puntos faciales de Media Pipe, identifica al usuario registrado en el sistema.

Figura 71

Diagrama de flujo identificación facial.



Una vez aplicado se obtiene la predicción del rostro en tiempo real, tal como se muestra en la Figura 72.

Figura 72

Identificación Facial unida a 468 puntos faciales de Media Pipe

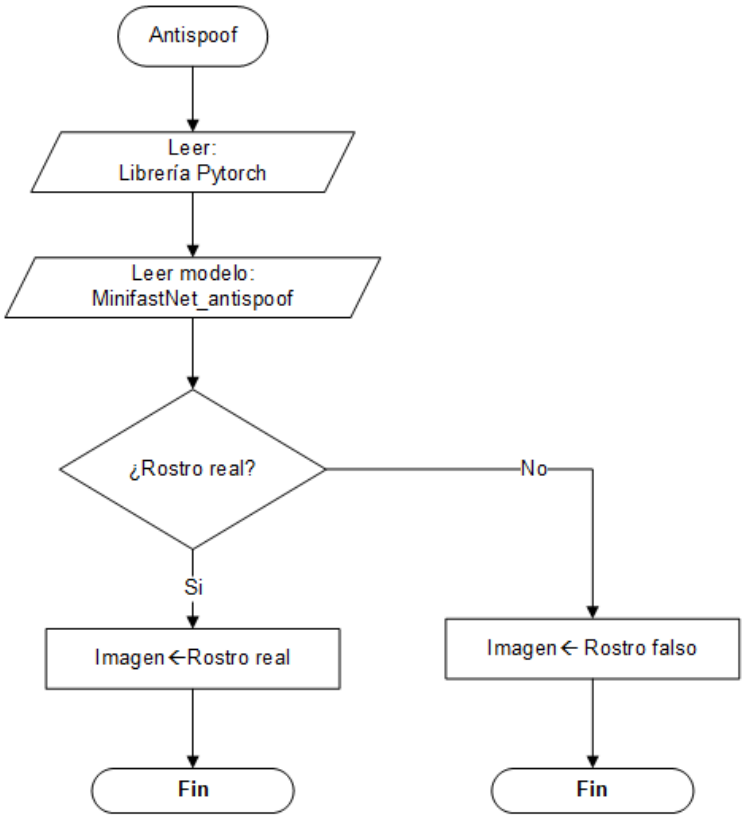
Nota: La figura muestra la predicción del usuario identificado en el sistema de reconocimiento facial y los 468 rasgos faciales del modelo de MediaPipe.

Algoritmo de anti plagio o detección de vida

La detección de suplantación de rostros, también conocida como detección de vida, es un desafío y una de las áreas de investigación más activas en visión por computadora vinculada a sistemas de identificación facial se refiere que se tiene hoy en día. Es por esto que se ha optado por dar una robustez al sistema de identificación facial y evitar de manera significativa posibles ataques al sistema detectando si el rostro es real o falso. Se observa en la Figura 73 el diagrama de flujo del algoritmo de vida implementado en el sistema de identificación facial.

Figura 73

Diagrama de flujo para el algoritmo de vida.



El modelo que se usó se ha adaptado al sistema de reconocimiento facial, como se explicó anteriormente viene con un detector de rostros, el modelo de RetinaFace, se detalla en la siguiente Tabla 16:

Tabla 16

Síntesis del algoritmo de vida

Librería	Función
PyTorch	Librería de aprendizaje automático de uso libre
anti_spoof_predict	Envía la predicción si el rostro detectado es real o falso

Nota: En la tabla se detalla las librerías y modelo aplicado al algoritmo de vida.

Aplicado el algoritmo de vida en tiempo real se tiene la predicción de un rostro real y un rostro falso, tal como se muestra en la Figura 74.

Figura 74

Izquierda: Identificación de un rostro real. Derecha: Identificación de un rostro falso.

Nota: En la figura de la izquierda muestra el reconocimiento facial de un usuario registrado al sistema en vivo (rostro real), mientras que en la imagen de la derecha se observa el rostro del mismo usuario pero desde un móvil (rostro falso).

Diseño y levantamiento de la interfaz web

Para el levantamiento de la interfaz web se utilizan las siguientes librerías que se muestran en la siguiente Tabla 17.

Tabla 17

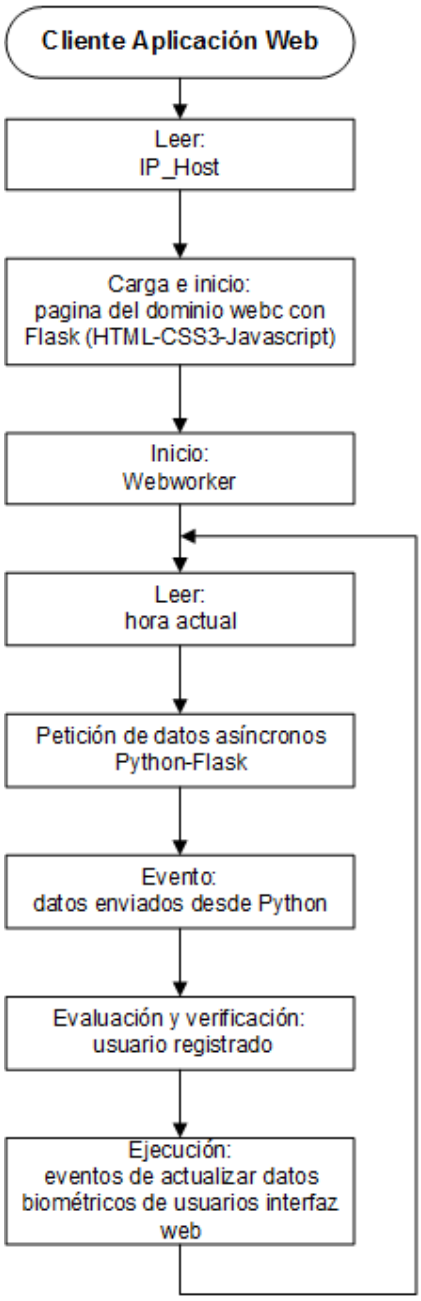
Librería aplicada para el diseño de la interfaz web

Librería	Función
Flask	Micro-frame, para el desarrollo de aplicaciones Web

Tomando en consideración los aspectos recomendados en el capítulo 3, se ha tomado un diseño acorde al usuario, tal como se muestra en el siguiente diagrama de flujo, ver Figura 75.

Figura 75

Diagrama de flujo para el diseño web.



Se inicia con una presentación de la interfaz web, tal como se muestra en la Figura 76.

Figura 76

Interfaz web, ventana de inicio

En la parte superior proporciona el acceso a diversas configuraciones del sistema y la detección e identificación en vivo, ver Figura 77.

Figura 77

Interfaz web, menú de navegación

El usuario debe dirigirse a configurar la cámara IP, tal como se lo realizó en el acceso a la cámara, esto se lo realiza dirigiéndose al menú de opciones en Cámara-Configuraciones, ver Figura 78.

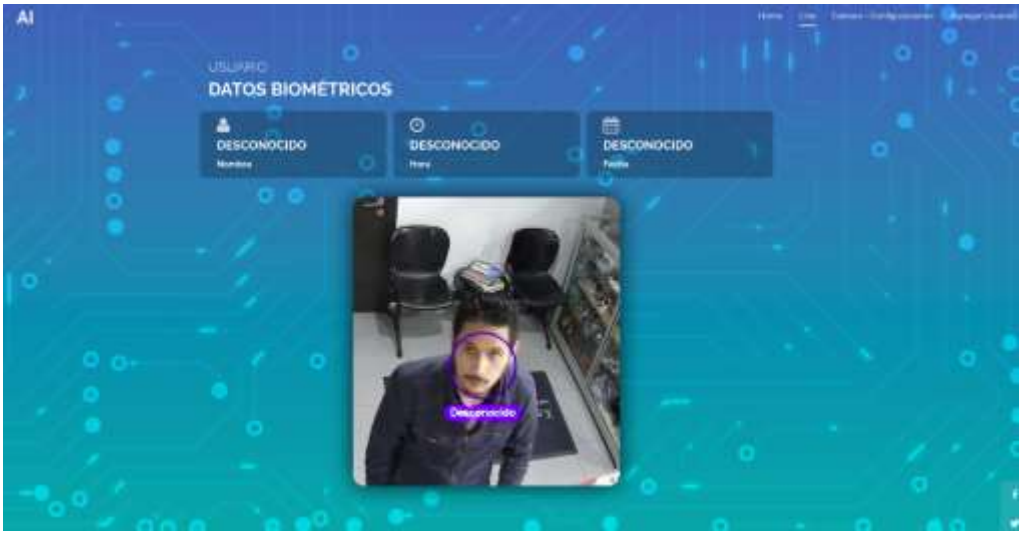
Figura 78

Interfaz web, Cámara configuración

Ya configurada la cámara se muestra la zona de detección de la cámara IP, esto se lo realiza accediendo al menú Live, ver Figura 79.

Figura 79

Interfaz web, Live



Situado en este punto, se puede observar los datos de usuario, la hora y fecha de registro, tal como se muestra en la Figura 80.

Figura 80

Datos biométricos de registro

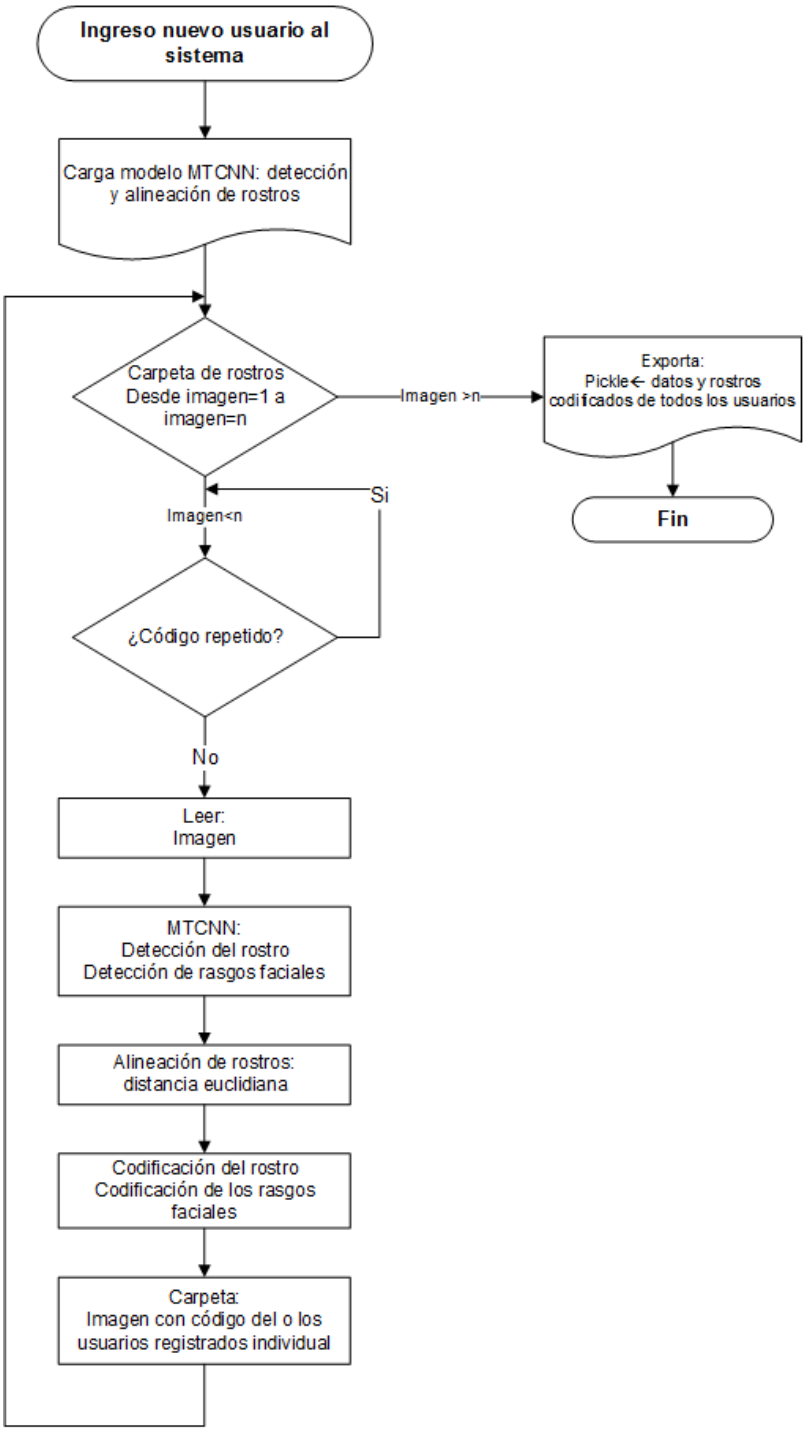
Procedimiento para el registro de los rostros.

En esta fase del proyecto se toman muestras de fotos del personal que está ligado al módulo de reconocimiento facial a través de algoritmos y modelos de aprendizaje profundo junto con visión por computadora. Las variables de los datos que se tomarán se indican en la tabla.

Estos datos se colocaran en un archivo xlxs, en donde deben constar siempre la información de cada usuario a registrar, para así cargar al sistema por medio de la interfaz web con una función de ajuste que extraerá la información biométrica del usuario, asignándole un código único el cual servirá para su identificación y posterior validación del sistema. Todo esto se explica en el siguiente diagrama de flujo de la Figura 81.

Figura 81

Diagrama de flujo de ajuste nuevo usuario



Se debe situar en la interfaz web, en la parte de nuevo usuario, tal como se muestra en la Figura 82.

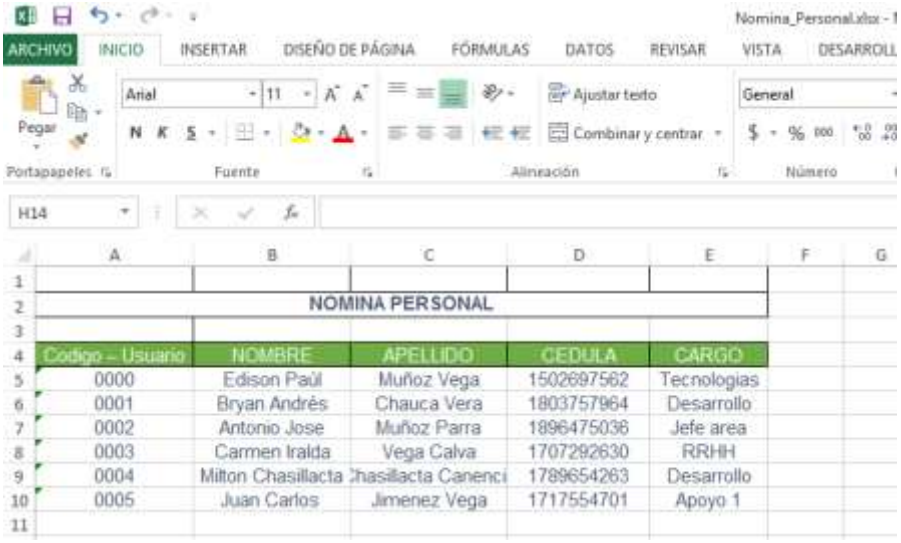
Figura 82

Interfaz web, Agregar Usuario

Se puede apreciar en la Figura 82, las opciones de *Agregar, Eliminar Usuario,* e *Iniciar ajuste.* Al crear un nuevo usuario es recomendable se recalca que también se encuentre anteriormente registrado en la base de datos (archivo Excel .xlsx) del registro de usuarios, Figura 83. Puesto que el sistema de reconocimiento facial usa el código identificador de cada persona para extraer los datos necesarios desde aquel archivo para posteriormente ser mostrados en la interfaz web durante el reconocimiento facial, ver Figura 84.

Figura 83

Hoja de datos de los usuarios registrados al sistema de identificación facial



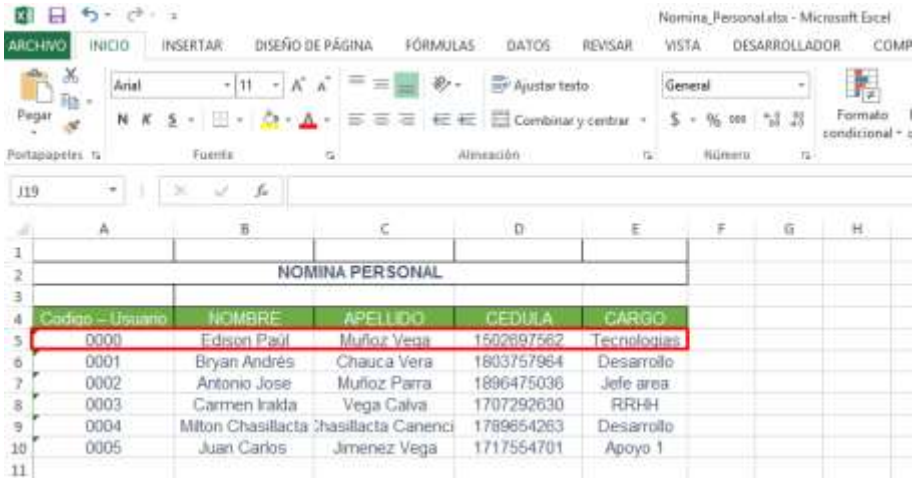
NOMINA PERSONAL

Codigo – Usuario	NOMBRE	APELLIDO	CEDULA	CARGO
0000	Edison Paúl	Muñoz Vega	1502697562	Tecnologias
0001	Bryan Andrés	Chauca Vera	1803757964	Desarrollo
0002	Antonio Jose	Muñoz Parra	1896475036	Jefe area
0003	Carmen Iralda	Vega Calva	1707292630	RRHH
0004	Milton Chasillacta	Chasillacta Canenci	1789654263	Desarrollo
0005	Juan Carlos	Jimenez Vega	1717554701	Apoyo 1

Se sugiere que los campos de la cabecera de la base de datos no sean modificados ya que podrían presentar inconvenientes el sistema al momento de obtener los datos biométricos de los usuarios.

Figura 84

Información y código personal del usuario registrado en el sistema facial



NOMINA PERSONAL

Codigo – Usuario	NOMBRE	APELLIDO	CEDULA	CARGO
0000	Edison Paúl	Muñoz Vega	1502697562	Tecnologias
0001	Bryan Andrés	Chauca Vera	1803757964	Desarrollo
0002	Antonio Jose	Muñoz Parra	1896475036	Jefe area
0003	Carmen Iralda	Vega Calva	1707292630	RRHH
0004	Milton Chasillacta	Chasillacta Canenci	1789654263	Desarrollo
0005	Juan Carlos	Jimenez Vega	1717554701	Apoyo 1

En base al punto anterior se crea diariamente el archivo de asistencia con la información de entrada-salida de cada usuario, en el siguiente diagrama de flujo se explica mejor cómo está estructurado, ver Figura 85. Así como el archivo Xlxs, Figura 86.

Figura 85

Diagrama de flujo del reporte de asistencia diario

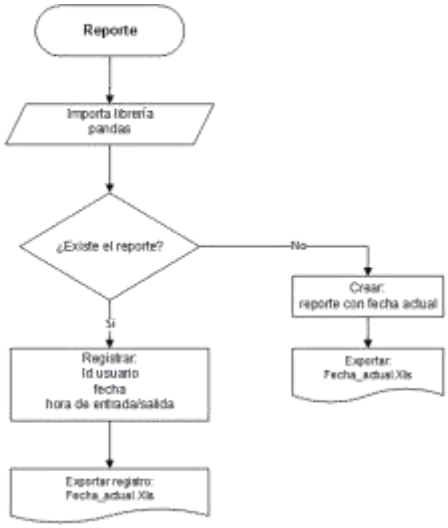
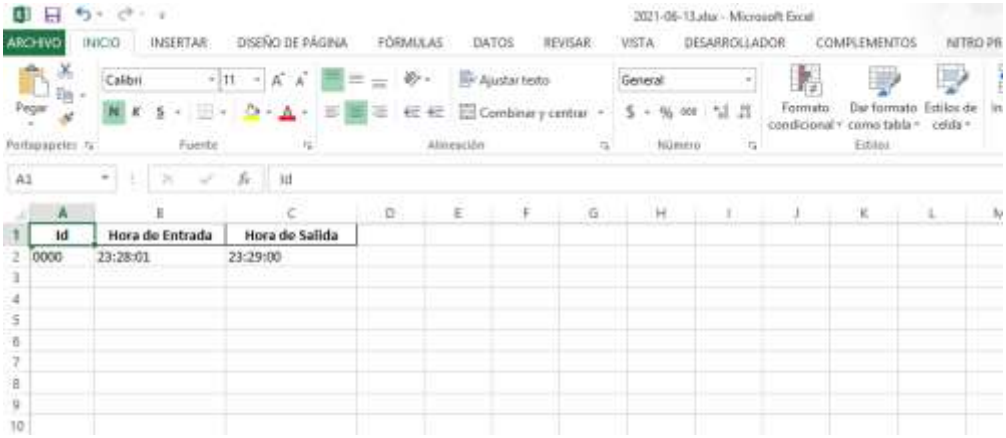


Figura 86

Archivo de Registro de Asistencia



Nota: En la figura se muestra el archivo de registro formado por el ID del usuario, ingresó-hora y salida-hora.

Ya explicados estos puntos ahora se inicia con el ingreso del nuevo usuario, se debe ingresar el código identificador registrado en la opción *Agregar Usuario*. Como ejemplo se ha tomado el código identificador ‘0000’, ver Figura 87, del archivo base de los datos que contiene la información de los usuarios.

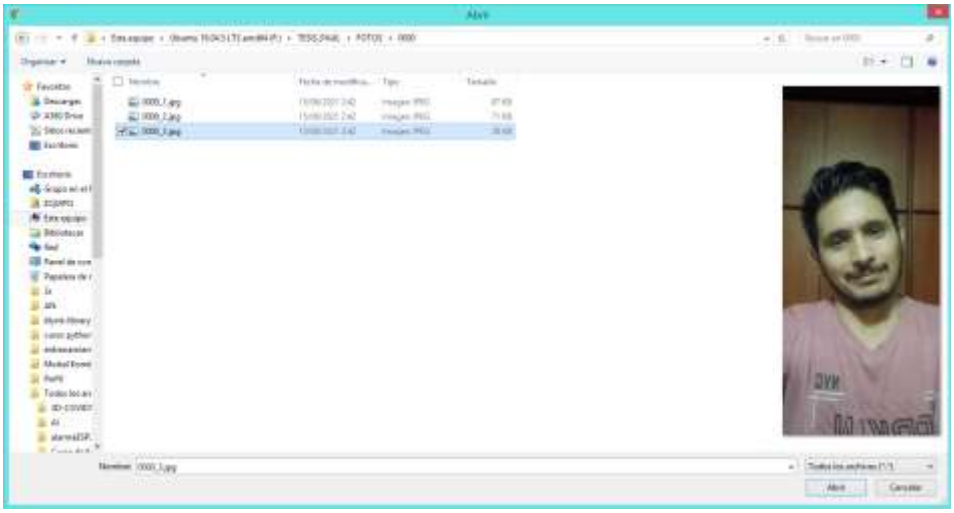
Figura 87

Creación del nuevo usuario

Se presiona el botón *Elegir Archivos,* ver Figura 87 y se agrega la imagen del nuevo usuario, pueden ser varias imágenes, como sugerencia se lo realiza con 2 imágenes correspondientes al nuevo usuario, es recomendable que estas imágenes tengan una resolución entre 3 y 4MP, ver Figura 88.

Figura 88

Carga de las imágenes para del nuevo usuario



Una vez agregadas la imagen se presiona el botón *Agregar.* De manera inmediata se muestra un mensaje que los archivos fueron cargados exitosamente y el sistema admite automáticamente al nuevo usuario, ver Figura 89.

Figura 89

Izquierda: Información pre-cargada. Derecha: Información subida con éxito

A continuación hay que cargar al sistema biométrico la información del usuario, esto se lo realiza presionando en INICIAR AJUSTE, ver Figura 90.

Figura 90

Ajustar datos al sistema

Eliminación de usuario

Se toma el código identificador del usuario a eliminar (pueden encontrarlo en el listado de usuarios), se coloca en la casilla de la opción *Eliminar Usuario* y se presiona el botón *Eliminar*, a continuación, se muestra un mensaje que el archivo fue eliminado correctamente, ver Figura 91.

Figura 91

Izquierda: Eliminación del usuario. *Derecha: Usuario eliminado*

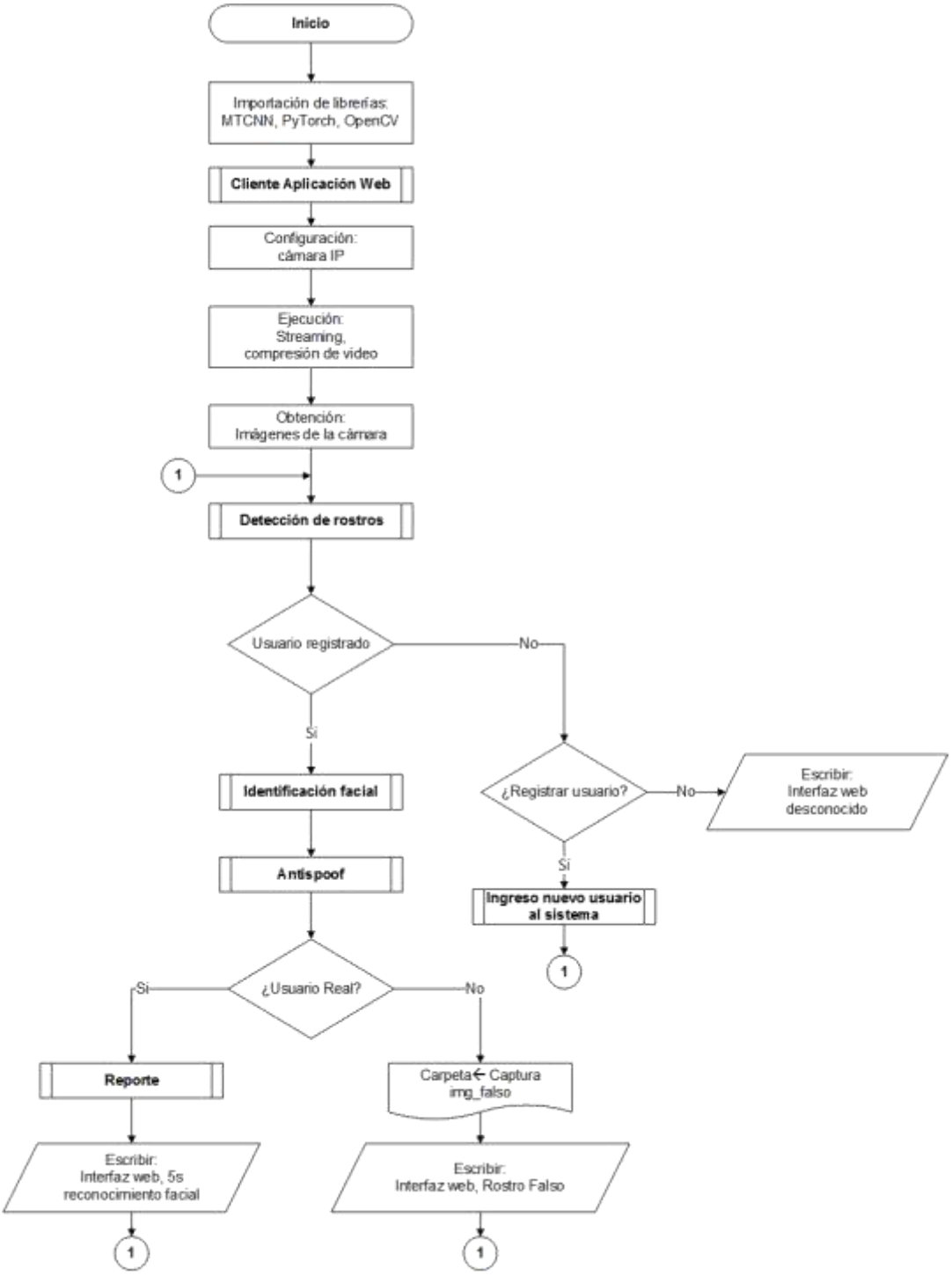
De forma inmediata el sistema elimina la información del usuario actualizando de forma automática, pero aún queda eliminar la información biométrica del usuario borrado, y esto se realiza con tal como se explicó en la Figura 90.

Todo esto ocurre en tiempo real y sin necesidad de detener el sistema para cargar los datos del nuevo usuario, siendo esto una característica importante a la hora de ajustar usuarios de manera más rápida y efectiva.

Finalmente se tiene estructurado el sistema de identificación facial, tal como se muestra en el diagrama de flujo de la Figura 92.

Figura 92

Diagrama de flujo del sistema de identificación facial



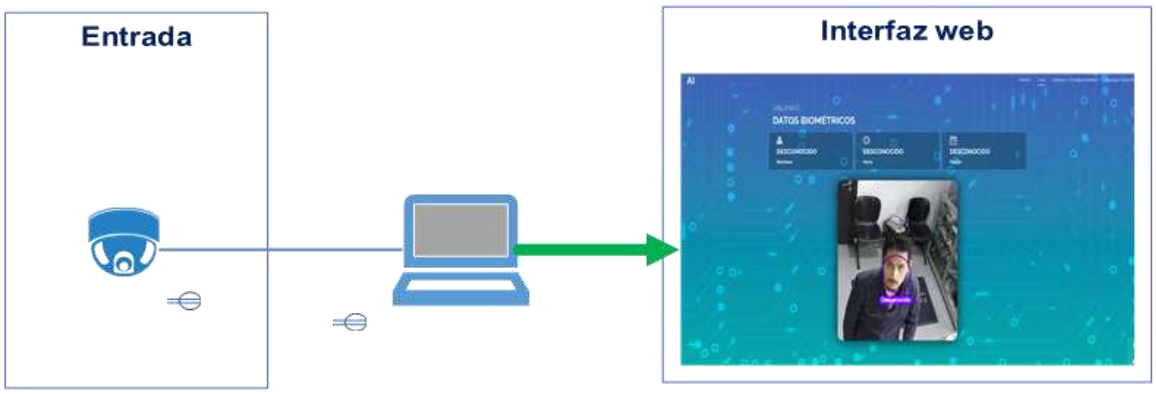
Nota: La figura muestra el diagrama de flujo del sistema de identificación facial con cada una de sus funciones.

Diagrama de conexión

El diagrama de conexión es sumamente sencillo y claro de explicar, se inicia con el ingreso de las imágenes a través de la cámara, estas imágenes son procesadas por medio de la computadora para finalmente mostrar en la interfaz gráfica la detección e identificación facial. Tal como se muestra en la Figura 93.

Figura 93

Diagrama de conexión



Nota: En la figura se muestra la conexión del sistema de identificación facial.

Se detalla a su vez los componentes que están constituido en la siguiente Tabla 18:

Tabla 18

Simbología de conexión

Símbolo	Descripción
	cable UTP-cat5 o 6e
	punto de energía
	Computadora portátil
	cámara

Nota: En la tabla se observa la simbología utilizada para la conexión del sistema de identificación facial

Capítulo V

Pruebas y resultados

Análisis

Los resultados se concentran en la respuesta del sistema a una cantidad de experimentos realizados por medio de voluntarios que ingresaron en el sistema de identificación facial con ambientes controlados y variaciones de luz natural.

A continuación se establece el contexto donde se desarrolla las pruebas y para el análisis de los resultados se siguió el siguiente procedimiento:

- Realizar un estudio del porcentaje de acierto en el reconocimiento de personas que tiene el sistema en diferentes ambientes: distancia, nivel de lúmenes, diferentes perfiles del rostro con vista a la cámara, eficiencia del algoritmo de reconocimiento frente a los diferentes escenarios, y el tiempo de identificación que demora el sistema en reconocer al usuario registrado.
- Analizar la tasa de falsa aceptación (FAR, False Accept Rate, traducido del inglés) y tasa de falso rechazo (FRR, False Rejection Rate, traducido del inglés) de las personas que ingresan al sistema por medio del algoritmo de vida
- Además de las pruebas mencionadas se realiza un estudio del gasto computacional que consume el sistema de identificación facial

Se inicia con el tipo de pruebas a realizar además de analizar el porcentaje de acierto que tiene el sistema de identificación facial, en los cuales interfieren el nivel de luz ya sea natural y artificial. Cabe mencionar que el sistema reconoce usuario por usuario, y no es múltiple, siendo este aspecto a futuro como mejora del presente proyecto de investigación.

El objetivo a determinar fue: cumple su objetivo de reconocimiento facial y no detectar al usuario registrado en el sistema facial si es una foto o imagen del teléfono móvil.

Normas de seguridad

Antes de iniciar con las pruebas se debe aclarar que si bien como se detalla en la Tabla 19 conforme a los voluntarios, no todos están disponibles en todo momento y es debido a dos factores:

1. En el escenario de las pruebas no todos residen en el mismo lugar. Por tal razón no se puede efectuar pruebas a toda hora y solo se incluyen usuarios cercanos a la zona de las pruebas, aproximadamente 5 km a la redonda.
2. Debido a la pandemia que azota nuestra era solo se incluyeron a personas que se encuentran parcial o totalmente vacunadas.

Cabe mencionar que se acata las normas de seguridad conforme a la no aglomeración de personas dadas por las entidades de gobierno y de esta forma evitar contagios, ya que si bien ha disminuido porcentualmente los fallecidos todavía no se puede tener 100% de confianza de tener riesgos mortales debido a las variantes que existen del covid-19 (El Comercio, 2021) (El universo, 2021), además por tal razón no se puede tener tantos usuarios para validar los datos debido a la explicación antes mencionada.

Desarrollo de pruebas

Se inicia con los voluntarios, los cuales fueron 11 personas, todas ellas con su respectiva foto como se explica en la Tabla 19.

Tabla 19

Usuarios registrados

Usuario	Código	Cantidad de fotos
Paul Muñoz	0000	2
Antonio Muñoz	0001	2
Carmita Vega	0002	2
Juan Jiménez	0003	2
Jimena Muñoz	0004	2
Andrés Vila	0005	2
Milton Chasillacta	0006	2
Diego Chasillacta	0007	2
Luis Salcedo	0008	2
Germánico Chasillacta	0009	2
Nelly Canencia	0010	2

Para iniciar el sujeto debe mantener la cara firme durante 2 segundos; esto genera un vector con 250 predicciones de las distancias más cercanas entre todos los rostros ajustados al sistema de datos; según la métrica de distancia explicada por el modelo de FaceNet, en el cual dos rostros coinciden si su distancia es menor a 0,4 (umbral predeterminado), se observa en la Figura 94.

Figura 94

Identificación facial frontal

Para conseguir una medida de probabilidad se debe convertir el puntaje de distancia facial predicho en un puntaje porcentual de coincidencia, esto se refleja en la Figura 95.

Figura 95

Valor del porcentaje de coincidencia

Entonces bajo este valor umbral la imagen positiva que ingrese al sistema comparada con la imagen ancla ajustada al sistema de identificación facial (explicado en el capítulo 4 y en la función de triple pérdida), se regirá para decir si ha identificado o no es al usuario o en su defecto si existe un falso positivo.

Prueba 1

A continuación se detalla la prueba con la que se inicia la evaluación, en la siguiente Tabla 20.

Tabla 20

Prueba de identificación facial a distintas distancias

Prueba 1	Detalle
Distancia 1	120 cm- vista frontal-nivel de lúmenes (luz natural)
Distancia 2	90 cm-vista frontal-nivel de lúmenes (luz natural)
Distancia 3	50 cm-vista frontal-nivel de lúmenes (luz natural)

Para esta prueba se toma varias métricas como son, el nivel de lúmenes de la luz natural, la distancia y el tiempo respuesta de la detección de personas, esta prueba se la realizó a partir de las 13:00 pm que proporciona un alto grado de luz natural, en los anexos se muestran los voluntarios disponibles para esta prueba, en siguiente Tabla 21 se muestran los resultados de esta prueba:

Tabla 21

Resultados de identificación facial a distintas distancias

		Distancia 1		Distancia 2		Distancia 3	
Usuario -código	**Lúmenes [lx]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**
0000	328	99.92	2.91	99.78	3.24	99.96	2.39
0001	334	99.90	2.58	99.55	3.46	99.93	2.74
0002	336	92.28	2.88	99.86	2.89	99.88	2.14
0004	329	99.92	3.16	99.85	2.15	99.88	1.99
0005	318	98.52	2.88	99.74	2.72	99.66	2.85
0006	334	99.11	3.04	99.54	2.96	99.05	1.96
0007	306	98.00	3.85	99.77	2.96	99.70	2.31
0008	320	99.66	3.61	99.35	3.93	99.08	2.16
0009	315	99.97	3.64	99.96	3	99.24	2.13
0010	320	99.17	3.67	99.29	3.4	99.6	2.14
Prom. Total	**324**	**98.645**	**3.222**	**99.669**	**3.24**	**99.598**	**2.281**

Nota: En esta tabla se muestran los resultados de la prueba 1 a diferentes distancias, nivel de lúmenes y tiempo de respuesta de la identificación, además se incluye el promedio obtenido en cada una de las variables mencionadas.

En las siguientes Figura 96 y Figura 97 se muestran las columnas agrupadas en los cuales se observa de mejor forma el contenido de la Tabla 21.

Figura 96

Precisión del sistema (%) del reconocimiento facial en la prueba 1

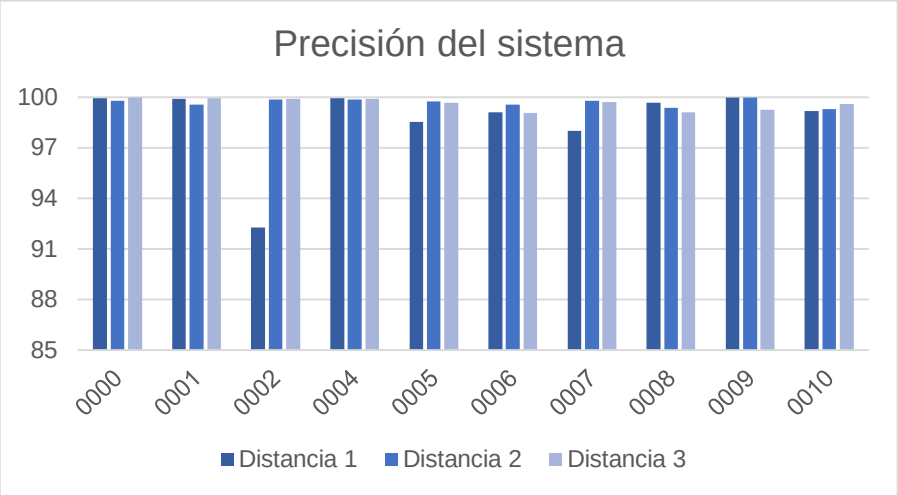
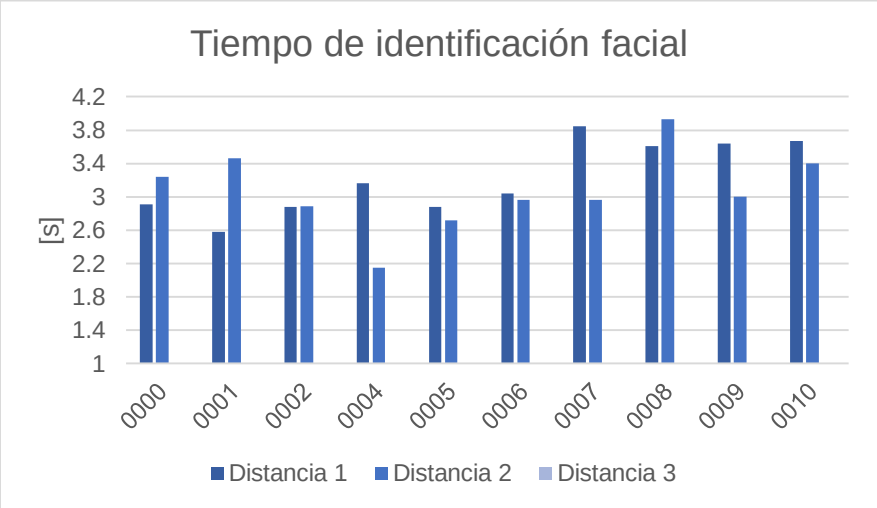


Figura 97

Tiempo de respuesta (seg) del sistema del reconocimiento facial en la prueba 1

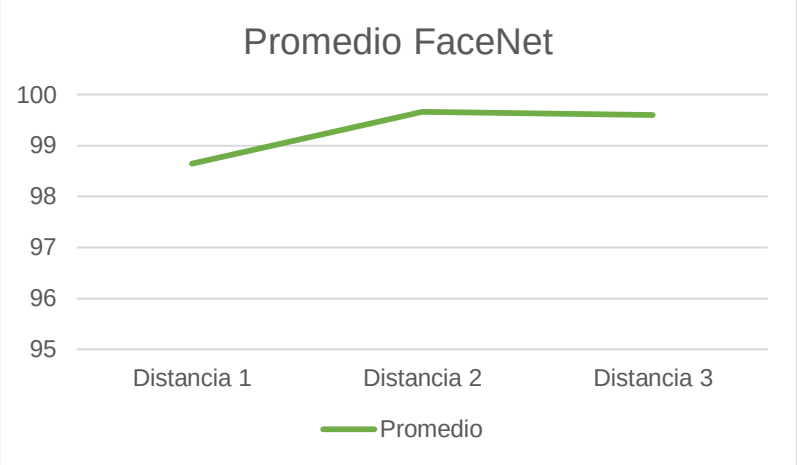


Análisis de la prueba 1

Ya obtenidos los datos se puede realizar una análisis adecuado del rendimiento obtenido en esta prueba. En la Tabla 21, también se obtuvo un promedio de la precisión conforme a la distancia se refiere, estos resultados se pueden observar en la siguiente Figura 98.

Figura 98

Promedio (%) de la distancia del sistema del reconocimiento facial en la prueba 1



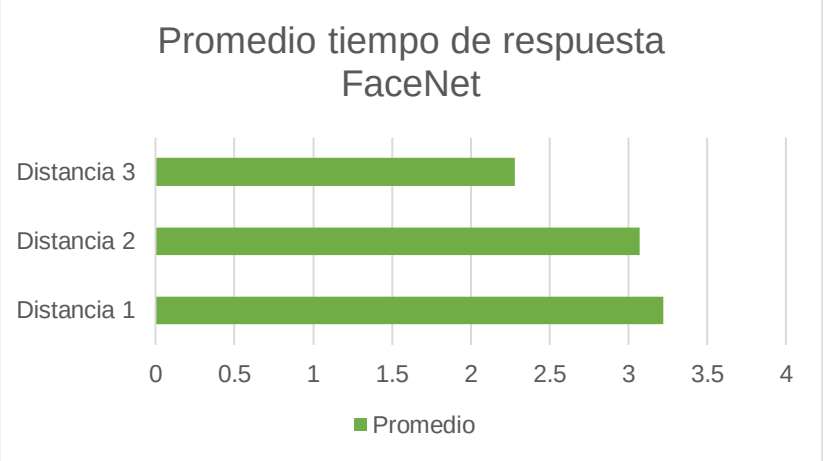
Se puede observar que a una distancia 3 (0.50 mts) alcanza un precisión elevada de identificación bastante coincidente con el rostro del sujeto registrado.

A la distancia 2 (0.90 mts), sobresale un porcentaje de coincidencia elevado cercano al 100%, finalmente aunque no menos alentador en la distancia 1 (1.20 mts), existe un pequeño declive en cuanto a la precisión de identificación se refiere, pero se encuentra en un 98% de precisión.

El tiempo de identificación con respecto a la distancia es otro parámetro que se obtiene en la tabla, por medio de su promedio. Se puede apreciar en la figura, que a menor distancia (distancia 3) el tiempo de respuesta es más bajo, a diferencia de la distancia 1, en el cual hay más tiempo en cuanto a reconocer al usuario se refiere.

Figura 99

Promedio del tiempo de respuesta (seg) del sistema del reconocimiento facial en la prueba 1



Prueba 2

En la prueba 2, como se explica en la Tabla 22, se cambia el nivel de luz conforme a la distancia, esto ocurre para evaluar qué tan bueno es el sistema a cambios de luz abruptos.

Tabla 22

Prueba de identificación facial a distintos niveles de luz

Prueba 2	**Detalle**
Nivel de luz 1	1.20 cm- vista frontal-nivel de lúmenes (luz artificial)
Nivel de luz 2	90 cm-vista frontal-nivel de lúmenes (exceso de luz)
Nivel de luz 3	50 cm-vista frontal-nivel de lúmenes (poca luz)

En la siguiente Tabla 23 se obtienen se encuentran los resultados obtenidos:

Tabla 23

Resultados de identificación facial a distintas niveles de luz

	Distancia 1.20mts Luz 1≈365 lx		**Distancia 0.90 mts Luz 2 ≈660 lx**		**Distancia 0.50 mts Luz 3≈40 lx**	
Usuario-Código	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**
0000	99.75	3.36	99.88	3.49	99.98	3.97
0001	95.03	3.39	99.30	3.45	99.73	3.81
0002	94.95	2.4	99.89	3.08	99.65	3.49
0003	98.53	3.02	99.97	2.92	99.98	3.1
0005	99.55	2.53	92.88	3.21	99.66	3.46
0006	99.96	4.13	99.03	3.39	97.19	3.77
0007	98.1	3.00	99.83	3.37	96.85	3.81
0008	98.91	2.91	98.94	3.68	98.66	3.71
0009	98.00	3.04	96.91	4.09	95.55	4.43
0010	94.78	3.24	95.36	3.26	94.44	3.98
Prom. Total	**97.756**	**3.102**	**98.199**	**3.394**	**98.169**	**3.753**

Nota: En esta tabla se muestran los resultados de la prueba 2 a diferentes niveles de luz y tiempo de respuesta de la identificación, además se incluye el promedio obtenido en cada una de las variables mencionadas.

En las Figura 100 y Figura 101 se muestran las columnas agrupadas en los cuales se observa de mejor forma el contenido de la Tabla 23.

Figura 100

Precisión del sistema (%) del reconocimiento facial en la prueba 2

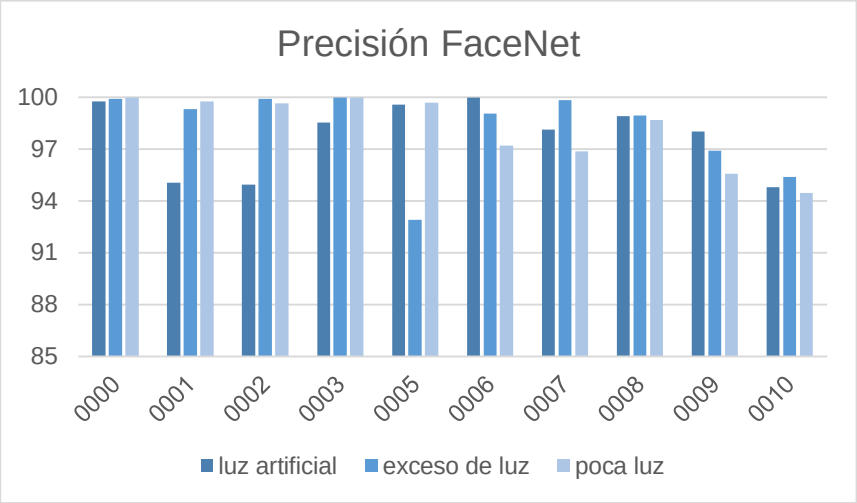
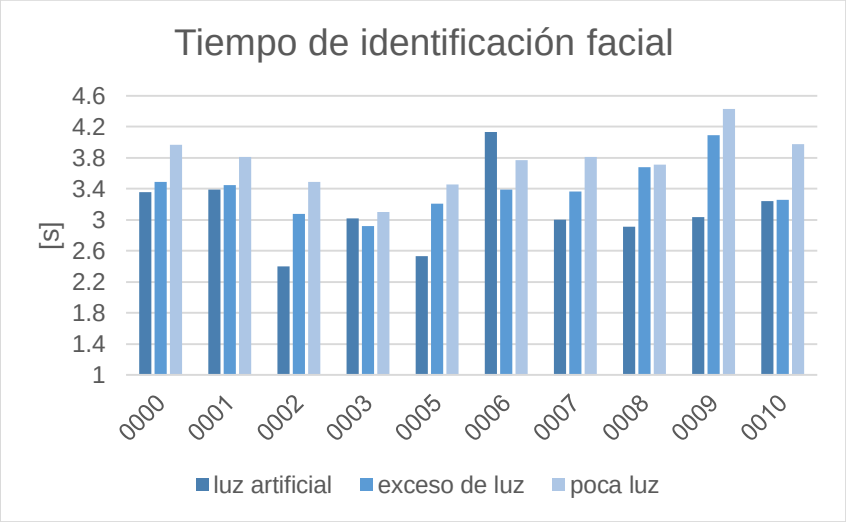


Figura 101

Tiempo de respuesta (seg) del sistema del reconocimiento facial en la prueba 2

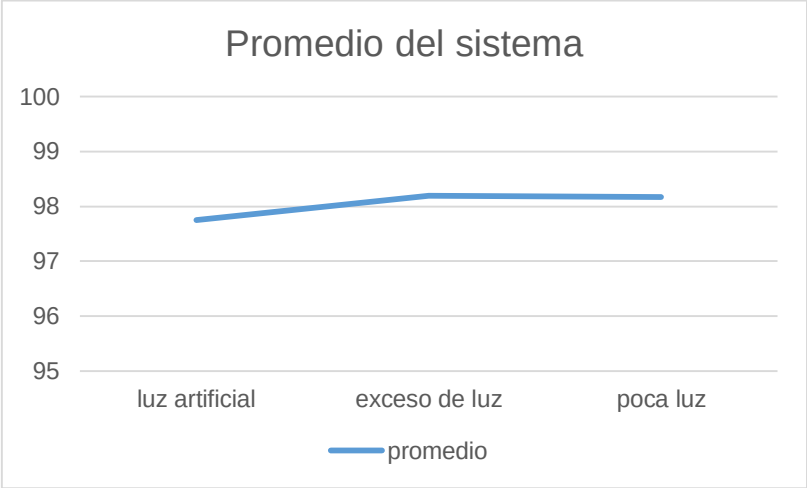


Análisis de la prueba 2

En la Tabla 23, se obtuvo un promedio de los resultados coincidentes con los distintos cambios de luz aplicados a esta prueba, una interpretación gráfica se observa en la Figura 102, que muestra el rendimiento del sistema ante estos cambios de luz.

Figura 102

Promedio del nivel de luz (%) del sistema del reconocimiento facial en la prueba 2

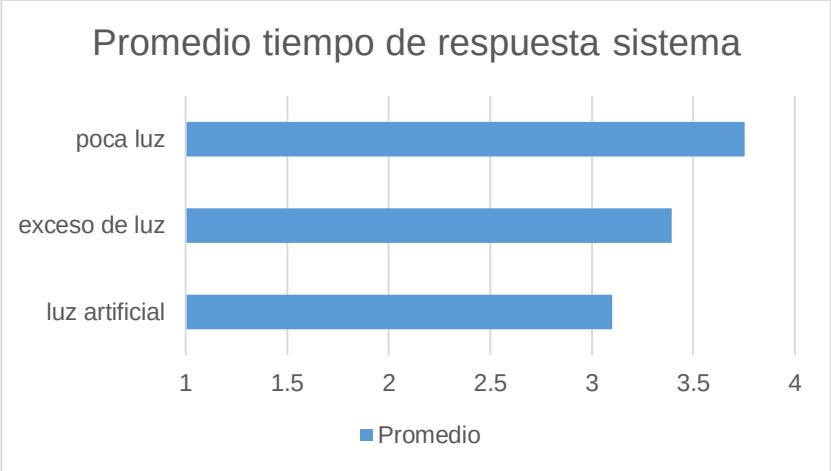


Para condiciones de luz en el cual la luz artificial incide directamente en el rostro del usuario a detectar, se observa que el rendimiento del sistema bordea el 98% de precisión en la detección. Con un exceso de luz, se aprecia que el sistema incrementa su rendimiento aproximadamente en 98.2%. Y finalmente con poca luz el rendimiento del sistema se mantiene en un 98.2%, identificando y registrando al usuario de forma satisfactoria.

No obstante como en la prueba 1, se toma el promedio del tiempo de identificación con respecto a la iluminación que temporiza el sistema ante estos cambios de luz, tal como se muestra en la Tabla 23 y se detalla en la Figura 103.

Figura 103

Promedio del tiempo de respuesta (seg) del sistema del reconocimiento facial en la prueba 2



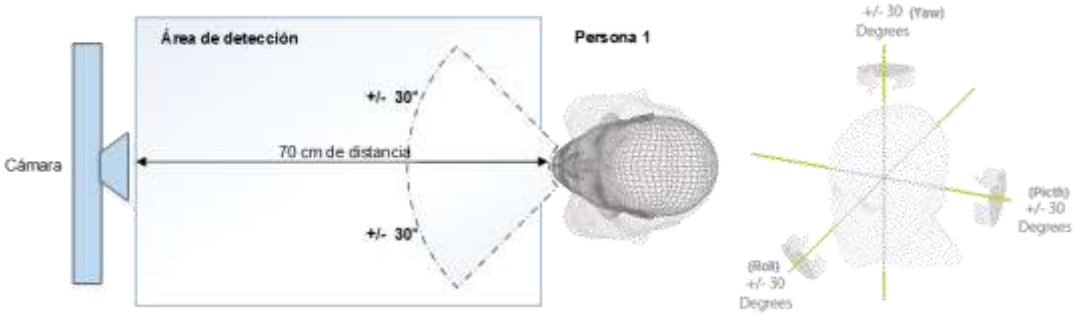
La Figura 103, es clara, con luz artificial existe un tiempo menor en cuanto a la identificación se refiere, ya que si bien el sistema tiene un rendimiento elevado tal como se explicó en el ítem anterior, al tener poca luz tiende a demorarse más en identificar al usuario.

Prueba 3

Para esta prueba se tomó una distancia fija y además se observa la identificación del usuario a distintos perfiles del rostro tal como se muestra en la Figura 104 siguiente, se toma el ángulo en Yaw para los perfiles del rostro:

Figura 104

Prueba de identificación de perfil del usuario.



Nota: El gráfico muestra cómo se realizó la prueba de identificación de perfiles del rostro. Tomado y adaptado de *ZKTeco*: *G4 Datashet,* (ZKTeco, 2019)

Esta prueba está detallada en la Tabla 24:

Tabla 24

Prueba de identificación facial de perfil.

Prueba 3	Detalle
Distancia 4	70 cm-perfil izquierdo (30°) del rostro-nivel de lúmenes (luz natural)
Distancia 4	70 cm-perfil frontal del rostro-nivel de lúmenes (luz natural)
Distancia 4	70 cm-perfil derecho (-30°) del rostro-nivel de lúmenes (luz natural)

Los resultados se pueden observar en la siguiente Tabla 25:

Tabla 25

Resultados de identificación facial con perfiles del rostro

		Perfil Izquierdo		Perfil Frontal		Perfil derecho	
Usuario -código	**Lúmenes (lx)**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**
0000	370	98.93	3.37	99.14	1.77	99.96	3.07
0001	345	99.76	2.61	99.93	2.01	99.98	3.12
0002	360	99.86	2.83	99.95	2.2	99.88	3.82
0004	355	99.96	1.93	99.99	2.46	99.99	2.11
0005	352	98.52	2.67	99.55	2.16	98.44	2.92
0006	356	99.36	2.35	99.62	2.24	98	2.92
0007	345	99.32	2.59	99.3	2.57	95.65	3.48
0008	369	99.98	3.33	99.99	2.98	99.91	3.87
0009	350	98.34	2.61	99.6	2.19	95.58	3.02
0010	366	92.42	1.98	99.17	3.12	91.66	2.96
Prom. Total	**356.8**	**98.645**	**2.627**	**99.624**	**2.37**	**97.905**	**3.129**

Nota: En esta tabla se muestran los resultados de la prueba 3 en diferentes perfiles del rostro, a una distancia fija y además el tiempo de respuesta de la identificación, se incluye el promedio obtenido en cada una de las variables mencionadas

En las siguientes Figura 105 y Figura 106 se muestran las columnas agrupadas en los cuales se observa de mejor forma el contenido de la Tabla 25.

Figura 105

Precisión del sistema (%) del reconocimiento facial en la prueba 3

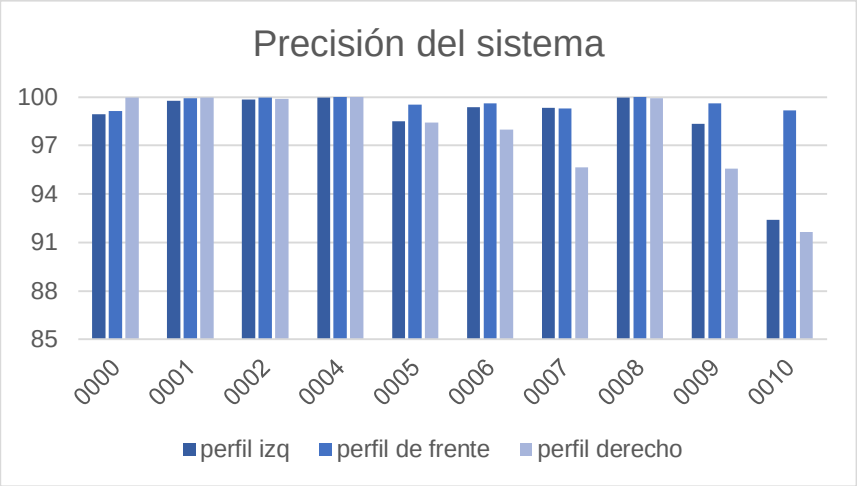
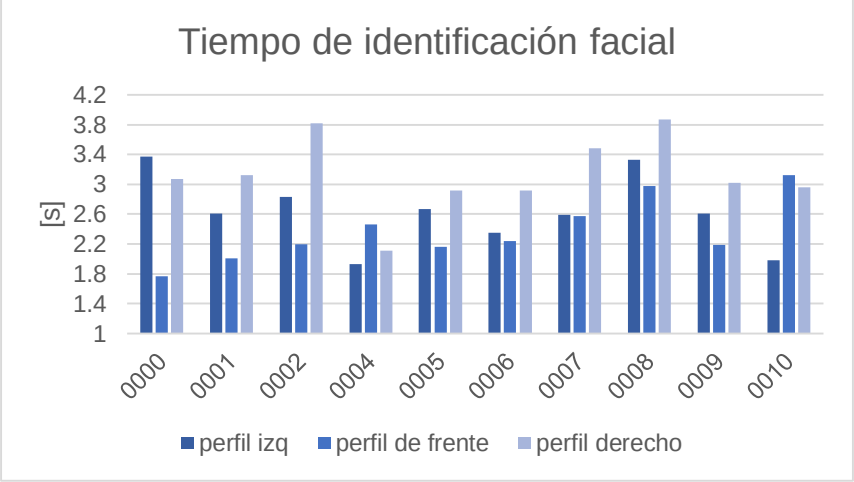


Figura 106

Tiempo de respuesta (seg) del sistema del reconocimiento facial en la prueba 3



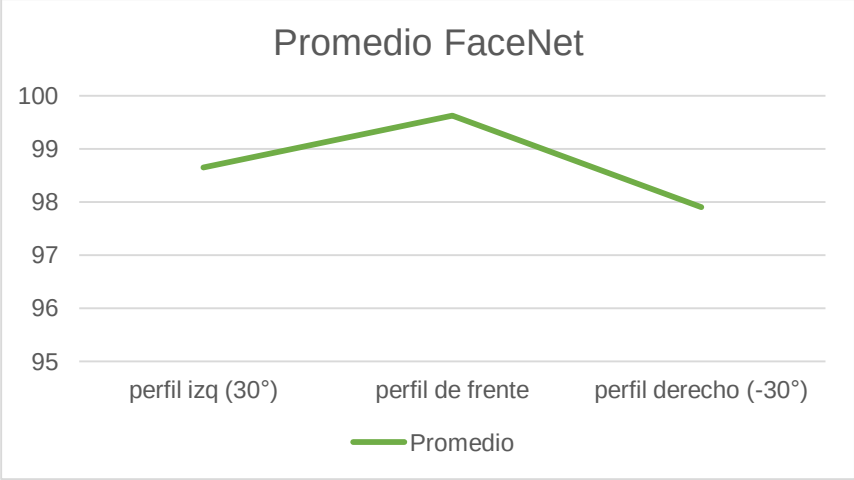
Análisis de la prueba 3

A diferencia de varios trabajos de investigación, no se toma muchas veces en cuenta esta prueba de la identificación del usuario de perfil ya sea izquierda o derecha, es por esta razón que se busca dar un valor agregado al sistema con esta prueba y verificar cuál es su rendimiento con respecto a este aspecto.

Se puede apreciar en la Tabla 25, el promedio obtenido del rendimiento del sistema cuando un usuario es identificado de perfil, la siguiente Figura 107 se observa a más detalle el rendimiento del sistema con esta prueba:

Figura 107

Promedio (%) de identificación de perfil del sistema del reconocimiento facial

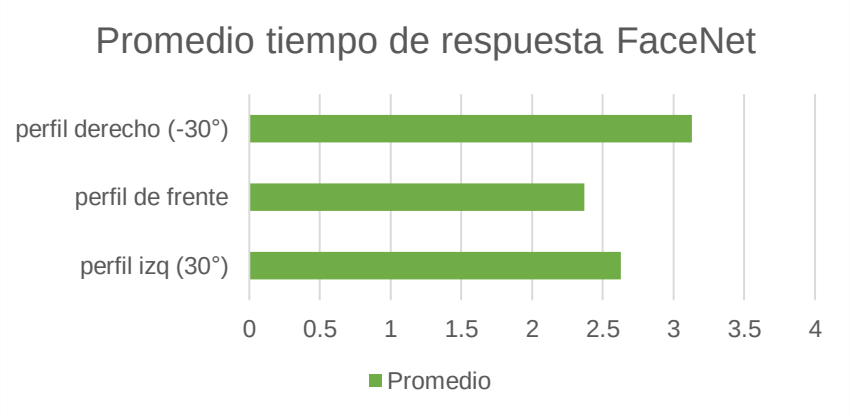


Es claro que en la Figura 107, existe una tendencia a identificar con mayor rendimiento cuando el individuo tiende a colocar su rostro en una posición frontal a la cámara con respecto a identificarlo si se encuentra de perfil, pero en términos de rendimiento se puede observar que tanto el perfil izquierdo y derecho bordean el 98% de rendimiento, siendo este aspecto muy eficiente para una identificación facial más precisa.

A su vez sumado a otras pruebas anteriores se debe evaluar qué tiempo se demora el sistema de identificación facial cuando el rostro se encuentra de perfil, este aspecto ligado al promedio obtenido en la tabla, se muestra la siguiente figura:

Figura 108

Promedio del tiempo (seg) de respuesta del sistema del reconocimiento facial en la prueba 3



Se aprecia claramente que el perfil derecho tiene un tiempo de detección más prolongado con respecto al perfil de frente, siendo este último es más rápido en todas las pruebas hasta el momento, es claro además que hay un tiempo más corto al momento de detectar el perfil izquierdo que el derecho, pero que no disminuye aun a la identificación del rostro con la postura de frontal de la cara del usuario.

Prueba 4

Sumado a la prueba 3, se agregó una prueba de las mismas características pero en un entorno con poca luz, con el perfil del rostro de los usuarios registrados parcialmente oscuro, en la Figura 109 y Tabla 26 se puede apreciar el detalle de la prueba 4:

Figura 109

Prueba de identificación de perfil del usuario con poca luz

Tabla 26

Prueba de identificación facial de perfil con poca luz

Prueba 4	Detalle
Distancia 4	70 cm-perfil izquierdo del rostro (30°)-nivel de lúmenes (poca luz)
Distancia 4	70 cm-perfil frontal del rostro-nivel de lúmenes (poca luz)
Distancia 4	70 cm-perfil derecho del rostro (-30°)-nivel de lúmenes (poca luz)

Los resultados obtenidos se detallan en la siguiente Tabla 27:

Tabla 27

Resultados de identificación facial con perfiles del rostro con poca luz

		Perfil Izquierdo		Perfil Frontal		Perfil derecho	
Usuario -código	**Lúmenes [lx]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**
0000	40	99.36	4.88	99.78	2.91	99.96	5.63
0001	65	99.68	6.53	99.85	3.17	99.73	6.4
0002	50	99.49	5.14	99.32	3.48	99.43	4.67
0003	48	97.61	4.77	99.98	2.52	99.98	5.81
0005	45	93.36	5.81	99.91	3.44	96.93	6.28
0006	69	85.43	5.25	99.67	2.28	85.74	5.27
0007	60	98.7	5.44	98.45	3.32	97.86	4.3
0008	40	91.35	6.16	98.78	2.49	94.94	4.12
0009	49	98.59	5.07	99.2	3.77	93.33	5.08
0010	45	95.35	5.96	99.24	3.62	97.19	5.42
Prom.. Total	**51.1**	**95.892**	**5.501**	**99.418**	**3.1**	**96.509**	**5.298**

Nota: En esta tabla se muestran los resultados de la prueba 4 en diferentes perfiles del rostro, a una distancia fija, con poca luz y además el tiempo de respuesta de la identificación, se incluye el promedio obtenido en cada una de las variables mencionadas.

En las Figura 110 y Figura 111 se muestran las columnas agrupadas en las cuales se observa de mejor forma el contenido de la Tabla 27.

Figura 110

Precisión del sistema (%) del reconocimiento facial en la prueba 4

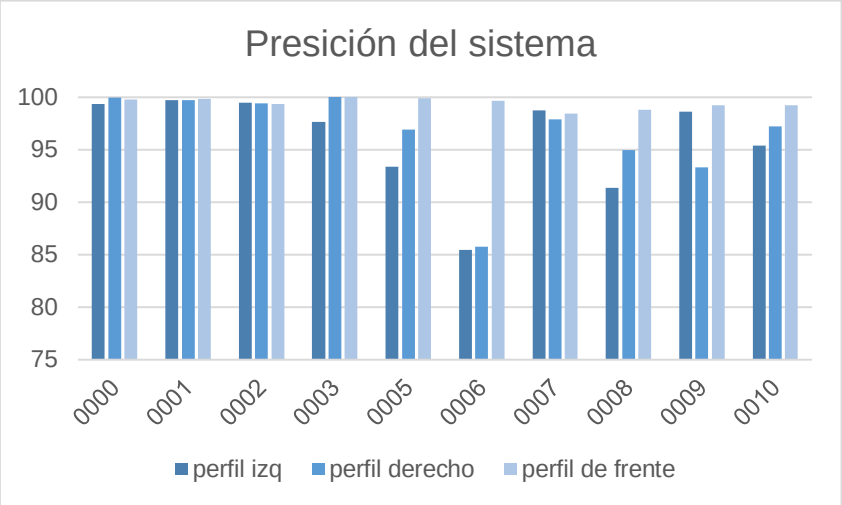
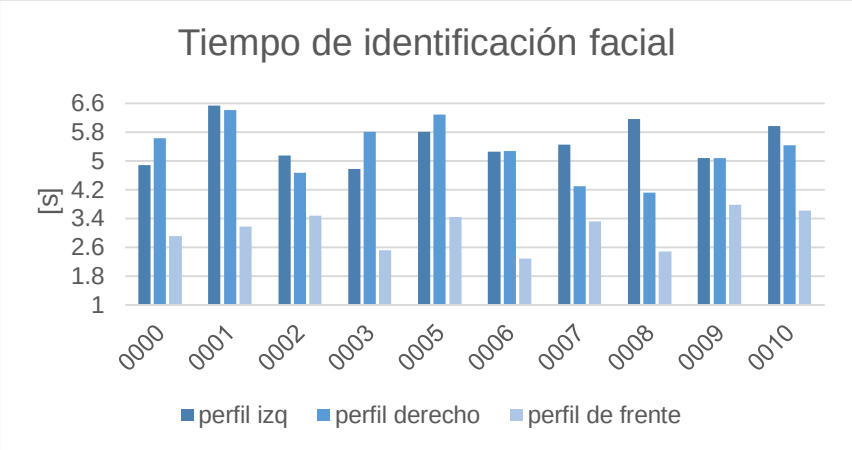


Figura 111

Tiempo de respuesta del sistema (seg) del reconocimiento facial en la prueba 4

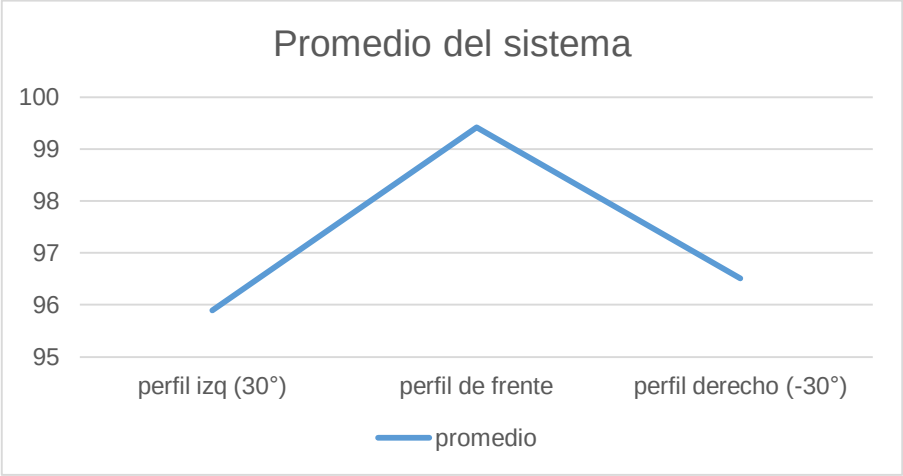


Análisis de la prueba 4

En la Tabla 27, se obtuvo el promedio de rendimiento del sistema, para tener más claro del rendimiento se puede apreciar en la siguiente Figura 112:

Figura 112

Promedio de identificación (%) de perfil del rostro a poca luz

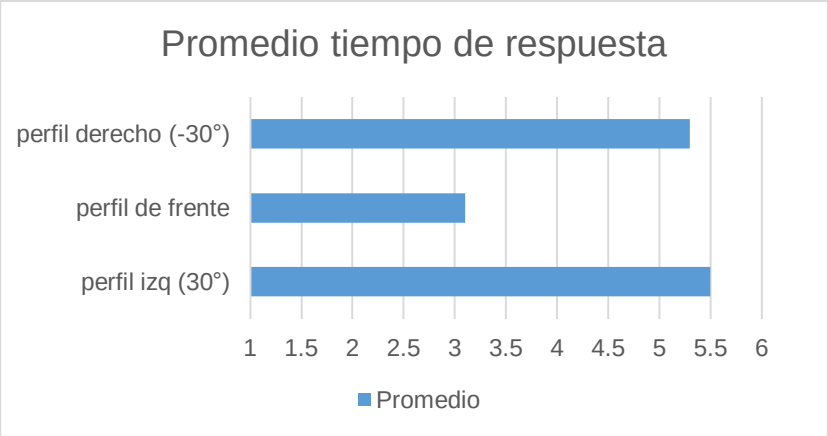


Tal como se analizó en la prueba 3 y ahora en la prueba 4, y como se observa en la Figura 112, la tendencia de identificación aumenta si el rostro se encuentra con un perfil frontal a la cámara llegando casi al 100% de rendimiento, a diferencia de una identificación del usuario cuando este se encuentra en un perfil izquierdo o derecho, aunque no se acerca a un máximo rendimiento, tiene buena aproximación pasando el 95% de rendimiento

En esta prueba 4 se cotejo si existe un aumento del tiempo en cuanto a la identificación se refiere, con el cambio de iluminación, la Figura 113 es clara, en ella se puede observar que los perfiles del rostro tienden a incrementar el tiempo de respuesta de la identificación facial, a diferencia del perfil frontal del rostro, pero si bien le cuesta al sistema más tiempo identificar al usuario en condiciones donde existe poca luz permite reconocer al usuario aun cuando se encuentre en estas condiciones.

Figura 113

Promedio del tiempo de respuesta (seg) del sistema del reconocimiento facial en la prueba 4



Prueba 5

En la siguiente prueba que se detalla en la Tabla 28, se ingresó obstáculos al sistema como una gorra, cofia o sombrero, lentes y una mascarilla para saber si es sistema es capaz de detectar al usuario registrado.

Tabla 28

Prueba de identificación facial con accesorios en el rostro

Prueba 5	**Detalle**
Obstáculo 1	a 70 cm, vista frontal, con gorra-sombrero-nivel de lúmenes (luz natural)
Obstáculo 2	a 70 cm, vista frontal, con lentes-nivel de lúmenes (luz natural)
Obstáculo 3	a 70 cm, vista frontal, con mascarilla-nivel de lúmenes (luz natural)

La prueba se la realiza tal como se muestra la siguiente Figura 114:

Figura 114

Prueba de identificación del usuario con accesorios en el rostro.

Los detalles de la prueba se muestran a continuación en la siguiente Tabla 29:

Tabla 29

Resultados de identificación facial con accesorios en el rostro

		Con gorra-sombrero		Con lentes		Con mascarilla	
Usuario -Código	**Lúmenes [lx]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**	**Precisión %**	**Tiempo [s]**
0000	354	99.08	4.06	98.42	6.05	96.76	5.43
0001	360	99.84	4.26	95.54	2.88	97.26	5.43
0002	355	99.97	3.63	97.72	5.15	99.8	6.01
0004	351	99.56	5.51	98.2	3.99	97.61	4.59
0005	364	99.36	4.36	94.21	4.2	95.28	5.34
0006	370	99.32	3.82	91.16	5.41	90.72	5.48
0007	365	97.93	3.93	95.16	3.47	96.37	4.35
0008	357	99.93	5.1	99.74	4.23	99.94	5.62
0008	359	99.6	4.11	91.46	5.76	85.63	5.72
0009	362	99.47	3.97	97.66	6.45	93.1	4.36
Prom. Total	**359.7**	**99.406**	**4.275**	**95.927**	**4.759**	**95.247**	**5.233**

Nota: En esta tabla se muestran los resultados de la prueba 5 con diferentes accesorios en el rostro, a una distancia fija, además se obtiene el tiempo de respuesta de la identificación, se incluye el promedio obtenido en cada una de las variables mencionadas.

En las siguientes Figura 115 y Figura 116 se muestran las columnas agrupadas en los cuales se observa de mejor forma el contenido de la Tabla 29.

Figura 115

Precisión del sistema (%) del reconocimiento facial en la prueba 5

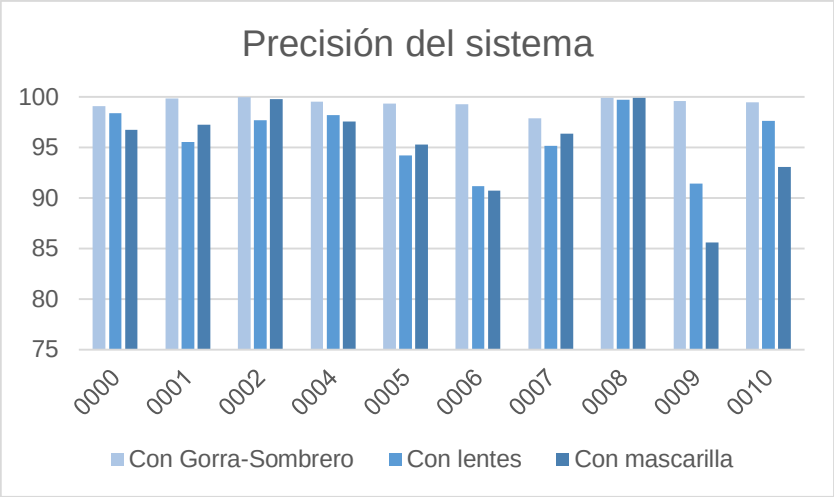
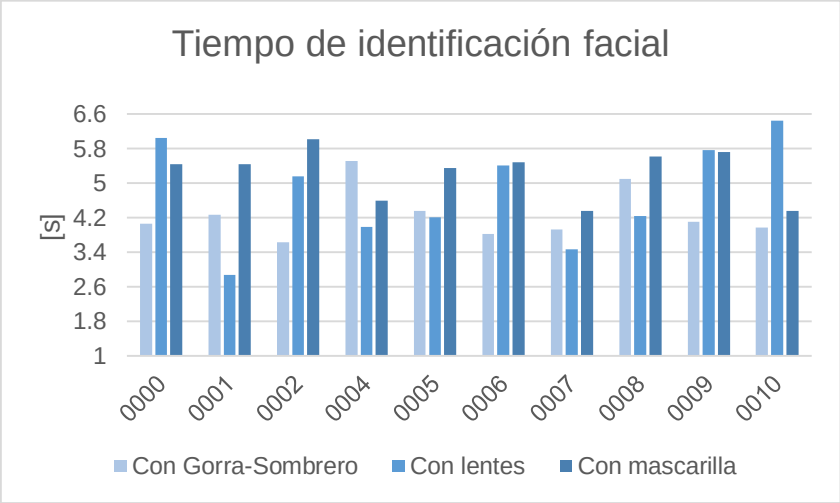


Figura 116

Tiempo de respuesta del sistema (seg) del reconocimiento facial en la prueba 5



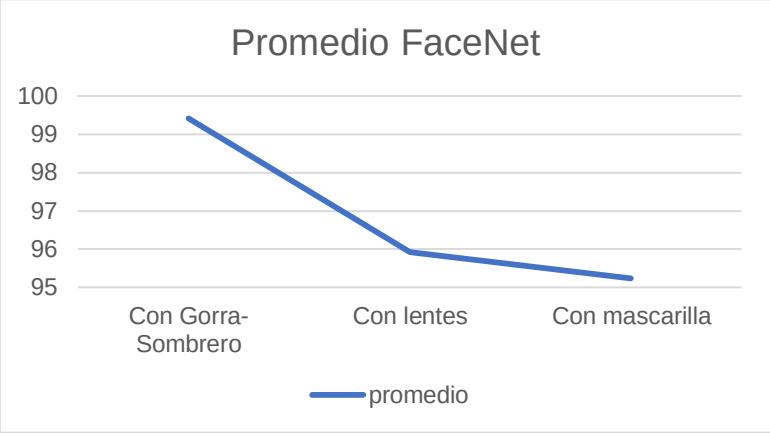
Análisis de la prueba 5

Esta prueba es de suma importancia, debido a que se necesitaba conocer si el sistema es capaz de identificar a los usuarios, incluso cuando tuvieran objetos en su

rostro, en la Tabla 29 se obtuvo el promedio del rendimiento del sistema con esta prueba y se puede observar en la Figura 117 en detalle este aspecto:

Figura *117*

Promedio de identificación (%) con accesorios en el rostro

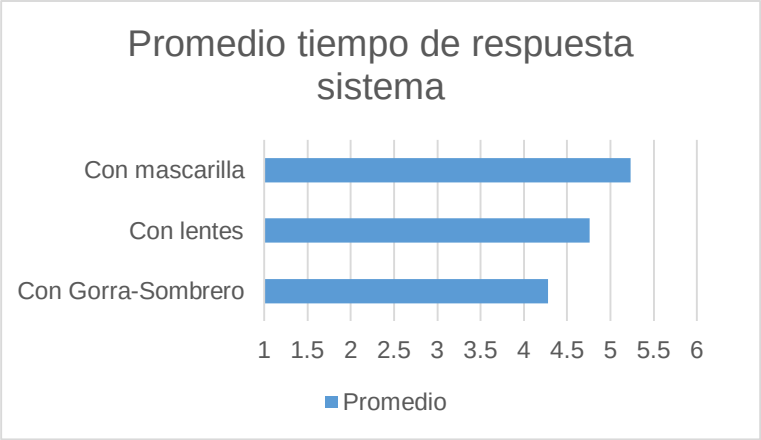


Se observa en la Figura 117, claramente como tiende a bajar el rendimiento conforme los usuarios se colocan diversos obstáculos en el rostro, dónde accesorios como la gorra o sombrero, el sistema tiene un elevado rendimiento aproximado al 100%, a diferencia de otros accesorios como lentes e incluso mascarilla, dónde se observa claramente un descenso cercano al 95%, pero siendo este aspecto aún muy bueno ya que cumple el objetivo primordial que es reconocer al usuario.

Como se viene explicando se debe también evaluar el tiempo que toma cada identificación, sumado a esto se obtiene un promedio tal como se realizó en la tabla, dando los resultados en la siguiente Figura 118.

Figura 118

Promedio del tiempo de respuesta (seg) del sistema del reconocimiento facial en la prueba



Se observa en la Figura 118, que existe un menor tiempo cuando el usuario tiene una gorra o sombrero que le cubre parte del rostro, el sistema lo identifica pero demora un promedio de 4.5 s, a diferencia de si el usuario se coloca lentes o se quiere una hacer una prueba extrema como colocarse una mascarilla en su rostros, el sistema se demora en identificarlo, pero finalmente cumple el objetivo que es reconocer a la persona.

Prueba 6

Finalmente culminan las pruebas midiendo si el modelo del algoritmo de vida detecta a los usuarios con fotos impresas de varios usuarios, imágenes del celular. En la Tabla 30 y Figura 119 se detalla cómo se realizó esta prueba.

Tabla 30

Prueba del algoritmo de vida

Prueba 6	**Detalle**
5 Intentos de registro	Se coloca una imagen del teléfono móvil o tamaño del rostro al frente de la cámara de un usuario registrado,

Figura 119

Prueba de autentificación del usuario con el modelo antispoof

Izquierda: Imagen del celular. *Derecha: Imagen tamaño real del rostro*

Los detalles de la prueba se muestran a continuación en la

Tabla ***31***:

Tabla 31

Resultados de autenticación facial de los usuarios registrados

	Identificación rostro falso									
Pruebas	**Intento 1**		**Intento 2**		**Intento 3**		**Intento 4**		**Intento 5**	
Usuarios	**Si**	**No**	**Si**	**No**	**Si**	**No**	**Si**	**No**	**Si**	**No**
0000		x		x		x		x		x
0001		x		x		x		x		x
0002		x		x		x		x		x
0003		x		x		x		x		x
0004		x		x		x		x		x
0005		x		x		x		x		x
0006		x	x		x			x		x
0007		x		x		x		x		x
0008		x		x		x		x		x
0009		x		x		x		x		x
0010		x		x		x		x	x	
Prom. Total	**0**	**100**	**9.09**	**90.91**	**9.09**	**90.91**	**0**	**100**	**9.09**	**90.91**

Nota: En esta tabla se muestran los resultados de la prueba 6 en el cual se pretende identificar al usuario ya sea con la foto del celular o una imagen del tamaño del rostro, se incluye el promedio obtenido de los intentos por registrar el rostro.

Matriz de confusión

Una matriz de confusión es instrumento que permite observar los datos que el modelo entrega cuando ya ha sido entrenado, se puede analizar el rendimiento y desempeño del modelo ya sea de aprendizaje automático o profundo. Las columnas de la matriz son el número de predicciones obtenidas de una o varias clases que presente el modelo, en cambio las filas representan el índice de aciertos y errores que presenta el modelo al momento de entrenarlo (Barrios Arce, 2019).

Para esta prueba se analizará el algoritmo que permite determinar si un usuario registrado que está realizando su identificación es real o solo es una imagen del usuario registrado en el sistema de identificación facial. Para esta prueba del grupo de 11 usuarios registrados se tiene las siguientes características:

- 8 usuarios se identificaron como rostro real colocándose frente a la cámara,
- 2 usuarios se identificaron cómo rostro falso con una imagen frente a la cámara,
- 1 usuario se identificó como rostro verdadero con una imagen.
- Ningún usuario se identificó como rostro falso a pesar de estar frente a la cámara.

Todas estas pruebas se realizaron en 5 intentos de registro. El algoritmo de vida de aprendizaje profundo permite determinar cuál es el % de acierto en los intentos de autenticación facial. Las 4 opciones siguientes son las que conforman la matriz de confusión (en este caso al ser solo dos posibilidades: positivo o negativo se habla de una matriz binaria).

Por lo tanto y de acuerdo a la prueba se tendría lo siguientes cuatro opciones:

- Usuario que se identificó como rostro real y el modelo lo identificó como rostro real (+). Esto sería un verdadero positivo (VP).

- Imagen del usuario registrado que lo identificó como rostro falso y el modelo lo identificó como rostro falso (-). Este sería un verdadero negativo (VN).
- Usuario que se identificó como rostro real y el modelo lo identificó como rostro falso (-). Éste sería un falso negativo (FN).
- Imagen del usuario registrado que identificó como rostro real y el modelo lo identificó como rostro real (+). Este es un falso positivo (FP).

Ahora de forma más clara, se puede identificar la matriz donde se ubican los errores (cajas rojas).

Figura 120

Matriz de confusión binaria.

Nota: La figura muestra la matriz de confusión binaria, Tomado de *BigData: La matriz de confusión y sus métricas,* (Barrios Arce, 2019)

Métricas de la matriz de confusión

Exactitud

Conocida también como Accuracy (por sus siglas en inglés), representa el porcentaje de predicciones correctas frente al total. Se calcula a continuación.

Datos

VP=8

VN=2

FP=1

FN=0

$$\textbf{Accuracy} = \frac{(VP + VN)}{(VP + FP + FN + VN)}$$

$$\boldsymbol{Accuracy} = \frac{8 + 2}{11} = 0.9091 = 90.9\%$$

Precisión

Hace referencia a cómo se acerca el resultado de una predicción del valor verdadero (Barrios Arce, 2019). Se calcula de la siguiente manera

$$\boldsymbol{Presicion} = \frac{VP}{VP + FP}$$

$$\boldsymbol{Presicion} = \frac{8}{8 + 1} = 0.888 \ = 88.8\%$$

Sensibilidad

Llamado también Recall (traducido del inglés), esta es la medida que representa la tasa de verdaderos positivos donde se dice que es el equilibrio de casos positivos que fueron correctamente identificadas por el algoritmo (Barrios Arce, 2019).

Se calcula de la siguiente manera:

$$\boldsymbol{Recall} = \frac{VP}{VP + FN}$$

$$\boldsymbol{Recall} = \frac{8}{8 + 0} = 1 = 100\%$$

De este parámetro se agrega el FAR (False Accept Rate, traducido del inglés taza de falsa aceptación), donde se calcula de la siguiente manera:

$$\boldsymbol{FAR} = \frac{FP}{FP + VN}$$

$$\boldsymbol{FAR} = \frac{1}{1 + 2} = 0.333 = 33.3\%$$

Especificidad

Es la proporción entre los casos negativos bien clasificados por el modelo, respecto al total de negativos (Telefonia Tech, 2021). Se la conoce también como la tasa de verdaderos negativos y se calcula de la siguiente manera

$$\boldsymbol{Especificidad} = \frac{VN}{VN + FP}$$

$$\boldsymbol{Especificidad} = \ 2/(2+1) = 0.666 = 66.66\%$$

De este parámetro se añade también el FRR (False Reject Rate, traducido del inglés tasa de falsa rechazo), donde se calcula de la siguiente manera:

$$\boldsymbol{FRR} = \frac{FN}{VP + FN}$$

$$\boldsymbol{FRR} = \frac{0}{8+0} = 0 = 0\%$$

Resumen

Entonces a partir de estas métricas, se puede concluir que el algoritmo de vida - antispoof, tiene los siguientes resultados:

- **Accuracy** = 90.9%
- **Precision** = 88.8%
- **FAR** = 33.3%
- **FRR** = 0%

Estas características del modelo están un poco bajas en cuanto al algoritmo de vida se refiere, cabe mencionar que se requiere más datos para analizar las muestras, pero debido a la pandemia que azota nuestros días, es por ahora imposible ajustar nuevos usuarios y validar estos datos, pero queda como trabajo futuro para en un tiempo venidero poder mejorar el presente trabajo de investigación.

Con respecto a las métricas FAR y FRR, se concluye que para 5 intentos, de un total de 11 usuarios, se encontró 1 falsa aceptación y que no hay falso rechazo en ningún

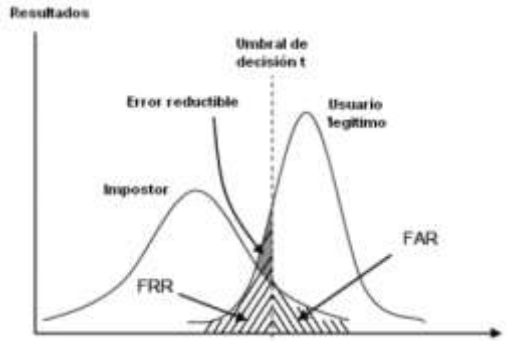
usuario registrado en el sistema de identificación facial, esto se debe también a que va ligado con el umbral (0.4) colocado por la métrica de la distancia del coseno del modelo de FaceNet. Entonces se concluye que:

- Si se tiene un valor de umbral bajo (0.4 es el mínimo), esto significa que el FAR aumentará y el FRR disminuirá, dando como resultado que el usuario registrado lo identifique pero también lo hará con usuarios no registrados.
- Por el contrario si se tiene un valor de umbral alto (bajo la métrica del 0.4), esto significa que el FAR disminuirá y el FRR aumenta, dando como resultado que el usuario registrado necesite más de una vez estar en frente de la cámara buscando que lo registre el sistema.

La necesidad de un compromiso en el valor del umbral que haga que los valores tanto de FAR, como de FRR, permitan funcionar al sistema de manera correcta, va de la mano con las necesidades de seguridad para que este valor aumente o disminuya, tal como se muestra en la Figura 121.

Figura 121

Umbral de decisión



Nota: La figura muestra el rendimiento de los sistema de identificación facial con un equilibro del FAR y FRR, Tomado de *Aplicaciones de la Biometría a la Seguridad,* (Sánchez Ávila, 2007)

Esta métrica de umbral decisión, va ligado también en cuanto agregar muchos usuarios al sistema de reconocimiento facial, ya que mientras más usuarios ingresen al

sistema, puede haber confusiones en cuanto a rasgos faciales de cada persona, que si bien son características únicas de cada individuo, esto puede significar un problema porque se tendría muchos falsos positivos por la similitud de rasgos que existiría además el rendimiento asi como la precisión del sistema comenzaría a disminuir (Sánchez Ávila, 2007).

En resumen, el umbral de la distancia del coseno, proporcionado por el modelo FaceNet (por definición es 0.4), al agregar muchos usuarios, se debe disminuir este parámetro para ser más preciso si se tiene una gran cantidad de usuarios y de esta manera evitar falsos positivos.

Costo computacional del sistema

En esta prueba se realiza un estudio del consumo del sistema y cuantos sistemas de identificación facial a la vez se podrían correr en paralelo con las características del computador mostrado en la Figura 122.

Figura 122

Características del computador

El sistema está distribuido de la siguiente manera:

- 1 control de identificación facial para el ingreso-salida de los usuarios

Para obtener un cálculo correcto de cuanto consume el sistema de reconocimiento facial, se debe conocer cuántos procesos requiere, sumado al número de núcleos que

cuenta el procesador del servidor para correr los algoritmos de aprendizaje profundo, adicional a esto cuanta memoria RAM como VRAM de la GPU requiere por proceso.

Entonces se dedujo por pruebas y se ha llegado a la conclusión que el sistema de reconocimiento facial arroja los siguientes datos en cuanto a núcleos y memoria RAM se refiere, tal como se puede observar en la Figura 123:

Figura 123

Consumo de núcleos del procesador y memoria RAM

A continuación se realiza el cálculo:

Datos

$SRF = 1$

$\#N = 4$

$\#uP = 8$

$CS = 4\ hilos$

$TS = ?$

- SRF = # de sistemas reconocimiento facial requeridos
- #N: Número núcleos del procesador
- #uP: Número de hilos del procesador
- CS: Consumo por hilos de un sistema de reconocimiento facial
- TS: Total de consumo del sistema de reconocimiento facial levantado

Una conclusión del sistema de reconocimiento facial se determinó que es directamente proporcional al número de hilos que tiene el procesador, es decir si el usuario necesita más puntos para instalar el sistema, necesita más hilos del procesador, sabiendo esto tenemos el siguiente cálculo:

$$TSRF = SRF * CS$$

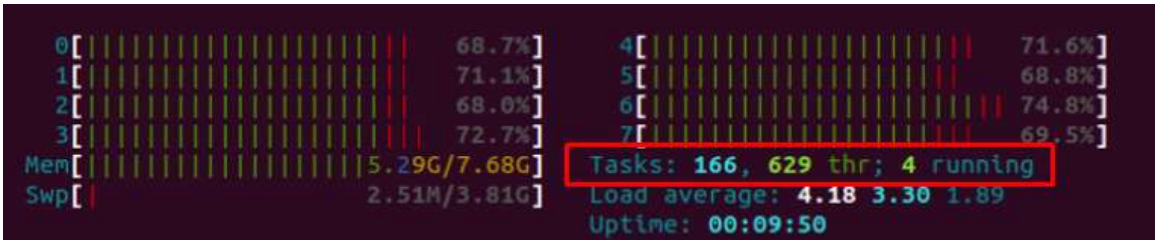
$$TS = 1 * 4$$

$$TS = 4\ hilos$$

Lo cual corrobora con la siguiente Figura 124;

Figura 124

Consumo de hilos del procesador



Consumo de la GPU

En la siguiente Figura 125 se muestran las características de la GPU usada en el diseño del sistema de reconocimiento facial.

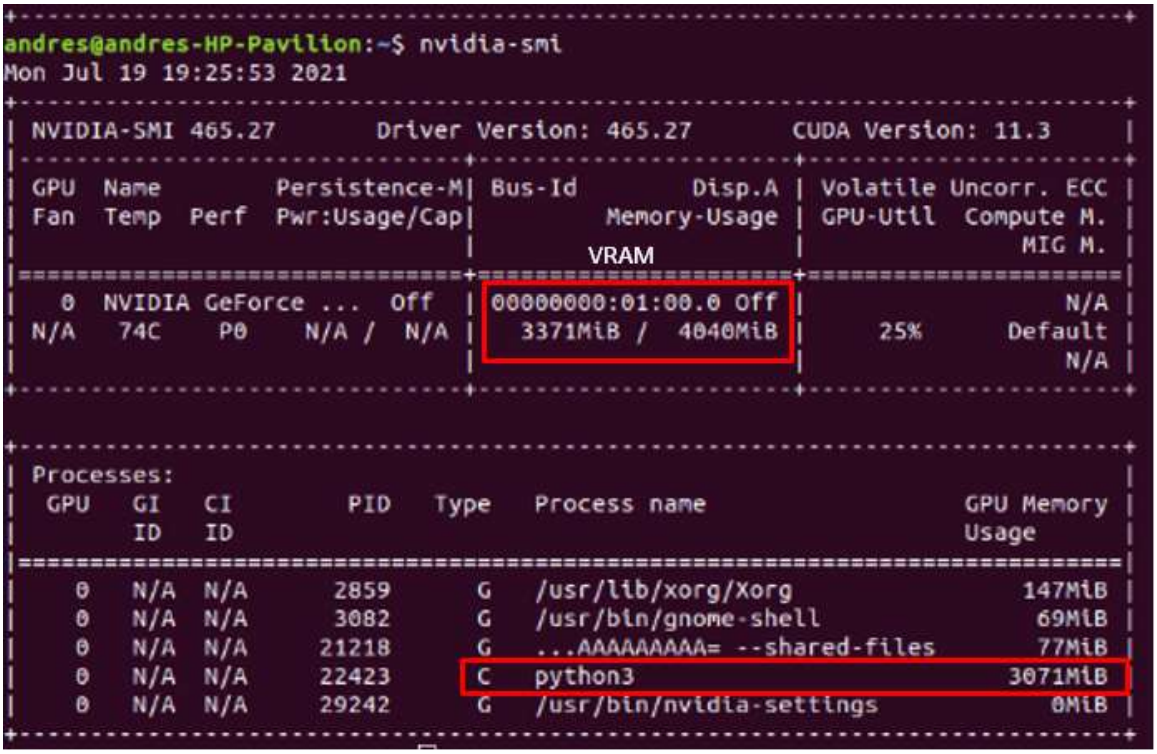
Figura 125

Datos de la tarjeta de video

Con respecto al consumo de la GPU, en la Figura 126 el consumo de cada servicio de reconocimiento facial levantado en el computador.

Figura 126

Consumo de la memoria VRAM de la GPU



Se determinó que el sistema de reconocimiento facial consume un aproximado de VRAM detallado a continuación:

- Sistema de identificación facial unido al sistema de entrenamiento individual levantado: 3071MB

Datos

$CGPU = 3071MB --> 3GB$

$TCGPU = ?$

- CGPU: Consumo de VRAM de GPU por cada Sistema de reconocimiento facial levantado en el servidor
- TCGPU= Total consumo de GPU

Entonces teniendo un panorama más claro, se puede deducir que el sistema también depende de la memoria VRAM de la GPU, y es directamente proporcional, esto

significa que para agregar más puntos de instalación del sistema de reconocimiento facial se requiere más VRAM de la GPU, pero la variable del sistema de entrenamiento no se la tomara en cuenta porque no es un proceso que se lo realice de forma continua con respecto al sistema de reconocimiento facial, entonces agregando un par de variables adicionales al cálculo, entonces quedaría asi.

$$TCGPU = PRF * CGPU$$

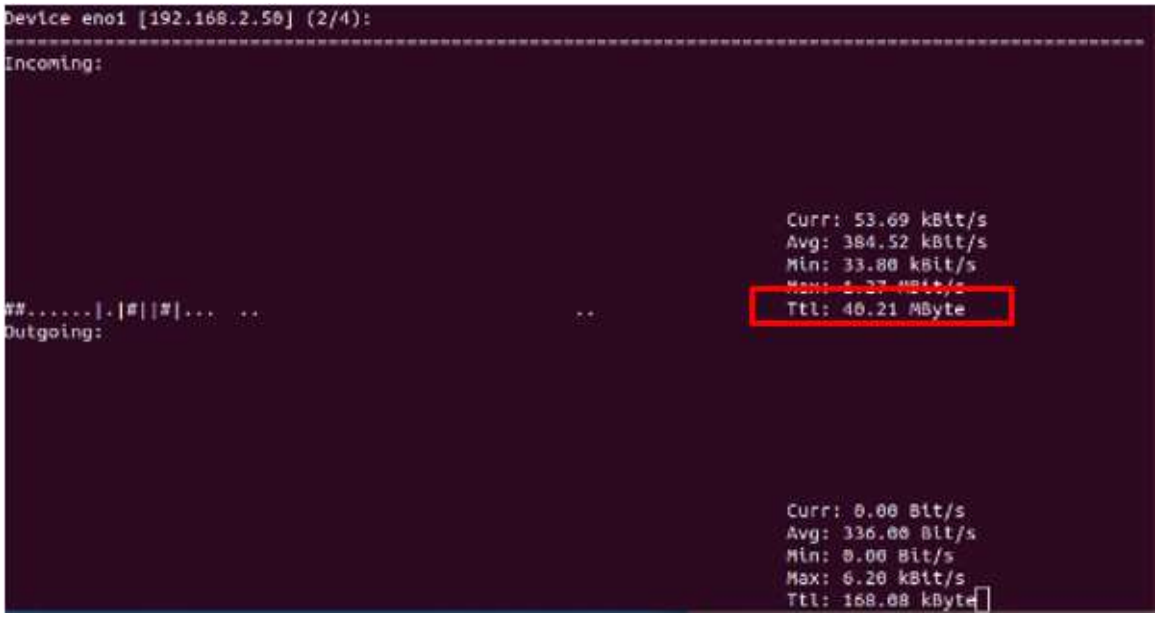
$$TCGPU = 1 * 3 = 3GB$$

Consumo de red

El sistema tiene un aproximado de consumo de red que bordea los 40MBps, esto quiere decir que se necesitaría una red promedio de 50 a 100MBps con un ancho de banda elevado, con tecnología de fibra óptica o categoría 6e si es por medio de transmisión cableado.

Figura 127

Consumo de la red con el servicio levantado en el servidor



Resultados

El sistema de reconocimiento facial para 1 estación de reconocimiento facial requiere de 2 núcleos del procesador, y 4 hilos para trabajar en paralelo para que el sistema no colapse, además se necesita aproximadamente 3GB y la GPU tiene máximo 4GB de VRAM.

En este punto se recomienda no trabajar con la máxima capacidad del procesador, ya que debería llegar a un 80% de su capacidad y además se acortaría la vida útil del hardware. Por lo tanto el sistema puede abarcar máximo 1 estación, ya que por características del computador no podría soportar más estaciones trabajando en paralelo, si se requiere agregar más sistemas de identificación se debe mejorar el micro procesador y GPU con mayor capacidad de memoria VRAM, para balancear la carga del procesamiento de los algoritmos de aprendizaje profundo.

Capítulo VI

Conclusiones y Recomendaciones

Conclusiones

- Se diseñó un sistema de control de acceso de personal por medio de reconocimiento facial donde se aplicó técnicas de aprendizaje profundo tanto para la detección como identificación del usuario registrado.
- Como se observa los diversos algoritmos de reconocimiento facial que existen en el campo del aprendizaje profundo, donde el modelo denominado FaceNet obtiene mejores respuestas en cuanto a las métricas de rendimiento, acurracy y precisión se refiere.
- Los resultados de los experimentos para 70 pruebas realizadas se demuestran que el sistema de reconocimiento facial arroja un elevado rendimiento en cuanto a identificación facial se refiere, cuando el usuario registrado se coloca con vista al frente de la cámara, cercanos al 100% y además esto se logró con tan solo 2 fotos por cada usuario registrado.
- Se concluye que hay una respuesta de identificación más rápida con respecto a una vista frontal, con tiempos de 2 segundos, de una vista de perfil del usuario registrado, pero en cuanto a la identificación, posee un alto rendimiento porcentual en reconocer al usuario, valores aproximados al 97%.
- Se comprobó que en diferentes entornos, cambios de luz y distancias el algoritmo, identifica al usuario y aunque puede existir un aumento en cuanto al tiempo de identificación tal como se menciona en la pruebas, el sistemas logra el objetivo el cual es reconocer al usuario registrado.
- Se puede observar que el sistema es capaz de identificar al usuario incluso si el individuo tiene accesorios en su rostro tales como lentes, una gorra o sombrero, incluso si tiene colocado una mascarilla en su rostro, todo esto con un nivel de lúmenes adecuado.

- Se pudo apreciar que si el usuario no está registrado en el sistema de identificación facial no lo reconoce, este aspecto es fundamental para disminuir la tasa de falsa aceptación (FAR), y limitar la detección de falsos positivos.
- Cómo conclusión se tiene que el algoritmo de vida presenta un rendimiento con métricas de accuracy del 90% y una precisión del 88.8%, siendo un resultado más que aceptable debido a que se utilizó solo software para el desarrollo de esta función de seguridad y brindando robustez al sistema de reconocimiento facial.
- Se evidencia como el sistema consume muchos recursos computacionales y esto se debe a como se explicó en el capítulo 2, el campo del aprendizaje profundo requiere mucha tecnología para el desarrollo e implementación en producción.
- Se concluye que es un sistema desarrollado de costo económico, adaptado con las cámaras de video vigilancia que puede trabajar sin ningún problema, además es personalizado, intuitivo al usuario y adaptable, para las diversas necesidades en cuanto a control de acceso biométrico se refiere.

Recomendaciones

- Al momento de ajustar a un nuevo usuario al sistema de identificación facial, es recomendable que tenga una buena foto de perfil frontal, con una resolución mínima de 3 megapíxeles, ya que será la única foto comparativa que el sistema tendrá para clasificar de los otros usuarios registrados, asi mismo para que el sistema lo identifique incluso con mascarilla, debe tomarse una foto con este accesorio en su rostro.
- Se recomienda realizar la transmisión del video por cable, esto se debe a que por medios inalámbricos reduciría a más de la mitad la velocidad de transmisión de los datos y el reconocimiento facial no se lo ejecutaría en tiempo real teniendo retrasos de video muy prolongados.
- Si en un futuro se quisiera implementar este proyecto en producción se debe colocar un buen sistema de disipación de calor, esto más que nada para prolongar la vida útil del hardware, ya que en el presente proyecto por cuestiones de

portabilidad del computador se lo desarrolló en un computador portátil, y el sistema elevaba las temperaturas del computador, pero con los sistemas de refrigeración actuales se soluciona este inconveniente.

- Es importante limpiar el historial del navegador ya que eso permite que la latencia en la captura de fotogramas sea la menor posible para lograr una recepción de video en tiempo real.
- El sistema actualmente realiza un reconocimiento único, se sugiere para una identificación múltiple, cambiar a modelos de aprendizaje profundo tales como MXNet, o DeepFace, además se debe mejorar el hardware en cuanto a CPU y GPU se refiere (ir por la serie 30 de Nvidia)
- En caso de que existiera una tasa de falsas detecciones elevada, es recomendable modificar con la métrica de la distancia o umbral de detección, buscando siempre un equilibrio para que el sistema de identificación facial funcione correctamente.

Trabajos Futuros

- Como trabajos futuros, para darle un grado más elevado de seguridad en cuanto a que el usuario registrado este presente frente a la cámara del sistema de reconocimiento facial se refiere, se podría optar por una cámara extra, y mediante algoritmos de detección de profundidad validar este aspecto (cámaras estéreo).
- En el presente proyecto de investigación no se usó una base de datos de uso profesional, pero se podría incluir en el sistema, para darle mayor robustez en cuanto a seguridad, una sugerencia podría ser utilizar Django, es cual es de uso gratuito y de código abierto unido con lenguaje de Python manejable para base de datos.
- Una limitación del sistema es el tema del paso de los años del usuario ya que va modificando sus rasgos faciales y esto dificultará para la identificación, se podría estudiar un algoritmo más robusto que vaya modificando este aspecto o a su vez como una solución rápida, ajustar el sistema con una nueva foto.
- En el presente proyecto se utiliza una cámara IP para capturar las imágenes que serán enviadas al sistema, para mejorar la velocidad de la transmisión de datos se puede tener una cámara del tipo CSI, esto ayudaría a optimizar este aspecto disminuyendo la latencia del sistema.
- Se puede incluir al sistema de identificación el estado de ánimo de la persona, por medio de los gestos, esto serviría para obtener medidas de trazabilidad y de esta manera observar en los datos qué nivel de estrés tiene el usuario con respecto a la función que desempeña en la empresa.

Referencias Bibliográficas

Alvarado Zambrano, B. E., Landeta Rodríguez, C. J., Sánchez Jiménez, J. L., & Castro Arreaga , R. A. (2010). *Análisis, diseño e implementación de un sistema de inmótica, para el edificio administrativo de la facultad técnica U. Guayaquil.* Guayaquil: U Catolica.

Amat Rodrigo, J. (16 de junio de 2021). *cienciadedatos.net*. Obtenido de www.cienciadedatos.net/documentos/34_maquinas_de_vector_soporte_support_vector_machines

Freire, E., & Silva, S. (2019). *Redes neuronales.* Bootcamp AI.

Huang, G., Ramesh, M., Berg, T., & Learned-Miller., E. (2007). *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments.* Massachusetts-Usa: University of Massachusetts.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *25th Conference International Neural Information Proccesin System* (págs. 1097-1105). New York-Usa: Association for Computing Machinery.

Sucar, L. E., & Gómez, G. (2014). *Visión Computacional.* Puebla-Mexico: Instituto Nacional de Astrofísica, Opticá a y Electrónica.

Zeiler, M., & Fergus, R. (2013). *Visualizing and Understanding Convolutional Networks.* New York-Usa: U. Cornell.

Abadi, M., Agarwal, A., Barham , P., Brevdo, E., Chen, Z., Citro, C., . . . Devin, M. (2015). "TensorFlow: Large-scale machine learning on heterogeneous systems". Google.

Acuña Escobar, D. A. (2019). *Visión artificial aplicada a la detección e identificación de personas en tiempo real.* Quito: EPN.

Adamczyk, J. (2021). *Robust face detection with MTCNN.* India: towards data science.

Aguirre Balcázar, N. B., & Bautista Toapanta, C. M. (2016). *Diseño, acondicionamiento e implementación de un sistema de adquisicin y tratamiento de imágenes y rediseño de sistema mecánicos para una plataforma CNC de corte por láser.* Sangolqui-Ecuador: Espe.

Albornoz, M. C., Berón, M., & Montejano, G. (2017). *Interfaz Gráfica de Usuario: el Usuario como Protagonista del Diseño .* San Luis-Argentina: Universidad de San Luis.

Alegre, E., Pajares, G., & de la Escalera, A. (2016). *Conceptos y Metódos en Visión por Computador.* España: CEA.

Amos, B., Bartosz , L., & Satyanaray, M. (2016). *OpenFace: A general-purpose face recognition and library with mobile applications.* Pensilvannia: CMU School of Computer Science.

Andrew , G., Howard , M., Zhu , B., & Dmitry, K. (2017). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.* Usa: Google Inc.

Apple. (17 de Diciembre de 2017). Face ID Security. Retrieved from. Silicon Valley, California, Usa.

Aprende Machine learning. (06 de Junio de 2021). *Aprendemachinelearning.com*. Obtenido de https://www.aprendemachinelearning.com/como-funcionan-las-convolutional-neural-networks-vision-por-ordenador/

Arias Melendres, B. G. (2020). *Desarrollo de un sistema prototipo de control de acceso al Laboratorio de Comunicaciones Unificadas de la Facultad de Ingeniería Eléctrica y Electrónica (FIEE) de la EPN empleando reconocimiento facial.* Quito: EPN.

Avilés Pincay, A. I., & Barcia, J. (2016). *Elaborar un clasificador para el procesamiento digital de imágenes.* Guayaquil: U. Guayaquil.

Barrios Arce, J. I. (2019). *La matriz de confusión y sus métricas.* Barcelona-España: Big Data.

Barrios, J. (06 de Junio de 2021). *www.juanbarrios.com*. Obtenido de https://www.juanbarrios.com/redes-neurales-convolucionales/

BBC Mundo Tecnología. (2017). *3 métodos más usados por los hackers para secuestrar tu cuenta de Google.* Londres: BBC News.

Biometrica. (2015). *Metodos biometricos.* Buenos Aires: Biometria.

Bochkovskiy A., C.-Y. H.-Y. (2020). "YOLOv4: Optimal Speed and Accuracy of Object Detection". *Cornell , 4*, 1-17.

Calvo, D. (2017). *Arquitectura de red neuronal convolucional.* Diego Calvo.

Cao, Q., Shen, L., Xie, W., Parkhi, M., & Zisserman, A. (2018). *VGGFace2: A dataset for recognising face across pose and age, International Conference on Automatic Face and Gesture Recognition.* Londres: Oxford.

Caparrini, F. (2019). *Redes Neuronales: una visión superficial.* Madrid: CS-US.

Cedeño Navarrete, J. R., & Párraga Vera, C. L. (2017). *Sistema biométrico de control de acceso para el laboratorio de cómputo de la Unidad Educativa Francisco González Álava.* Manabi-Ecuador: ESPAMMFL.

Cerron Zarcco , J. F. (15 de junio de 2015). Metodos Cuantitativos de Colorimetria, Reflectancia y Dureza.

Chasiquiza Molina, E. O. (2008). *Reconocimiento digital de imágenes aplicado a rostros humanos basado en PCA utilizando redes neuronales.* LATACUNGA: ESPE.

Chauca Vera, B. A. (2020). *Seguimiento y búsqueda de objetivos en entornos complejos usando micro vehículos.* Sangolqui: Espe.

Data Science Blogathon. (2021). *Face Detection and Recognition capable of beating humans using FaceNet.* Analitycs Vidhya.

Developer Mozilla. (09 de junio de 2021). *developer.mozilla.org*. Obtenido de developer.mozilla.org/es/docs/orphaned/Web/Guide/HTML/HTML5

Domingo Muñoz, J. (09 de Junio de 2021). *OpenWebinars*. Obtenido de openwebinars.net/blog/que-es-flask/

Dong , Y., Zhen , L., Shengcai , L., & Stan , Z. (2014). *CASIAWebface: Learning Face Representation from Scratch.* China: CASIA.

efecto digital. (09 de junio de 2021). *www.efectodigital.online*. Obtenido de ww.efectodigital.online/single-post/2018/04/18/dise%C3%B1o-de-interfaz-de-usuario-ui

efectodigital. (12 de junio de 2021). *www.efectodigital.online*. Obtenido de www.efectodigital.online/book-a-room

El Comercio. (26 de Julio de 2021). El Centro de Investigaciones de la UEES identifica seis nuevos casos de la variante Delta en Ecuador. Quito, Pichincha, Ecuador.

El universo. (26 de julio de 2021). Los casos de la variante delta del coronavirus en Ecuador aumentaron a 47, según el Gobierno. Quito, Pichincha, Ecuador.

EPN. (09 de junio de 2021). *epn.edu.ec*. Obtenido de www.epn.edu.ec/streaming/

Fernández Castilla, N. L. (2013). Visión Artifical Avanzada. *Introducción a la Visión Artificial*, 1-50.

Fernández Ruíz , M., Angós Ullate, J., & Salvador Oliván, J. (2006). Interfaces de usuario: diseño de la visualización de la información como medio para mejorar la

gestión del conocimiento y resultados obtenidos por el usuario. *V Congreso isko* (pp. 1-12). Zaragoza-España: U. Zaragoza.

Formadores IT. (09 de junio de 2021). Curso Flask. Madrid, España.

Gavilan, I. (2020). *Catálogo de componentes de redes neuronales (II): funciones de activación.* Mexico: bluechip.

Gege Bua, J. (03 de agosto de 2017). Reconocimiento del valor FAR del reconocimiento facial. *CSDN*, págs. 3-5.

Geitgey, A. (2020). *Face Recognition.* Usa: ageitgey.

Giraldo de la Caridad, L. R., & Silvia Margarita , V. B. (2017). *La inteligencia artificial en la educación superior. Oportunidades y amenazas.* Guayaquil: UIDE-INNOVA.

González, S., & Cajamarca, J. (2014). *Proyecto de creacion de un canal de tv online para la UPS sede Cuenca.* Cuenca: UPS-Cuenca.

Google. (2019). *Real-time Facial Surface Geometry from Monocular Video on Mobile GPUs.* New York: U. Cornell.

Guo , J., Deng, J., Jiankang, Xue, Niannan, & Zafeiriou, S. (2018). *Stacked Dense U-Nets with Dual Transformers for Robust Face Alignment.* BMVC.

Gutiérrez, J. M. (218). *Introducción a las redes neuronales.* Santander, España: U. Cantabria.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation.* Macmillan.

HikVision. (2021, Junio 09). *hikvision.com*. Retrieved from www.hikvision.com/es-la/core-technologies/storage-and-bandwidth/h-265-plus/

Ibarra Flores, W. I. (2020). *Sistema de control de acceso mediante identificación y verificación facial fundamentado en algoritmos de aprendizaje automático y*

redes neuronales. Quito: Universidad de las Fuerzas Armadas ESPE. Carrera de Ingeniería en Electrónica y Telecomunicaciones.

iDenfy. (07 de junio de 2021). *idenfy*. Obtenido de idenfy.com: www.idenfy.com/blog/spoofing-attack-prevention/

INECEL. (1973). Codigo ELectrico Ecuatoriano. *Instituto Ecuatoriano de Electrificacion, 1*(3), 35-56.

interempresas. (06 de Junio de 2021). *interempresas.net*. Obtenido de https://www.interempresas.net/MetalMecanica/Articulos/347471-Los-conceptos-de-Machine-Learning-y-Deep-Learning-en-la-industria.html

Jastrow, J. (1899). The mind's eye. Popular Science Monthly. Usa: Usa.

Josemon, F., Irshad , P., Sunil, R., & Sasikumar, S. (2017). *Attendance monitoring system using Face Recognition.* Thrissur: Computer Science and Engineering in Government Engineering College.

Kumar, D. (2019). *Introduction to FaceNet: A Unified Embedding for Face Recognition and Clustering.* India: Medium.

Kunal, B. (2020). *Face Recognition: Real-Time Face Recognition System using Deep Learning Algorithm and Raspberry Pi 3B.* India: Medium.

Li, Y., Li, M., Lin, M., Wang, N., Wang, M., Xiao, T., . . . Zhang, Z. (2015). *MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems.* China: Workshop on Machine Learning Systems.

Lopez, J. (31 de mayo de 2018). Cómo ‘hackear’ una cara: del reconocimiento facial. *Open Mind BBVA*, págs. 4-8.

Mainenti, D. (2017). *User Perception of Apple's Face ID.* Usa.

MediaPipe. (2020). *MediaPipe*. (MediaPipe Face Mesh) Retrieved 03 17, 2021, from google.github.io/mediapipe/solutions/face_mesh

Michael Schrartz, C. G. (2016). A new Way to see the light. *IEEE Industria Aplications Magazine* , 1-8.

Minivision. (30 de mayo de 2016). *ai.minivision.cn*. Obtenido de ai.minivision.cn/#/mainIndex

Morales Caporal, M. A. (2020). *Procesamiento Digital de Imagenes.* Mexico: UPTx.

Mordvintsev, A., & Abid , K. (2013). *Opencv:Face Detection using Haar Cascades.* Opencv.

Mur Igualada, G. (2015). *Inferencia de la Respuesta Afectiva en los rostros de los Espectadores de Un Video.* Madrid: U. Carlos III .

Nayak, S. (2019). *Training YOLOv3 : Deep Learning based Custom Object Detector.* Delhi-India: LearnOpencv.

Nielsen, M. (2015). How the backpropagation algorithm works. En M. Nielsen, *Neural Networks and Deep Learning* (págs. 39-57). Usa: eBook.

Obando Cisneros, D. I. (2019). *Implementación de un control de acceso biométrico mediante reconocimiento facial.* Quito: Universidad de las Fuerzas Armadas ESPE. Carrera de Ingeniería en Electrónica, Automatización y Control.

OpenCV-Web Official. (12 de junio de 2021). *docs.opencv.org/*. Obtenido de https://docs.opencv.org/master/d9/df8/tutorial_root.html

Paper With Code. (07 de junio de 2021). *paperswithcode.com*. Obtenido de https://paperswithcode.com/sota/object-detection-on-coco

Pardo, C. J. (2018). *Detección de Rostros basado en Filtros Haar + Adaboost.* Wordpress.

Pérez León, E. V., & Rojas Arévalo, D. I. (2019). *Impacto de la inteligencia artificial en las empresas con un enfoque global.* Lima-Peru: UPCC.

Python. (13 de junio de 2021). *www.python.org/*. Obtenido de www.python.org/

Quevedo Gonzalez, J. (2017). *Investigacion y prueba de ciberdelito.* Barcelona-España: U. Barcelona.

recogtech. (06 de Junio de 2021). *www.recogtech.com*. Obtenido de www.recogtech.com/en/knowledge-base/security-level-versus-user-convenience

Ren, S., He, K., Girshick, R., & Sun, J. (2016). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.* University of Science and Technology of China-Microsoft.

Renotte, N. (2020). *Facial Landmark Detection.* Sydney, Australia: nicknochnack.

Rishabh. (2020). *Tele Stroke System for Stroke Detection.* THES.

RNDS. (2010). Compresión de video. *Tecnologia*, 110-116.

Robins , M. (2020). *The Difference Between Artificial Intelligence, Machine Learning and Deep Learning.* Usa: Intel.

Rodríguez, V. (30 de octubre de 2018). *Conceptos básicos sobre redes neuronales.* Propio.

Rojas, I. G. (27 de 05 de 2015). *hugarcapella.com*. (Manual de sistemas de puesta a tierra) Recuperado el 23 de 06 de 2019, de https://hugarcapella.files.wordpress.com/2010/03/manual-de-puesta-a-tierra.pdf

Rolando Caldas. (09 de junio de 2021). *rolandocaldas.com*. Obtenido de rolandocaldas.com/php/css3-basico-1-php-paso-a-paso

Rosebrock, A. (2017). *Facial landmarks with dlib, OpenCV, and Python.* Usa: pyimagesearch.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., . . . Fei, L. (2017). *ImageNet Large Scale Visual Recognition Challenge (ILSVRC).* Princeton-Usa: Universidad de Stamford.

Sagonas, C., Antonakos, E., Tzimiropoulos, G., Pantic, M., & Zafeiriou, S. (2016, junio 16). *Special Issue on Facial Landmark Localisation "In-The-Wild".* Sydney, Australia: IMAVIS. Retrieved from ibug.doc.ic.ac.uk/resources/facial-point-annotations/

Sánchez Ávila, C. (2007). *Aplicaciones de la biometría a la seguridad.* Madrid-España: Politecnica de Madrid.

Schroff, F., Kalenichenko, D., & Philbin, J. (2016). *FaceNet: A Unified Embedding for Face Recognition and Clustering.* Usa: Google Inc.

Serengil, S. I., & Ozpinar, A. (2020). Deepface. En *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)* (págs. 23-27). INDIA: IEEE.

Serengil, Sefik Ilkin. (2021). *Deep Face Detection with RetinaFace in Python.* India: Sefik.

Sirovich, L., & Kirby, M. (1987). *Low-dimensional procedure for the characterizatio of human faces.* Rhode Island-Usa: Division of Applied Mathemetics, Brown University.

Solución Ingenieril. (06 de Junio de 2021). *solucioningenieril.com/*. Obtenido de solucioningenieril.com/vision_artificial/etapas_de_un_sistema_de_vision#:~:text=Escenario%20a%20analizar%3A%20Es%20el,unidad%20donde%20pueda%20ser%20procesada.

Telefonia Tech. (23 de julio de 2021). *empresas.blogthinkbig.com*. Obtenido de empresas.blogthinkbig.com/como-interpretar-la-matriz-de-confusion-ejemplo-practico/

TensorFlow Org. (2015). *Implementa modelos de aprendizaje automático en dispositivos móviles y de IoT.* Usa: Google Brain. Obtenido de https://www.tensorflow.org/.

Tianqi , C., Mu , L., Yutian, L., Min , L., Naiyan , W., Minjie,, W., . . . Zheng , Z. (2015). *MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems.* Workshop on Machine Learning Systems.

Viola, P., & Jones, M. (2001). *Algoritmo de Viola-Jones.* Compaq.

Warden, P., & Situnayake, D. (2020). *Machine Learning with Tensorflow lite on Arduino and Ultra-Low-Power Micro Controller.* Usa: O'Reilly.

Web Dev Simplified. (2020). *Face Detection JavaScript.* Nebraska: Web Dev Simplified.

Wei , L., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Yang Fu, C., & C. Berg, A. (2016). *SSD: Single Shot MultiBox Detector.* Usa: UMichigan.

Williams, C. (10 de octubre de 2011). Logo JavaScript. USA.

Wimmer, M., & Doherty, M. (2011). The development of ambiguous figure perception: Vi. conception and perception of ambiguous figures. Toronto: Universidad de Toronto.

Yuahn, S. (6 de junio de 2017). Tasa de reconocimiento falso (FAR) Tasa de rechazo (FRR), TPR, FPR y curva ROC. *CSDN*, págs. 7-8.

Zeusess. (2019). *Detección de rostros cooperativa Zeusee.* GitHub.

Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. In *IEEE Signal Processing Letters* (pp. 1499-1503). China: IEEE.

Zhao, H., & Ram, S. (2004). Constrained Cascade Generalization of Decision Trees. *IEEE Transactions on Knowledge and Data Engineerin, 16*, 727-739.

Zhou, Y., Deng, J., Yu, J., Kotsia, I., Guo, J., & Zafeiriou, S. (2019). *RetinaFace: Single-stage Dense Face Localisation in the Wild.* Londres: Middlesex University London.

ZKTeco. (2019). *Dathasheet G4.* Shenzen-China: ZKTeco.

Anexos