



UNIVERSIDAD CÉSAR VALLEJO

FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE
SISTEMAS

Algoritmo basado en una red siamesa, pérdida del triplete,
k-vecinos más cercanos y en el aprendizaje de una sesión para
el reconocimiento facial

TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE:

Ingeniero de Sistemas

AUTOR:

Melgarejo Vasquez, Jhon Victor (orcid.org/0000-0002-9059-3553)

ASESOR:

Dr. Alfaro Paredes, Emigdio Antonio (orcid.org/0000-0002-0309-9195)

LÍNEA DE INVESTIGACIÓN:

Sistema de Información y Comunicaciones

LÍNEA DE RESPONSABILIDAD SOCIAL UNIVERSITARIA:

Desarrollo económico, empleo y emprendimiento

LIMA – PERÚ

2024



UNIVERSIDAD CÉSAR VALLEJO

**FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

Declaratoria de Autenticidad del Asesor

Yo, ALFARO PAREDES EMIGDIO ANTONIO, docente de la FACULTAD DE INGENIERÍA Y ARQUITECTURA de la escuela profesional de INGENIERÍA DE SISTEMAS de la UNIVERSIDAD CÉSAR VALLEJO SAC - LIMA ESTE, asesor de Tesis titulada: "ALGORITMO BASADO EN UNA RED SIAMESA, PÉRDIDA DEL TRIPLETE, K-VECINOS MÁS CERCANOS Y EN EL APRENDIZAJE DE UNA SESIÓN PARA EL RECONOCIMIENTO FACIAL", cuyo autor es MELGAREJO VASQUEZ JHON VICTOR, constato que la investigación tiene un índice de similitud de 11%, verificable en el reporte de originalidad del programa Turnitin, el cual ha sido realizado sin filtros, ni exclusiones.

He revisado dicho reporte y concluyo que cada una de las coincidencias detectadas no constituyen plagio. A mi leal saber y entender la Tesis cumple con todas las normas para el uso de citas y referencias establecidas por la Universidad César Vallejo.

En tal sentido, asumo la responsabilidad que corresponda ante cualquier falsedad, ocultamiento u omisión tanto de los documentos como de información aportada, por lo cual me someto a lo dispuesto en las normas académicas vigentes de la Universidad César Vallejo.

LIMA, 10 de Junio del 2024

Apellidos y Nombres del Asesor:	Firma
ALFARO PAREDES EMIGDIO ANTONIO DNI: 10288238 ORCID: 0000-0002-0309-9195	Firmado electrónicamente por: EALFAROP el 10-06- 2024 22:14:03

Código documento Trilce: TRI - 0758238





UNIVERSIDAD CÉSAR VALLEJO

**FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

Declaratoria de Originalidad del Autor

Yo, MELGAREJO VASQUEZ JHON VICTOR estudiante de la FACULTAD DE INGENIERÍA Y ARQUITECTURA de la escuela profesional de INGENIERÍA DE SISTEMAS de la UNIVERSIDAD CÉSAR VALLEJO SAC - LIMA ESTE, declaro bajo juramento que todos los datos e información que acompañan la Tesis titulada: "ALGORITMO BASADO EN UNA RED SIAMESA, PÉRDIDA DEL TRIPLETE, K-VECINOS MÁS CERCANOS Y EN EL APRENDIZAJE DE UNA SESIÓN PARA EL RECONOCIMIENTO FACIAL", es de mi autoría, por lo tanto, declaro que la Tesis:

1. No ha sido plagiada ni total, ni parcialmente.
2. He mencionado todas las fuentes empleadas, identificando correctamente toda cita textual o de paráfrasis proveniente de otras fuentes.
3. No ha sido publicada, ni presentada anteriormente para la obtención de otro grado académico o título profesional.
4. Los datos presentados en los resultados no han sido falseados, ni duplicados, ni copiados.

En tal sentido asumo la responsabilidad que corresponda ante cualquier falsedad, ocultamiento u omisión tanto de los documentos como de la información aportada, por lo cual me someto a lo dispuesto en las normas académicas vigentes de la Universidad César Vallejo.

Nombres y Apellidos	Firma
JHON VICTOR MELGAREJO VASQUEZ DNI: 71692510 ORCID: 0000-0002-9059-3553	Firmado electrónicamente por: JMELGAREJOVA el 10- 06-2024 20:51:47

Código documento Trilce: TRI - 0758237



DEDICATORIA

A mis padres Jhon Melgarejo Calderon y Elida Vasquez Castro, por su apoyo incondicional, y a mi familia por acompañarme en cada paso de mi carrera universitaria.

AGRADECIMIENTO

He de expresar mi profundo agradecimiento a todas aquellas personas que me han ayudado a llevar a cabo mi trabajo, sin ellas no hubiese sido posible.

Agradezco a mi asesor Emigdio Antonio Alfaro Paredes, cuya importancia ha sido vital en el desarrollo del mismo y me enseñó la importancia de la investigación.

Índice de contenidos

Carátula.....	i
Declaratoria de autenticidad del asesor.....	ii
Declaratoria de Originalidad del Autor.....	iii
Dedicatoria	iv
Agradecimiento.....	v
Índice de contenidos.....	vi
Índice de Tablas.....	vii
Índice de Figuras	viii
Resumen.....	ix
Abstract.....	x
I. INTRODUCCIÓN	1
II. METODOLOGÍA.....	17
III. RESULTADOS	21
3.1 Análisis descriptivo	21
3.2 Análisis inferencial.....	24
3.3 Prueba de Hipótesis	28
IV. DISCUSIÓN.....	32
V. CONCLUSIONES	36
VI. RECOMENDACIONES.....	37
REFERENCIAS.....	39
ANEXOS	44

Índice de Tablas

Tabla 1 . Estadística descriptiva para el tiempo de reconocimiento facial	21
Tabla 2 . Estadística descriptiva para la precisión de reconocimiento facial.....	23
Tabla 3 . Prueba de normalidad para el tiempo de reconocimiento facial.....	24
Tabla 4 . Prueba de Wilcoxon de muestras emparejadas para el tiempo de reconocimiento facial.....	29
Tabla 5 . Prueba de Wilcoxon de muestras emparejadas para precisión de reconocimiento facial.....	30

Índice de Figuras

Figura 1 . Funcionamiento de algoritmo	12
Figura 2 . Fórmula de pérdida del triplete.....	12
Figura 3 . Arquitectura de redes siamesas	14
Figura 4 . Diferencia de tiempo entre algoritmo propuesto y algoritmo de pérdida del triplete	22
Figura 5 . Diferencia de precisión entre algoritmo propuesto y algoritmo de pérdida del triplete.....	24
Figura 6 . Gráfico de normalidad de tiempo de reconocimiento facial con algoritmo del pérdida del triplete	25
Figura 7 . Gráfico de normalidad de tiempo de reconocimiento facial con algoritmo propuesto	26
Figura 8 . Gráfico de normalidad de precisión de reconocimiento facial con algoritmo del pérdida del triplete	27
Figura 9 . Gráfico de normalidad de precisión de reconocimiento facial con algoritmo propuesto	27
Figura 10. Gráfico de curvas de aprendizaje y pérdida de algoritmo propuesto.....	93
Figura 11. Gráfico de curvas de aprendizaje y pérdida de algoritmo del triplete	94

Resumen

La investigación realizada tuvo como objetivo principal determinar el impacto de la implementación de un Algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y en el aprendizaje de una sesión para el reconocimiento facial. Las teorías que se plantearon respecto a la investigación son que la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y en el aprendizaje de una sesión, reducirá el tiempo de entrenamiento de reconocimiento facial y además de ello aumentará la precisión del mismo. La investigación se realizó mediante un enfoque cuantitativo, con un diseño de tipo pre-experimental. Para el presente estudio se tomó como población los datos digitales del dataset ORL, que incluyó 400 imágenes de representaciones faciales y como muestra se utilizó 40 imágenes de representaciones faciales. El instrumento que se planteó fue la ficha de registro. Los resultados de la investigación mostraron que la implementación del algoritmo no tuvo un impacto positivo en el reconocimiento facial. No se logró reducir el tiempo de reconocimiento facial frente al algoritmo del triplete, obteniendo un resultado de 0.96525 milisegundos, en comparación con los 0.7025 milisegundos del algoritmo de pérdida del triplete. Además, la implementación del algoritmo propuesto no tuvo un impacto positivo en el aumento de la precisión en el reconocimiento frente al algoritmo del triplete, obteniendo un 92.5% frente al 95% del algoritmo de pérdida del triplete.

Palabras clave: Inteligencia artificial, Algoritmo, Identidad.

Abstract

The primary objective of the research was to determine the impact of implementing an algorithm based on a siamese network, triplet loss, k-nearest neighbors, and one-shot learning for facial recognition. The hypotheses proposed in the research are that the implementation of an algorithm based on a siamese network, triplet loss, k-nearest neighbors, and one-shot learning will reduce the training time for facial recognition and increase its accuracy. The research was conducted using a quantitative approach with a pre-experimental design. For this study, the population consisted of digital data from the ORL dataset, which included 400 facial representation images, and a sample of 40 facial representation images was used. The instrument proposed was a registration form. The research results showed that the implementation of the algorithm did not have a positive impact on facial recognition. It did not manage to reduce the facial recognition time compared to the triplet loss algorithm, achieving a result of 0.96525 milliseconds compared to 0.7025 milliseconds of the triplet loss algorithm. Furthermore, the implementation of the proposed algorithm did not have a positive impact on increasing the accuracy of recognition compared to the triplet loss algorithm, achieving 92.5% compared to 95% for the triplet loss algorithm.

Keywords: Artificial intelligence, Algorithms, Identity.

I. INTRODUCCIÓN

En esta sección se exploró la realidad problemática en el contexto del procesamiento en sistemas de reconocimiento facial, seguido del objetivo y la meta de desarrollo sostenible relacionados. Se detalla el problema de investigación, seguido por la exposición de la justificación de la investigación y la conceptualización del objetivo general y específicos. Posteriormente, se presentan los antecedentes internacionales y nacionales relevantes, seguidos de las teorías relacionadas y las hipótesis que guiarán este estudio.

Para las personas, el reconocimiento de rostros es una tarea bastante sencilla. Un claro ejemplo de esto es que los niños aprenden rápidamente a identificar el rostro de sus padres. Esto podría llevar a pensar que enseñar a una computadora a reconocer diversos rostros sería algo fácil, pero lamentablemente no es así.

Según Sun, Luo, Rong, He y Yu (2022) el reconocimiento facial ha tenido un impacto importante en muchos aspectos de nuestra vida cotidiana, como método de verificación, por ejemplo, en el teléfono celular para la verificación de pago, registro de empleados, etc. Como indican Berrahal y Azizi (2023), en los últimos años ha habido un aumento en aplicaciones inteligentes en diversos campos. Uno de los avances más recientes se da en el desarrollo de aplicaciones de reconocimiento facial.

De igual manera, en el estudio de Kumar, Teja, Bhargav, Satpute, Naveen y Kamble (2023) se indicó que, para que el reconocimiento facial tenga éxito, las computadoras necesitan una enorme cantidad de datos, a diferencia de una persona que puede reconocer a alguien con solo verla una sola vez. Según los autores, no es el caso de las computadoras, ya que requieren mucha información o datos para poder reconocer a una persona. De igual forma, indicaron que el reconocimiento facial se enfrenta a problemas como el requerir una gran cantidad de datos para el aprendizaje, también surgen otros problemas cuando no se entrenan los modelos de reconocimiento facial con la cantidad necesaria de datos.

Como es el caso en el estudio realizado por Xu, Zhang y Zhou (2023) quienes señalaron que el reconocimiento facial en entornos no controlados es desafiante debido a grandes variaciones que puedan existir, como, por ejemplo, el

reconocimiento facial en un entorno con fondo complejo; en algunos casos oclusión y, en otros casos se pueden encontrar escenarios donde existe una gran variedad de iluminación, pose y expresiones faciales. Características que pueden complicar el reconocimiento facial y requerir una mayor cantidad de datos para que el sistema pueda distinguir y realizar el reconocimiento facial en ambientes con escenarios complejos.

Hernandez-Diaz, Alonso-Fernandez y Bigun (2023) indicaron que una debilidad de los algoritmos de aprendizaje para el reconocimiento facial es entrenar los modelos para una nueva tarea. Esto presenta un desafío para sistemas que requieran emplear el reconocimiento con datos de usuarios actualizados en todo momento.

Como también indicaron Saxena y Varshney (2021), en su estudio sobre soluciones de seguridad para hogares inteligentes, quienes expusieron que por lo general la clasificación de imágenes requiere una gran cantidad de imágenes de cada clase para el entrenamiento y que, además, los usuarios en algunas situaciones necesitaran probar el modelo entrenado en nuevos ambientes o entornos que el modelo entrenado no reconocería previamente y, por ende no mostrarían resultados exactos. Para resolver dicho problema, indicaron los autores que sería necesario volver a entrenar el modelo de reconocimiento.

Teniendo en cuenta la realidad problemática de la investigación, el objetivo y meta de desarrollo sostenible al que se enfocó esta investigación en contribuir fue promover la innovación tecnológica y el desarrollo de infraestructuras resilientes y sostenibles mediante el desarrollo de algoritmos eficaces para mejorar el procesamiento en sistemas de reconocimiento facial.

A partir de la realidad problemática expuesta, se formuló el problema general, y de la misma manera, se formularon también los problemas específicos para esta investigación teniendo como problema general ¿Cuál será el impacto que tendrá la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales? Teniendo como primer problema específico: ¿Cuál será el impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y en el aprendizaje de una sesión para la reducción del tiempo de reconocimiento de imágenes faciales con respecto al

algoritmo de pérdida del triplete? También se contó con un segundo problema específico: ¿Cuál será el impacto de la implementación de un Algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para el incremento de precisión en el reconocimiento de imágenes faciales en base al algoritmo de Pérdida del triplete?

Asimismo, se proporcionó una justificación desde los enfoques teórico, tecnológico y económico. En cuanto a la justificación teórica de la investigación, se consideró aportar una herramienta tecnológica que pueda ser utilizada en la comunidad o por estudiantes en proyectos que tengan como finalidad resolver problemas computacionales relacionados al reconocimiento facial donde existe escasez de datos.

Como mencionaron Mohr y Frochte (2023), los sistemas de aprendizaje por inteligencia artificial enfrentan desafíos computacionales en la etapa de pre procesamiento y entrenamiento de modelos de inteligencia artificial, especialmente en entornos donde hay escasez de datos o pocos ejemplos para entrenar, y donde el sistema debe tomar decisiones para resolver problemas complejos. En su investigación los autores implementaron el aprendizaje de una sesión para identificar objetos que solo han sido vistos una vez y de esa forma resolver el problema de escasez de datos.

La base de la justificación tecnológica de esta investigación radica en la imperativa necesidad de perfeccionar la verificación e identificación facial, especialmente en situaciones caracterizadas por la escasez de muestras de imágenes y la limitada variabilidad de poses. Este desafío cobra una relevancia particular en escenarios del mundo real donde se pretende implementar sistemas de reconocimiento facial. Djamaluddin, Munir, Utama y Kistijantoro (2023) han resaltado que el reconocimiento facial en modelos de inteligencia artificial, que deben reconocer imágenes mediante el aprendizaje en una única sesión, plantea un desafío significativo para los sistemas de reconocimiento facial. Que esta investigación se propone a abordar.

La justificación económica de la investigación se apoya en la necesidad de reducir costos para entrenar modelos con pocas sesiones. Como indicaron Sun, Jiang, Dou, Li, Miao, Deng y Zhao (2023) en su artículo "*Invisible Backdoor Attack With Dynamic Triggers Against Person Re-Identification*", el aprendizaje profundo ha progresado

considerablemente y se ha desarrollado a tal grado que se ha utilizado en aplicaciones de reconocimiento de imágenes, y también indicaron que el éxito de un sistema de reconocimiento de imágenes se basa en la escala de los conjuntos de datos. Sin embargo señalaron que, una de las consideraciones más importantes a tener en cuenta es que el desarrollo de conjuntos de datos para entrenar modelos de inteligencia artificial para el reconocimiento de imágenes es costoso en tiempo y dinero, lo que resulta, según los autores en que una gran cantidad de desarrolladores opten por conjuntos de datos de terceros o de uso libre.

Por otro lado, en un estudio reciente, Jiménez-Bravo, Murciago, Mendes, Silva y De La Iglesia (2022) presentaron un sistema de reconocimiento facial y propusieron el desarrollo de un sistema que usa pocas sesiones para el aprendizaje. Como resultado de la implementación, se obtuvo que el sistema resolvió el problema de reconocimiento facial, y según los autores, se redujo el costo del sistema de reconocimiento facial y dicho sistema mantuvo una detección efectiva.

El objetivo general fue determinar el impacto que tendrá la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales. Además, se planteó un primer objetivo específico, el cual fue: "Determinar el impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para la reducción del tiempo de reconocimiento de imágenes faciales".

También se propuso un segundo objetivo específico, el cual fue: "Determinar el impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para la precisión del reconocimiento de imágenes faciales en base al algoritmo Pérdida del triplete".

La hipótesis general planteada fue si la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión incrementa la precisión y reduce el tiempo de reconocimiento de imágenes faciales con respecto al algoritmo de pérdida del triplete. En la investigación de Anil y Padma Suresh (2023, p. 8), se propuso un estudio que tuvo como objetivo reducir el tiempo, el número de épocas de entrenamiento y aumentar la precisión. Para lograrlo, presentaron un enfoque de entrenamiento basado en *Histogram of Oriented*

Gradients y Convolutional Kernel Extreme Learning Machine. La técnica que utilizaron evaluó el conjunto de datos de AT&T, Yale y JAFFE. Al comparar los resultados con técnicas tradicionales como HOG-CNN, la evaluación indicó que la técnica propuesta fue capaz de lograr un rendimiento superior en términos de precisión y tiempo. También se propuso una hipótesis específica, la cual fue: “La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión redujo el tiempo de reconocimiento de imágenes faciales con respecto al algoritmo de pérdida del triplete”.

La hipótesis específica planteada tiene como base el estudio realizado por Adithama, Maslim y Gunawan (2023) quienes en su investigación propusieron una solución para resolver el problema de un hospital en garantizar la seguridad de sus pacientes, dado que la identificación precisa de los pacientes se presentaba como una dificultad en la seguridad. Aunque, como indican los autores, el hospital contaba con el nombre y número de paciente, se necesitaron métodos adicionales para reducir la posibilidad de errores en la identificación de pacientes. Para ello, la solución propuesta en la investigación se presentó en forma de autenticación facial, en la cual se empleó la técnica de transferencia de aprendizaje utilizando el modelo pre entrenado VGGFace2 y ResNet 10 para realizar el pre procesamiento de imágenes. De igual forma, se utilizó un algoritmo de aprendizaje de una sesión, que implicó tener una única muestra por cada persona. Como resultado, se logró un registro de pacientes con una precisión entre el 90% y el 100%, y en el proceso de verificación del paciente, la precisión alcanzó el 100% gracias a la implementación de aprendizaje de una sesión, que resultó clave en la recolección de datos, permitiendo un entrenamiento eficiente. Los autores concluyeron que el algoritmo RestNet 10 es superior a los demás algoritmos del estudio debido a su mayor precisión en comparación a Viola Jones y a un tiempo más rápido que el algoritmo MRCNN.

Además, la hipótesis también se fundamenta en el trabajo llevado a cabo por Kim (2022), investigación donde se tuvo como finalidad plantear una tecnología para solucionar el problema de control de acceso de personas, ya que, debido a problemas causados por la COVID-19, se perdió tiempo al medir la temperatura corporal y verificar a las personas que ingresaban a un edificio, teniendo que requerir personal que verifique si el usuario que ingreso tenía o no la mascarilla. Por lo tanto, surge el problema de pérdida de mano de obra y la probabilidad de infección del personal. En

el estudio, se recurrió a desarrollar un sistema sin contacto para el ingreso de personas y control. Para la tecnología que se desarrolló en la investigación, se utilizó el algoritmo Single Shot Multibox Detector (SSD) de aprendizaje en una sesión. Los resultados mostraron una precisión del 100% cuando se probaron con 20 fotografías aleatorias de rostros humanos y una alta precisión del 90% cuando se probó a través de una cámara web en tiempo real.

Además, se planteó una segunda hipótesis específica, la cual fue: “La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión incrementó la precisión en el reconocimiento facial con respecto al algoritmo de pérdida del triplete”.

La hipótesis específica planteada tiene como base el estudio realizado por Djamaluddin et al. (2023), quienes propusieron un estudio en el cual se estudiaron los problemas de verificación facial en un conjunto de datos que solo tenía una imagen frontal y una de perfil por individuo. Se usó una red siamesa y como resultado se obtuvo una mejora de la precisión del 17.0%; de igual manera, el aprendizaje de una sesión fue fundamental para lograr el alto rendimiento.

Por otro lado, la hipótesis planteada también incluye como base el estudio de Sun et al. (2022), quienes propusieron una ResNet con pérdida del triplete modificada, que obtuvo el 99.20% de precisión usando el conjunto de datos LFW y con la ayuda del aprendizaje de una sesión. La función de pérdida triplete modificada mantuvo las distancias euclidianas entre diferentes categorías distinguibles y pudo así reconocer rostros con mascarillas faciales. De igual forma, en el estudio de Saxena y de Varshney (2021), se propuso un sistema para capturar la imagen de una persona y determinar si está usando mascarilla o no, para la seguridad de una casa inteligente. Se utilizó una red siamesa y el aprendizaje de una sesión mediante el modelo FaceNet para verificar si una persona lleva mascarilla o no. Según los autores, el aprendizaje de una sesión se destacó en situaciones donde el conjunto de datos de entrenamiento puede ser limitado, posiblemente con una o unas pocas imágenes por persona. Como resultado, el modelo propuesto reportó una precisión final del 82.71% para un sistema de casa inteligente.

Además, la hipótesis también se fundamenta en el trabajo llevado a cabo por Hindy, Tachtatzis, Atkinson, Bayne y Bellekens (2021), donde se desarrolló una investigación

con el objetivo de resolver el problema de rendimiento y robustez para un sistema de detección de intrusos informáticos. Usualmente, para un sistema tradicional, se requiere una gran cantidad de datos para entrenar, y el rendimiento depende en gran medida del tamaño del conjunto de datos. Por otro lado, los ataques de día cero exigen que los modelos de aprendizaje vuelvan a entrenar para identificar futuros ataques informáticos. Sin embargo, la sofisticación creciente de los ataques informáticos ha llevado a los autores a plantear nuevas soluciones y, basándose en la problemática, propusieron un nuevo modelo de sistema de detección de intrusos que puede aprender similitudes con pocos datos de entrenamiento mediante aprendizaje de una sesión y redes siamesas. El modelo se evaluó utilizando tres conjuntos de datos como IDS: CICIDS2017, KDD Cup y NSL-KDD. Como resultado, se obtuvo un modelo con una precisión que alcanza el 84% para la CICIDS2017, el 88% para KDD Cup, mientras que para la NSL-KDD la precisión alcanzó más del 91%, clasificando los ataques en base a la similitud.

Por otro lado, también se presentaron investigaciones previas internacionales y nacionales, y a continuación, se detallaron de manera exhaustiva las teorías relacionadas que guardaban relación con la presente investigación.

En el estudio realizado por Cadena (2021), se abordó el problema del reconocimiento facial en imágenes de rostros relacionados a la posición, expresión, edad, iluminación, ruido y la presencia de objetos como, por ejemplo, lentes, sombreros o barba. La solución que se ideó en la investigación fue utilizar imágenes en tres dimensiones, para lo cual se seleccionaron algoritmos eficientes para la extracción de características, específicamente aquellos que pudieran realizar la tarea, como los filtros de Gabor. El proceso involucró etapas de entrenamiento y prueba. La técnica utilizada en el método de validación fue la validación cruzada. Para el entrenamiento 2D y 3D se utilizó el 100% de la colección de datos FERET y MUCT, así como TEXAS3DFRD. Como resultado, se obtuvo que en imágenes de 2 dimensiones la eficiencia fue del 96% y del 98.4% en el caso de imágenes 3D logrando resultados prometedores.

Por otro lado, en el estudio realizado por Tsai, Tsai y Chi (2023), se abordó el problema de la necesidad de desarrollar un sistema de detección y reconocimiento facial que pueda ser usado como un sistema de control de acceso en tiempo real,

utilizando un enfoque de reconocimiento facial de una sesión. En base al problema el objetivo de la investigación fue superar las limitaciones de los métodos tradicionales de reconocimiento facial el cual se basa en el entrenamiento con una gran cantidad de datos para los modelos de inteligencia artificial. La solución propuesta al problema consistió en implementar un sistema de detección y reconocimiento facial mediante el uso de aprendizaje profundo. Para ello, se utilizaron dos modelos en el sistema: SSD (*Single Shot Multibox Detector*) para situaciones que requieran detección facial o reconocimiento de características faciales, y por otro lado, se empleó VGG-Face para el reconocimiento facial. En el entrenamiento, se utilizaron técnicas como el aumento de datos, así como el pre procesamiento y postprocesamiento de imágenes. Para evaluar los resultados, se utilizaron videos con una resolución de 1280 x 960 píxeles en pruebas experimentales. Como resultado del estudio, se logró obtener un sistema de reconocimiento facial que se ejecuta a una velocidad aproximada de 30 fps para cada frame donde se reconoce el rostro.

Por otro lado, en la investigación de Lin y Yen (2023) se abordó el problema existente en la detección de imágenes falsas. Aunque las técnicas de reconocimiento de imágenes falsas son eficientes para detectar imágenes que el sistema ya conoce, según los autores no muestran resultados positivos cuando se trata de reconocer imágenes falsas que aún no se han identificado o no se han entrenado. Para resolver este problema, se planteó la generación manual de imágenes falsificadas al sistema, los autores de la investigación indicaron que esta técnica muestra limitaciones al no poder reconocer nuevas imágenes falsificadas. Para superar estas limitaciones, se propuso un enfoque de aprendizaje que implica el entrenamiento de un detector de imágenes falsificadas utilizando técnicas de aprendizaje meta. Esto permite al detector de imágenes falsas detectar con solo unas pocas muestras nuevas imágenes falsificadas, tomando un pequeño número de imágenes falsificadas como entrada y ajustando los pesos basándose en características estadísticas de las imágenes falsificadas de entrada. Como resultado, se observó que la solución propuesta mostró mejoras en la detección de nuevos métodos de falsificación, con mejoras en los índices de superposición (IoU) desde el 35.4% hasta el 127.2%, y mejoras en el área bajo la curva (AUC) desde el 2.0% hasta el 48.9%, dependiendo del método de falsificación.

Asimismo, en el estudio de Deng, Chiang, Kang y Li (2023), se propuso una investigación en la cual se buscó plantear una tecnología que resuelva el problema de recursos limitados en dispositivos móviles que usan modelos de inteligencia artificial para el reconocimiento facial, con el objetivo de optimizar el rendimiento sin comprometer la eficiencia computacional y el tamaño de memoria, problemas a los que usualmente se enfrentan las tecnologías en tiempo real, como controles de seguridad, sistemas de asistencia, etc. Frente a ello, se desarrolló un modelo al que denominaron FN8, un modelo diseñado para el reconocimiento facial que utiliza aprendizaje de una sesión o pocos ejemplos para obtener características efectivas. La tecnología desarrollada se inspiró en FaceNet, con la diferencia de que incorporó cambios significativos para lograr un tamaño de memoria de solo 3.5 MB, aproximadamente 30 veces más pequeño que FaceNet. Como resultado de la tecnología desarrollada, se obtuvo que el modelo propuesto superó a modelos ligeros en el consumo de recursos computacionales (FLOPs) para el uso en dispositivos ligeros.

Por otro lado, en la investigación liderada por Zhang, Yi, Guo, Tang, Feng y Liu (2023), se presentó una solución al problema de estudios ecológicos en los cuales se utilizan cámaras trampa para obtener imágenes detalladas de la vida silvestre. El problema se encontró en que dichas cámaras capturaban solo una pequeña fracción de las imágenes de animales que aún no se conocen y que viven en hábitats naturales. Según los investigadores a menudo, solo se lograba capturar un destello de ellos, resultando en fotografías de animales desconocidos incompletas o con fondos complejos, y a veces no se registraba ninguna imagen. Para abordar el problema, se propuso un método de aumento de datos basado en una red adversaria cíclica que cuenta con dos discriminadores y dos generadores. La solución propuesta implicó alimentar a los generadores con imágenes generadas manualmente. La propuesta combinó datos manuales y estilización de imágenes a través de una red adversaria cíclica para superar las limitaciones de muestras pequeñas o imágenes de animales incompletas. Como resultado, se obtuvo una precisión de clasificación del 92.2% y un puntaje del 93.3% al clasificar seis especies de vida silvestre previamente no encontradas. Esto se logró gracias al aprendizaje del modelo utilizando aprendizaje de una sesión en el entrenamiento.

Asimismo, en el trabajo realizado por Chatterjee, Chatterjee, Islam y Khan(2023), se llevó a cabo una investigación con el objetivo de crear un filtro para el contenido multimedia que se sube en plataformas, ya que como se explicó en la investigación, la generación masiva de datos multimedia plantea un desafío en la evaluación eficiente de la calidad del contenido visual, con el riesgo de desencadenar disturbios públicos y amenazar el orden social. Según los investigadores a pesar de los filtros y la moderación implementados por las plataformas de redes sociales, resulta difícil verificar la gran cantidad de contenido multimedia que se carga en estas plataformas. La investigación propuso una solución basada en RCNN, EfficientDet y YOLOv5 para diseñar una arquitectura que permita identificar de manera eficiente rostros humanos y diferentes tipos de armas de fuego en imágenes y videos en las plataformas. Para lograr el objetivo, se empleó un enfoque de aprendizaje de una sesión, optimizando así el rendimiento del modelo en situaciones donde no se cuenta con grandes cantidades de datos multimedia.

Como resultado, se obtuvo que el modelo YOLOv5 se destacó como el más efectivo en la investigación realizada, logrando una precisión promedio del 80.39% y un rendimiento del 35.22% para el reconocimiento de objetos en el contenido multimedia.

Por otro lado, en el estudio de AUFAR y SITANGGANG (2022) sobre el reconocimiento facial, se llevó a cabo una investigación en la cual se abordó el problema de reconocimiento de nuevos rostros, que un modelo de reconocimiento facial tradicional no podría identificar al no estar entrenado para reconocer dicho rostro. Para resolver el problema, se desarrolló una aplicación móvil que utilizó un modelo con arquitectura de redes convolucionales, incluyendo una red siamesa la cual se empleó para que el modelo aprendiera la similitud entre dos representaciones de objetos, especialmente en tareas de aprendizaje de una sesión.

Se utilizó el método de aumento de datos para mejorar el reconocimiento al generar, a partir de una sola imagen, más imágenes con las mismas características. Como resultado de la investigación, se obtuvo que el uso del aprendizaje de una sesión con una red siamesa y el método de aumento de datos produjo resultados prometedores al emplear un conjunto de datos de 9000 imágenes faciales y un entrenamiento de 5 épocas o ciclos, logrando una precisión del 98% al validar con datos de prueba.

Asimismo, en la investigación realizada por Holkar, Walambe y Kotecha (2022) se exploró el aprendizaje de una sesión para el reconocimiento facial y presentaron una investigación en la cual se enfrentaron a las limitaciones principales del aprendizaje profundo, que fue la dependencia de grandes cantidades de datos para un entrenamiento efectivo de un modelo. Según los investigadores, al tener pocos datos para el entrenamiento, el resultado es un modelo que no puede generalizar bien. Para resolver el problema, se utilizaron el conjunto de imágenes SCAAI_FSL con redes siamesas, y como resultado se obtuvo un 72.72% de precisión en el reconocimiento facial de imágenes con pocos datos de entrada.

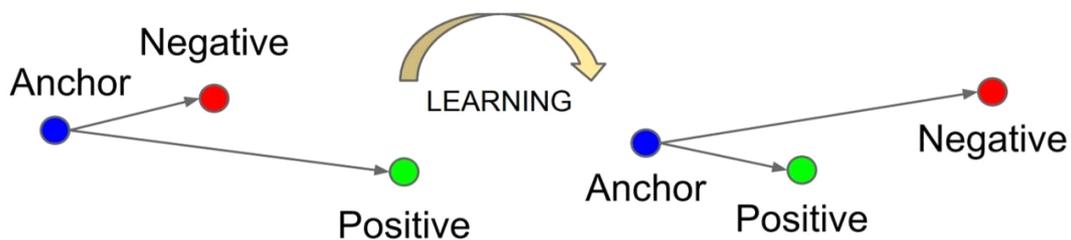
Por otro lado, en el desarrollo de esta investigación, se emplearon teorías relacionadas como la pérdida del triplete que, según Kumar, N, Saranya, Priyadharshini, E, Derrick, y M, Kaleel (2023), tenía como finalidad entrenar un modelo para que las características de imágenes similares se ordenaran y agruparan, mientras que las características de imágenes que no se relacionaban estuvieran separadas. Este enfoque se basaba en tener tres ejemplos simultáneamente, conocido como triplete, donde cada triplete constaba de un ancla que estaba aprendiendo la representación y que iba tomando las características positivas mediante el entrenamiento, con el objetivo de minimizar la distancia entre el ancla y positivo.

Según los investigadores la pérdida del triplete también constaba de un elemento negativo, que era diferente al ancla, y las características negativas no se acercaban al ancla durante el proceso de entrenamiento, sino que se iban distanciando. También indicaban que el proceso mejoraba la capacidad del modelo para reconocer patrones en imágenes y que la pérdida del triplete se calculaba utilizando la fórmula de distancia euclidiana.

Según Hernandez-Diaz et al. (2023), la pérdida del triplete también se utilizaba para extraer un vector de incrustación que maximizara la distancia entre clases y minimizara la distancia intraclasses hasta cierto margen. De igual forma, Wright y Stewart (2020) indicaban que la pérdida del triplete servía para minimizar la distancia entre el ancla y el positivo y maximizar las distancias entre el negativo y el ancla actualizando los pesos.

Asimismo, según O'Mahony, Campbell, Carvalho, Krpalkova, Hernandez, Harapanahalli, Riordan y Walsh (2019), la pérdida del triplete es una función más avanzada que la pérdida contrastiva que tiene como base tres tripletes de imágenes: por un lado, tenemos la imagen ancla, una positiva y una imagen negativa, y tiene como finalidad encontrar un espacio de incrustación donde las clases o diferencias entre imágenes sean mínimas al margen establecido. La pérdida del triplete se podría visualizar de la siguiente forma:

Figura 1 . Funcionamiento de algoritmo



Fuente: obtenido de “One-shot learning for custom identification tasks”, por O'Mahony et al., 2019

De igual manera, Sahito, Frank y Pfahringer (2020) indicaron que la pérdida del triplete se puede clasificar en tripletes fáciles: aquellos con pérdida de 0, tripletes fuertes: aquellos donde el positivo está más cerca al ancla, y los tripletes semifuertes: aquellos en los que los negativos no están cerca del ancla. También indicaron que la fórmula para el cálculo de distancia euclidiana que usa la pérdida del triplete sería la siguiente:

Figura 2 . Fórmula de pérdida del triplete

$$\mathcal{L} = \max(d(a, p) - d(a, n) + m, 0)$$

Fuente: Formula de pérdida del triplete, obtenido de “Transfer of pretrained model weights substantially improves semi-supervised image classification”, por(Sahito et al., 2020).

El desglose de componentes en la fórmula de la función de pérdida del triplete para el reconocimiento facial sería el siguiente:

l: Es la función de pérdida del triplete.

(a, p) : Representa la distancia entre el ancla y positivo.

(a, n) : Representa la distancia entre ancla y negativo.

m : Es un margen positivo, es el valor mínimo esperado para la diferencia entre las distancias de los negativos y positivos.

0: Asegura que la pérdida no sea negativa.

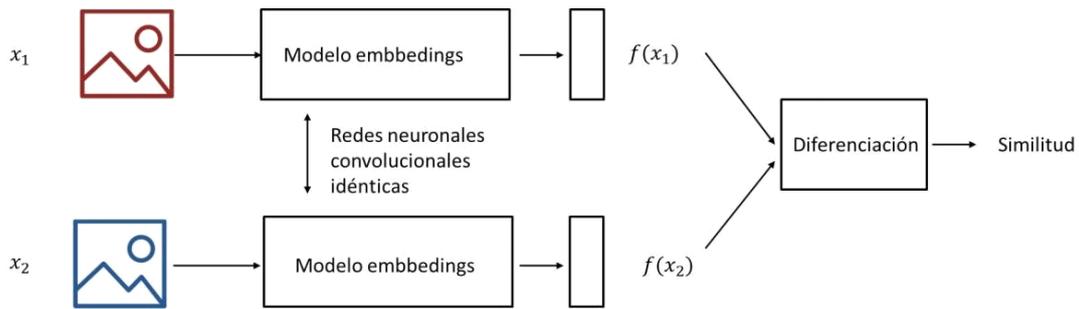
Por otro lado, en el desarrollo de esta investigación, también se han empleado teorías relacionadas, como la red siamesa que, según Kumar et al. (2023), es una arquitectura de red neuronal que utiliza dos ramas idénticas para procesar datos de entrada de manera simultánea. De igual manera, Sahito et al. (2020) indicaron que las redes siamesas son redes neuronales para entrenar utilizando una sola función de pérdida, aunque en algunos casos también pueden usar varias. Según los autores se podría tomar como referencia la red siamesa como dos copias idénticas de la misma red que comparten pesos.

Asimismo, Hayale, Negi y Mahoor (2023), sustentaron que las redes siamesas se componen de dos redes idénticas que comparten los pesos. También O'Mahony et al. (2019), indicaron que una red neuronal siamesa tiene como finalidad medir la similitud entre dos objetos y consiste en dos subredes idénticas, ambas con los mismos parámetros y pesos.

Por otro lado, según Wright y Stewart (2020), las redes siamesas se utilizan para conocer la similitud entre las imágenes y esta función se desarrolla entrenando una red en la que se usan 3 entradas a la vez, donde 2 de las entradas pertenecen a un tipo de imagen que tiene características similares y una tercera que pertenece a una imagen distinta a las dos. Asimismo, Saxena y Varshney (2021) indicaron que las redes neuronales siamesas pueden trabajar con un conjunto de datos limitado utilizando un enfoque de aprendizaje de una sesión.

De igual forma, Cardaioli, Conti, Orazi, Tricomi y Tsudik (2023) indicaron que las redes siamesas están ganando terreno en el área de reconocimiento facial debido a su capacidad para escalar, lo que equivaldría a reconocer a distintas personas con pocas muestras cada una.

Figura 3 . Arquitectura de redes siamesas



Fuente: Elaboración propia

En esta investigación, también se han empleado teorías relacionadas, como el aprendizaje de una sesión que, según Jiménez-Bravo et al. (2022), la técnica de aprendizaje en una sesión fue introducida por primera vez en el año 2000 y se caracteriza por ser un subtipo de aprendizaje supervisado que utiliza datos etiquetados o ejemplos para aprender y generar el modelo. Usualmente, los modelos requieren una gran cantidad de datos para funcionar correctamente, pero esta técnica permite hacer lo contrario, ya que con solo unas pocas muestras puede entrenar y generar un modelo. Según Hernandez- Diaz et al. (2023), el aprendizaje de una sesión se utiliza cuando no hay datos disponibles para entrenar.

Según O'Mahony et al. (2019), los sistemas de reconocimiento de objetos o rostros, en caso de no reconocer a una persona, significaría que la imagen de esa persona no se almacena en la base de datos del sistema. Para solucionar ese problema, no es conveniente volver a entrenar el modelo cada vez que se agrega la imagen de una nueva persona, y es en ese tipo de situaciones donde nace el aprendizaje de una sesión o a veces llamado one-shot.

Según Hindy et al. (2021), el aprendizaje de una sesión también puede usarse para diseñar y construir sistemas de detección de intrusos para seguridad informática, ya que resuelve el problema de ataques de día cero, permitiendo que los sistemas estén al día con las nuevas amenazas en tiempo real. Según los autores, también se puede usar el aprendizaje de una sesión en tareas muy diversas, como en el caso del aprendizaje de una sesión para detectar malware, basándose en la transformación del malware a una estructura similar a una imagen. Según las teorías explicadas sobre el aprendizaje de una sesión, se puede entender que se trata de una tecnología

que permite utilizar reconocimiento de objetos donde no se tienen grandes cantidades de datos para entrenar.

En este estudio, también se han utilizado teorías afines, como K-Vecinos más cercanos, que según Fathian, Phan, Ho y Rouhani (2023), el K-Vecinos más cercanos (KNN) es un clasificador que se utiliza para clasificar datos y detectar similitudes en base a la votación de "K" vecinos, siendo "K" una variable que indica la cantidad de muestras que se tomará para la votación entre vecinos.

Según Ahmed y Mahmood (2023), el algoritmo de K-Vecinos más cercanos es un método estadístico para la identificación de patrones y se usa como algoritmo de clasificación supervisado que tiene como objetivo capturar muestras de vecinos para poder reconocer patrones en base a decisiones entre grupos de vecinos establecidos por la variable "K", lo que permite la flexibilidad en capturar datos complejos. Según Yan, Hu y Liu (2023), el algoritmo de K- Vecinos en principio funciona con un conjunto de muestras etiquetadas y se utiliza para calcular las muestras más cercanas calculando la distancia de nuevos datos.

Según el estudio de Rajani, Patil, Marathe, Mhaisane y Tundalwar (2023), el algoritmo de K-Vecinos más cercanos en situaciones de clasificación implica varios pasos a tener en cuenta. Como primer paso, se necesita elegir un valor para K, que representa el número de vecinos más cercanos al realizar una predicción del nuevo punto de datos. A continuación, se hace un cálculo de distancia entre el nuevo punto y todos los puntos ya existentes. En algunas situaciones, se usa la distancia euclidiana para identificar los puntos más cercanos y para determinar una predicción para un nuevo punto de datos, se utiliza el voto mayoritario de las etiquetas de los K-Vecinos más cercanos. En caso de empate, se aplican otras reglas, como seleccionar la etiqueta del vecino más cercano.

En un estudio reciente, Rajani, Patil, Marathe, Mhaisane y Tundalwar (2023) mencionaron que el algoritmo para construir un modelo KNN para un problema de clasificación es el siguiente: Definición de parámetros iniciales: Seleccionar un valor para K, que indica la cantidad de vecinos cercanos a tomar en cuenta al realizar una predicción para un nuevo punto de datos. Cálculo de distancias: Determinar la distancia entre el nuevo punto de datos y todos los puntos en el conjunto de entrenamiento. La métrica de distancia puede ser euclidiana, manhattan u otras

medidas de distancia. Identificación de los K puntos más cercanos: Identificar los K puntos en el conjunto de entrenamiento que están más cercanos al nuevo punto de datos basándose en las distancias calculadas. Predicción basada en la mayoría de votos: Para hacer una predicción para el nuevo punto de datos, tomar el voto mayoritario de las etiquetas de clase de los K vecinos más cercanos. En caso de empate (especialmente cuando K es par), romper el empate utilizando reglas como seleccionar la etiqueta de clase del vecino más cercano. Repetición del proceso para todos los nuevos puntos de datos: Repetir los pasos 2 a 4 para todos los nuevos puntos de datos a clasificar.

En este estudio, también se han utilizado teorías afines, como el reconocimiento facial, que según Kumar et al. (2023), el software de reconocimiento facial básicamente funciona comparando las características faciales y geométricas de la persona.

Así también, Cardaioli et al. (2023) indicaron que el reconocimiento facial se basa en métodos discriminativos que pueden diferenciar rostros. Según Jiménez-Bravo et al. (2022), una de las áreas de investigación más activas de los últimos años es el reconocimiento de objetos y uno de los sub campos es el reconocimiento facial, el cual es un conjunto de técnicas que tienen como objetivo identificar rostros dentro de un conjunto de imágenes y, en base a los patrones mostrados, poder realizar el reconocimiento de rostro comparando con datos entrenados previamente.

Según O'Mahony et al. (2019), el reconocimiento facial es una de las aplicaciones de visión artificial de gran importancia para el reconocimiento de objetos, ya que no solo tiene un interés de valor comercial sino también en ámbitos de seguridad y re identificación de personas en aplicaciones.

II. METODOLOGÍA

En este capítulo, se abordaron elementos clave de la investigación, tales como la naturaleza y estructura del estudio, así como se describieron los criterios utilizados para la elección de la muestra, junto con consideraciones éticas específicas.

Para obtener una comprensión más profunda del tipo de investigación, resultó valioso explorar los conceptos fundamentales. En este sentido, Hernández et al. (2014) proporcionaron una detallada exploración sobre la metodología de la investigación en su obra "Metodología de la investigación". En la obra se señaló que toda indagación perseguía dos objetivos fundamentales: generar conocimiento y también teorías, y abordar problemas específicos, como sería el caso de la investigación aplicada.

De igual forma, según Nieto (2018) en su obra "Tipos de Investigación", Se señaló que el objetivo de la investigación aplicada era abordar los desafíos asociados a los procesos de producción, distribución, circulación y consumo de bienes y servicios en distintas actividades humanas. También indicó que la denominación de "aplicada" se basaba en que, a partir de investigaciones fundamentales en ciencias fácticas o formales, se planteaban problemas o hipótesis de trabajo con el fin de resolver las problemáticas vinculadas a la vida productiva de la sociedad. Se eligió el tipo de investigación aplicada debido a que, al desarrollar un algoritmo de reconocimiento facial, se pretendía resolver un problema existente en el reconocimiento de imágenes faciales.

Este estudio se ejecutó bajo un enfoque cuantitativo, ya que, según Hernández et al. (2014), este enfoque en la investigación seguía un proceso que buscaba medir y estimar las magnitudes de los fenómenos o problemas investigados.

En cuanto al diseño de la investigación, fue de tipo pre-experimental, ya que, como mencionaron Hernández et al. (2014), este enfoque proporcionaba un marco riguroso para examinar las relaciones de causa y efecto en el ámbito de la investigación científica.

En este proyecto de investigación, la variable analizada fue el Impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales.

A continuación se detalla información relevante para las variables y operacionalización:

El uso del reconocimiento facial se ha convertido en un gran problema, principalmente debido al aumento de situaciones en las que existen datos nuevos en tiempo real (Hernandez-Diaz et al., 2023; Xu et al., 2023).

Para las dimensiones, se tuvo en cuenta el reconocimiento de imágenes (Cadena, 2021; Tsai et al., 2023; Lin y Yen, 2023; Deng et al., 2023; AUFAR y Sitanggang, 2022; Holkar y Walambe 2022; Cardaioli et al., 2023), así como la precisión (Zhang et al., 2023; Chatterjee et al., 2023; AUFAR y Sitanggang, 2022; Holkar et al., 2022). En cuanto a los indicadores, se evaluó el tiempo de reconocimiento facial (Adithama et al., 2023; Kim, 2022) y la precisión de reconocimiento facial (Djamaluddin et al., 2023; Sun et al., 2022; Saxena y de Varshney., 2021; Hindy et al., 2021).

La escala utilizada para esta investigación es la razón, ya que según Hernández et al. (2014), en esta escala se reúnen todas las características inherentes al nivel de intervalos, como la presencia de una unidad de medida común, intervalos uniformes entre las categorías y la capacidad para aplicar operaciones aritméticas básicas y sus derivaciones. A diferencia del nivel de intervalos, en este caso, el cero es real y absoluto, lo que significa que existe un punto en la escala donde la propiedad medida está completamente ausente o inexistente.

A continuación, se describen los términos relacionados con población, muestra, proceso de muestreo y unidad de análisis.

Según Hernández et al. (2014), la población se define como una serie de casos que cumplen con determinadas especificaciones. En consecuencia, para el presente estudio se tomó como población 400 imágenes de representaciones faciales del dataset ORL.

En un estudio reciente de Madmoune, El Ouariachi, Zenkouar y Zahi (2023) sobre reconocimiento facial, indicaron que ORL es un dataset de rostros que contiene un total de 400 imágenes distribuidas en 40 individuos, es decir, 10 imágenes por individuo, y es ampliamente utilizado en estudios de reconocimiento facial.

Las condiciones bajo las cuales se tomaron estas imágenes difieren ya sea por los detalles del rostro (con o sin gafas) o por la variación en la iluminación o la expresión

facial (sonriendo o no, ojos abiertos o cerrados). Según Benradi, Chater y Lasfar (2023), las imágenes están en escala de grises con un tamaño de 92x112 píxeles. La mayoría de las imágenes son fotos tomadas por fotógrafos profesionales y, por lo tanto, se consideran una buena opción para obtener buenos resultados.

A continuación, se detallan los criterios de inclusión y exclusión: los criterios de inclusión se refieren a archivos de tipo imagen de rostros humanos, mientras que los criterios de exclusión abarcan archivos de formatos diferentes a imágenes o archivos que no sean rostros humanos.

En cuanto a la muestra, para esta investigación se realizó el estudio sobre 40 imágenes de representaciones faciales existentes en el dataset ORL. López (2004) indicó que para tomar una muestra, lo recomendable es seleccionar el mayor tamaño posible, ya que cuanto más grande y representativa sea la muestra, menor será el error de la misma.

Una vez determinada la cantidad de la muestra, se procede a seleccionar el tipo de muestreo a utilizar. Se aplicará un muestreo estratificado con el objetivo de asegurar que cada estrato tenga la misma probabilidad de ser representado en la muestra. En este caso, se seleccionarán 10 imágenes de representaciones faciales de cada estrato para ser incluidas en la muestra. Según Hernández et al. (2014), el muestreo implica seleccionar una parte representativa de la población o universo con la finalidad de responder a un planteamiento sobre un problema de investigación.

Para el proyecto de investigación, la técnica utilizada fue la observación. En relación con esto, Bunge (2007) señaló que la observación constituye el procedimiento empírico fundamental en la ciencia, centrado en el estudio de uno o varios hechos, objetos o fenómenos de la realidad actual. En el estudio, se ha empleado la observación para analizar como la aplicación del algoritmo impactó en la reducción del tiempo de reconocimiento facial y en el aumento de la precisión en dicho reconocimiento. Para llevar a cabo esta investigación, utilizamos un instrumento diseñado para recopilar los resultados generados por el algoritmo propuesto.

Por otro lado, Moguel (2005) destacó que la observación no solo sugiere y motiva la sistematización de los datos, sino que también subrayó la necesidad de superar diversas limitaciones y obstáculos en la observación científica. Además, señaló que el término "observación" se refiere a la percepción visual y se utiliza para describir

todas las formas de percepción empleadas en el registro de respuestas. Enfatizó la importancia de diferenciar entre una respuesta y un dato, argumentando que una respuesta es, en cierto sentido, un manifiesto de acción, mientras que un dato es el resultado del registro de esa respuesta. También resaltó que una respuesta es observable, pero un dato es lo que se observa.

Durante la realización de este estudio, se adhirieron a principios éticos fundamentales, destacando especialmente el respeto a la autoría de las fuentes de información. Para asegurar una adecuada citación según el estilo ISO-690, se realizó una revisión exhaustiva de todas las fuentes utilizadas en el trabajo, garantizando que cada referencia estuviera correctamente atribuida y que se respetara la propiedad intelectual de los autores.

Se utilizó el dataset ORL, una base de datos pública de AT&T Laboratories Cambridge. Se reconoce y atribuye el crédito correspondiente a AT&T Laboratories Cambridge por su labor en el desarrollo y la recopilación de este dataset (AT&T Laboratories Cambridge, 2001). El dataset ORL se ha puesto a disposición del público para fines educativos y ha sido utilizado en numerosas investigaciones relacionadas con la inteligencia artificial.

Durante el proceso de investigación, se tomó especial cuidado para garantizar el cumplimiento de los términos de uso del dataset. Se aseguró que todas las imágenes y datos utilizados en la investigación fueran tratados con el debido cuidado y consideración.

El desarrollo de la investigación también se ajustó rigurosamente al código ético establecido por la Universidad César Vallejo. Se revisaron y siguieron todas las directrices y políticas éticas proporcionadas por la institución, asegurando que el estudio se llevara a cabo de manera ética y responsable en todos los aspectos. Se mantuvo una comunicación abierta y transparente con los supervisores pertinentes para garantizar el cumplimiento de las normativas y estándares éticos durante el proceso de investigación.

En resumen, se tomaron todas las precauciones necesarias para garantizar que la investigación se llevara a cabo de manera ética y responsable.

III. RESULTADOS

En este capítulo se detallaron los resultados en base a los indicadores "Tiempo de reconocimiento facial" y "Precisión de reconocimiento facial", mostrando el impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y en el aprendizaje de una sesión para el reconocimiento facial. Esta tesis se desarrolló en dos fases debido a la naturaleza pre-experimental de la investigación, la cual comprendía la aplicación de un pre-test y un pos-test. El pre-test se llevó a cabo usando el algoritmo de pérdida del triplete (**ver anexo 23**) antes de la implementación del algoritmo de pérdida propuesto (**ver anexo 24**), obteniendo resultados para el algoritmo del triplete (**ver anexo 8**). De igual manera, para el algoritmo propuesto se obtuvieron los resultados (**ver anexo 9**). Estos resultados se utilizaron para registrar la ficha en la que se contrastaba la precisión del algoritmo del triplete y el algoritmo propuesto (**ver anexo 2**). De igual forma, se registró una ficha para contrastar el algoritmo de pérdida del triplete y el algoritmo propuesto en base al tiempo (**ver anexo 3**).

En consecuencia, estos datos fueron analizados mediante una serie de procedimientos estadísticos, utilizando para ello el software SPSS de IBM.

3.1 Análisis descriptivo

Con la finalidad de evaluar los resultados en base al algoritmo de pérdida del triplete y el algoritmo propuesto, se utilizaron los siguientes análisis descriptivos para los indicadores.

Indicador: Tiempo de reconocimiento facial

Tabla 1 . Estadística descriptiva para el tiempo de reconocimiento facial

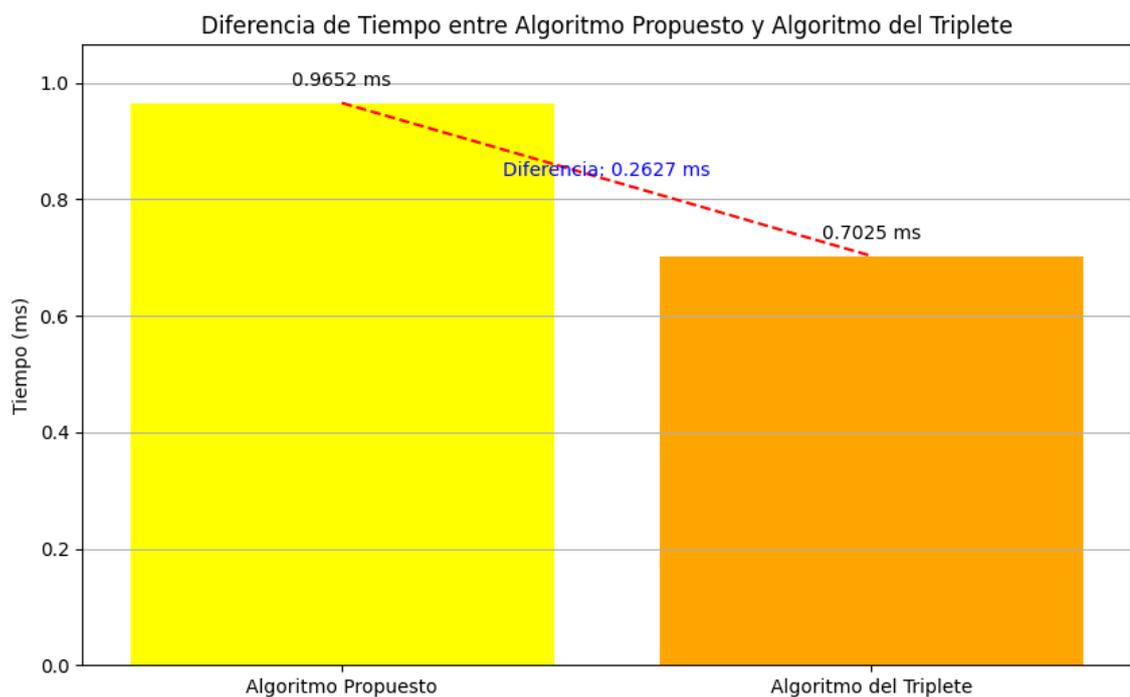
	Estadísticos descriptivos				
	N	Mínimo	Máximo	Media	Desv. Estándar
Algoritmo propuesto	40	.81	1.97	.9652	.23216
Algoritmo triplete	40	.64	.83	.7025	.06291

N valido 40
(por lista)

Fuente: Elaboración propia

Para el promedio del tiempo del algoritmo propuesto, se visualizó un valor de 0.9652, indicando los milisegundos. En cuanto al tiempo de reconocimiento facial del algoritmo del triplete, se observó que fue de 0.7025, cifra que represento de igual forma los milisegundos. Según los resultados, se evidenció que existió una diferencia entre el tiempo del algoritmo del triplete y el algoritmo propuesto, con un valor de 0.2627. De esta manera, se concluyó que la implementación del algoritmo propuesto no tuvo un mejor tiempo de reconocimiento facial frente al algoritmo de pérdida del triplete.

Figura 4 . Diferencia de tiempo entre algoritmo propuesto y algoritmo de pérdida del triplete



Fuente: Elaboración propia

Indicador: Precisión de reconocimiento facial.

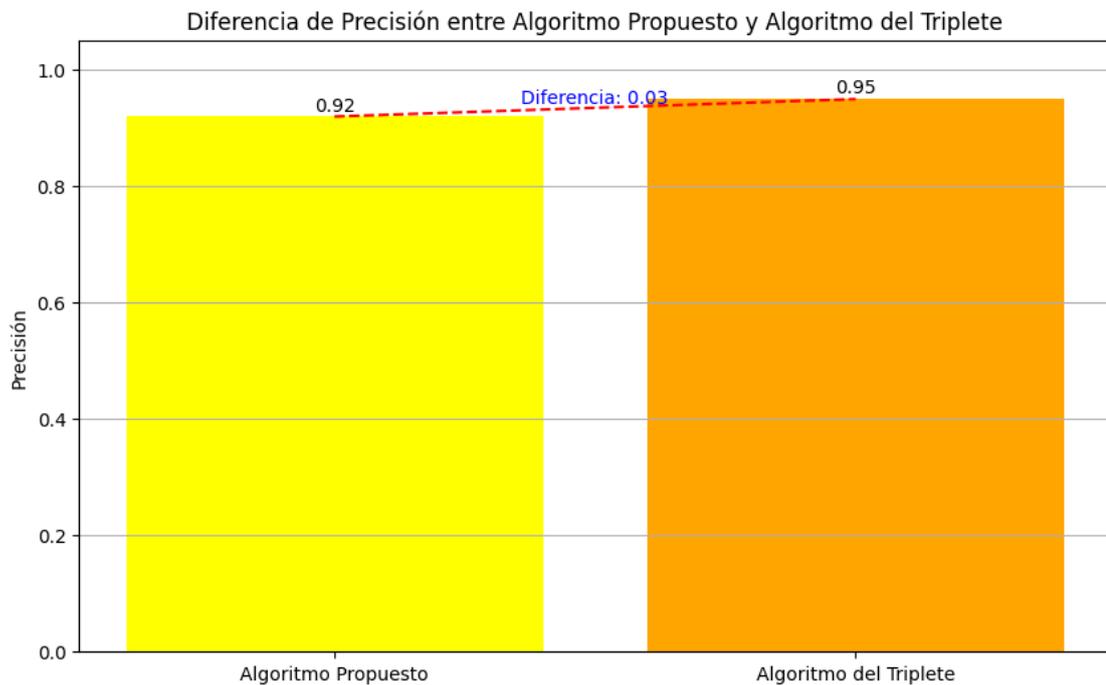
Tabla 2 . Estadística descriptiva para la precisión de reconocimiento facial

Estadísticos descriptivos					
	N	Mínimo	Máximo	Media	Desv. Estándar
Algoritmo propuesto	40	0	1	.92	.267
Algoritmo triplete	40	0	1	.95	.221
N valido (por lista)	40				

Fuente: Elaboración propia

Para el promedio de la precisión del algoritmo propuesto, se visualizó un valor de 0.92, indicando el porcentaje de precisión. En cuanto a la precisión de reconocimiento facial del algoritmo del triplete, se observó que fue de 0.95, cifra que representó de igual forma el porcentaje. Según los resultados, se evidenció que existió una diferencia entre la precisión del algoritmo del triplete y el algoritmo propuesto, con un valor de 0.03. De esta manera, se concluyó que la implementación del algoritmo propuesto no tuvo una mejor precisión de reconocimiento facial frente al algoritmo de pérdida del triplete.

Figura 5 . Diferencia de precisión entre algoritmo propuesto y algoritmo de pérdida del triplete



Fuente: Elaboración propia

3.2 Análisis inferencial

Prueba de normalidad

A continuación, se realizó la prueba inferencial para determinar la normalidad de los datos de cada indicador. Esta prueba se llevó a cabo utilizando Shapiro-Wilk debido a que la cantidad de muestras fue inferior a 50.

Las conclusiones de tomaran de acuerdo a lo siguiente: 'Sig. < 0.05 (no normal)' y 'Sig. >= 0.05 (normal)

Indicador: Tiempo de reconocimiento facial

Tabla 3 . Prueba de normalidad para el tiempo de reconocimiento facial

Pruebas de normalidad			
Shapiro-Wilk			
	Estadístico	Gl	Sig.
Algoritmo triplete	.702	40	.001

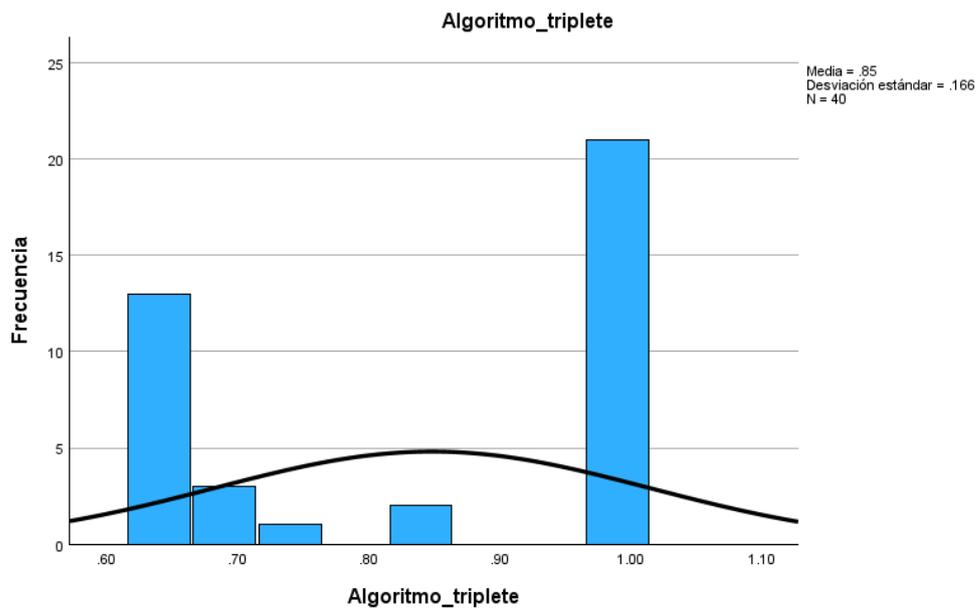
Algoritmo propuesto	.682	40	.001
--------------------------------	------	----	------

Fuente: Elaboración propia

Se observa que el valor de la significancia (sig.) tanto para el algoritmo del triplete como para el algoritmo propuesto es menor a 0.05. Por lo tanto, los datos del análisis se consideran no normales y se procederá a utilizar la prueba de hipótesis de Wilcoxon.

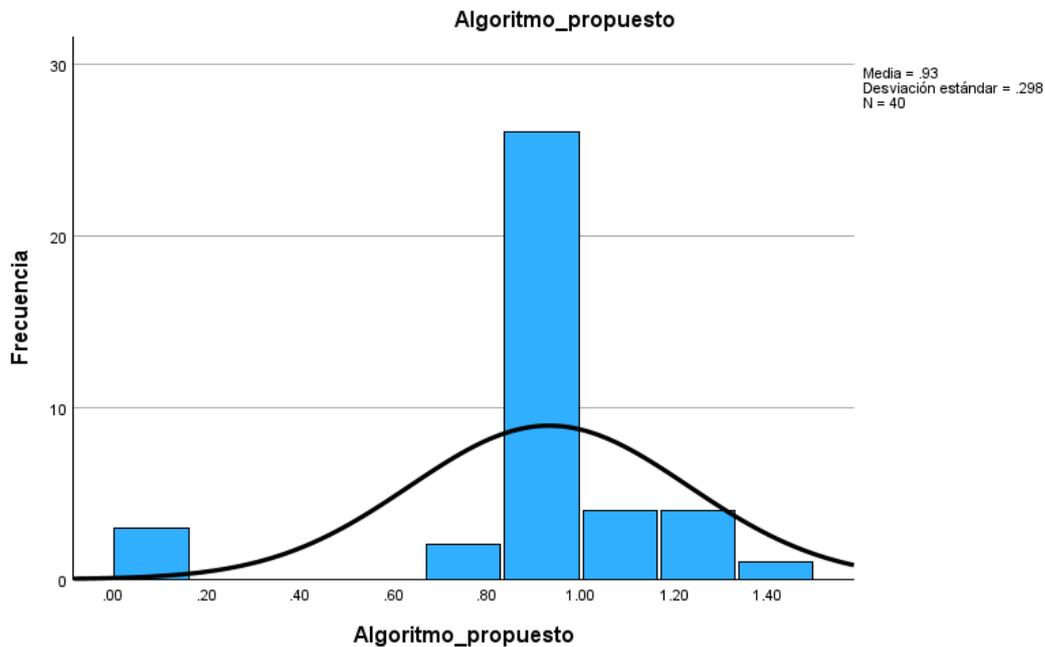
Esto se puede apreciar en los gráficos de campana de gauss que indican que la distribución de datos no es normal.

Figura 6 . Gráfico de normalidad de tiempo de reconocimiento facial con algoritmo del pérdida del triplete



Fuente: Elaboración propia

Figura 7 . Gráfico de normalidad de tiempo de reconocimiento facial con algoritmo propuesto



Fuente: Elaboración propia

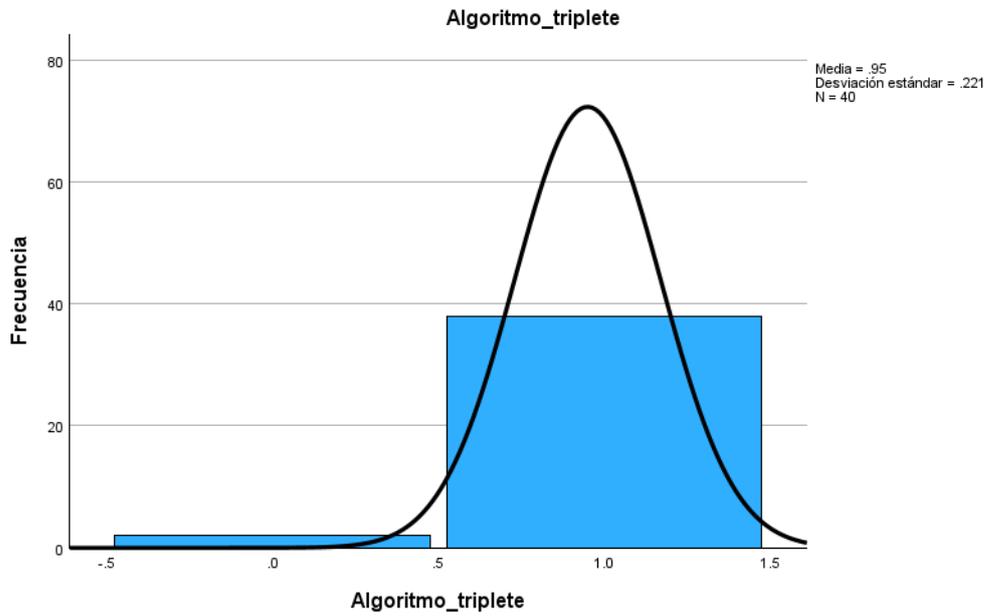
Indicador: Precisión de reconocimiento facial.

Pruebas de normalidad			
Shapiro-Wilk			
	Estadístico	Gl	Sig.
Algoritmo triplete	.229	40	.001
Algoritmo propuesto	.292	40	.001

Se observa que el valor de la significancia (sig.) tanto para el algoritmo del triplete como para el algoritmo propuesto es menor a 0.05. Por lo tanto, los datos del análisis se consideran no normales y se procederá a utilizar la prueba de hipótesis de Wilcoxon.

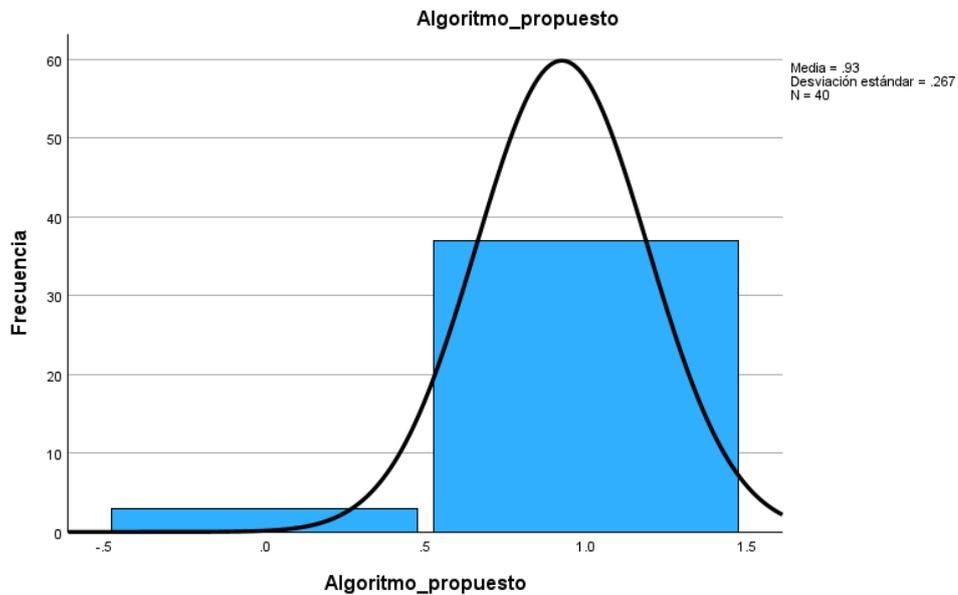
Esto se evidencia en los gráficos de campana de Gauss, los cuales muestran que la distribución de los datos no es normal.

Figura 8 . Gráfico de normalidad de precisión de reconocimiento facial con algoritmo del pérdida del triplete



Fuente: Elaboración propia

Figura 9 . Gráfico de normalidad de precisión de reconocimiento facial con algoritmo propuesto



Fuente: Elaboración propia

3.3 Prueba de Hipótesis

Para proceder a probar las hipótesis, previamente se analizó si la distribución de los datos fue normal o no. En este caso, se utilizaron pruebas de Wilcoxon para confirmar la hipótesis nula o rechazarla y aceptar la hipótesis alternativa, según sea el caso.

Hipótesis general:

H₀: La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión no incrementa la precisión y no reduce el tiempo de reconocimiento de imágenes faciales con respecto al algoritmo de pérdida del triplete.

H_a: La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión incrementa la precisión y reduce el tiempo de reconocimiento de imágenes faciales con respecto al algoritmo de pérdida del triplete.

Hipótesis específicas:

Indicador: Tiempo de reconocimiento facial

Hipótesis HE1:

H₀: La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión no redujo el tiempo de reconocimiento de imágenes faciales con respecto al algoritmo de pérdida del triplete.

H_a: La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión redujo el tiempo de reconocimiento de imágenes faciales con respecto al algoritmo de pérdida del triplete.

Para la evaluación de la hipótesis, se utilizó la prueba de Wilcoxon para muestras emparejadas con el propósito de analizar la hipótesis estadística.

Tabla 4 . Prueba de Wilcoxon de muestras emparejadas para el tiempo de reconocimiento facial

Prueba de Wilcoxon	
	Algoritmo propuesto
	Algoritmo del triplete
Z	-5.389
Sig. Asintótica (bilateral)	.001

Fuente: Elaboración propia

Después de realizar la prueba de Wilcoxon, se obtuvo un valor p (sig) de 0.001. Este valor p es menor que el nivel de significancia comúnmente utilizado de 0.05, lo que significa que hay suficiente evidencia para rechazar la hipótesis nula (H0). En otras palabras, la prueba de Wilcoxon indica que existe una diferencia estadísticamente significativa en los tiempos de reconocimiento de imágenes faciales entre los dos algoritmos.

Aunque la prueba de Wilcoxon muestra que hay una diferencia significativa en los tiempos de reconocimiento, es crucial considerar la dirección de esta diferencia. Observando los tiempos de latencia (**ver figura 4**), podemos ver que el algoritmo propuesto tiene un tiempo de 0.9652 milisegundos, mientras que el algoritmo del triplete tiene un tiempo de 0.7025 milisegundos. Esto significa que el algoritmo propuesto toma más tiempo para reconocer imágenes faciales en comparación con el algoritmo del triplete.

A pesar de que la prueba de Wilcoxon sugiere una diferencia significativa entre los algoritmos, la dirección de esta diferencia indica que el algoritmo propuesto no ha mejorado el tiempo de reconocimiento. De hecho, el tiempo de reconocimiento del algoritmo propuesto es mayor que el del algoritmo del triplete, lo que implica un peor desempeño en términos de latencia.

Dado el valor p de 0.001, se rechaza la hipótesis nula (H0). Sin embargo, la dirección de la diferencia indica que el algoritmo propuesto es menos eficiente en términos de tiempo de reconocimiento facial comparado con el algoritmo del triplete. Por lo tanto, la hipótesis alternativa (Ha) no se sostiene en el sentido esperado de reducción del tiempo de reconocimiento y se acepta la hipótesis nula, concluyendo que: la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión no redujo el tiempo de reconocimiento de imágenes faciales con respecto al algoritmo de pérdida del triplete.

Hipótesis HE2:

H0: La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión no incrementó la precisión en el reconocimiento facial con respecto al algoritmo de pérdida del triplete.

Ha: La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión incrementó la precisión en el reconocimiento facial con respecto al algoritmo de pérdida del triplete.

Tabla 5 . Prueba de Wilcoxon de muestras emparejadas para precisión de reconocimiento facial

Prueba de Wilcoxon	
	Algoritmo propuesto
	Algoritmo del triplete
Z	-.447
Sig. Asintótica (bilateral)	.655

Fuente: Elaboración propia

Dado que el valor de p obtenido es 0.655, el cual es mayor que el nivel de significancia comúnmente utilizado de 0.05, no hay suficiente evidencia para rechazar la hipótesis nula (H_0). Esto sugiere que la implementación del algoritmo basado en una red siamesa, pérdida del triplete, k -vecinos más cercanos y el aprendizaje de una sesión no incrementó la precisión en el reconocimiento facial con respecto al algoritmo de pérdida del triplete.

IV. DISCUSIÓN

En el ámbito de la investigación dedicada al reconocimiento de imágenes, se ha producido una tendencia de estudios que se centran en evaluar la precisión y el tiempo de una variedad de algoritmos y arquitecturas. Este campo, en constante evolución, se caracteriza por su dedicación a perfeccionar la precisión y la eficiencia de tiempo de los procesos vinculados a la identificación y clasificación de imágenes.

En esta sección se exploran los resultados obtenidos en la investigación, contrastándolos con los trabajos previos en cuanto al tiempo promedio de reconocimiento, la precisión del algoritmo. Asimismo, se realiza un análisis más detallado de los resultados presentados en el capítulo anterior.

La implementación del algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión no tuvo un impacto positivo en el reconocimiento facial, ya que no se logró el objetivo de reducir el tiempo en el reconocimiento facial, obteniendo un resultado de 0.96525 milisegundos en comparación con el tiempo de 0.7025 milisegundos obtenido por el algoritmo de pérdida del triplete.

El algoritmo propuesto buscó capturar los embeddings (vectores de características) de agrupaciones de tensores y obtener el valor más cercano, con el propósito de generar un modelo ligero, para que cuando se realice el post procesamiento se pueda encontrar los pesos más rápido, tanto por el tamaño del archivo como por el modelo optimizado que se genere, reduciendo así el tiempo de reconocimiento.

Por otro lado, la implementación del algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión no tuvo un impacto positivo en el reconocimiento facial, ya que no se logró el objetivo de aumentar la precisión en el reconocimiento facial, obteniendo un resultado del 92.5% de precisión en comparación con la precisión del 95% obtenido por el algoritmo de pérdida del triplete.

El algoritmo propuesto buscó capturar los embeddings (vectores de características) de agrupaciones de tensores y obtener el valor más cercano, con el propósito de aprender los patrones de las imágenes realizando cálculos de distancia de vecinos cercanos sobre los embeddings. Para un post entrenamiento de un modelo de representación de características, el algoritmo propuesto tuvo como objetivo ordenar

instancias similares cerca y no similares lejos del espacio de características, para que las representaciones de características aprendidas puedan clasificar nuevos ejemplos o imágenes en el espacio de características.

En este capítulo también se analizaron los resultados derivados de nuestra investigación. Este análisis detallado permitió una comprensión más profunda de la aplicación de nuestro algoritmo de reconocimiento facial. Además, se exploró cómo estos resultados se comparan y contrastaron con los hallazgos de otros investigadores referenciados en esta investigación.

En particular, resulta interesante analizar los hallazgos de Aufar y Sitanggang (2022), quienes lograron una precisión del 98% en 5 épocas de entrenamiento. Este rendimiento supera la precisión obtenida por nuestro algoritmo, que alcanzó el 92.5% de precisión. En la investigación desarrollada por los autores se llevó a cabo la implementación de una red siamesa en su arquitectura, al igual que en el caso de nuestro algoritmo que propusimos, ya que la inclusión de esta arquitectura pudo permitir que el modelo aprendiera la similitud entre dos representaciones de imágenes. Una de las razones detrás de esta diferencia porcentual de precisión pudo deberse al uso de un método de aumento de datos, estrategia que podría haber contribuido significativamente al mejoramiento del reconocimiento.

Por otro lado, Chatterjee et al. (2023) exploraron soluciones basadas en Redes Neuronales Convolucionales Basadas en Regiones, EfficientDet y YOLOv5, alcanzando una precisión promedio del 80.39%. Aunque el resultado es inferior al resultado obtenido de precisión, es crucial notar que el enfoque de nuestro algoritmo se centró en el análisis de similitud fina entre imágenes. En contraste, el uso de YOLOv5 se centra en escenarios donde se necesita un procesamiento con respuesta rápida, ya que el modelo está optimizado para los casos donde se tenga que realizar estudios o investigación que requieran tareas con baja latencia.

Asimismo, el estudio realizado por Holkar et al. (2022) también contribuye al panorama investigativo utilizando el conjunto de datos SCAAI_FSL con redes siamesas. Para la investigación se obtuvo una precisión del 72.72% en el reconocimiento facial. Este resultado resultó inferior a la precisión obtenida por el algoritmo que propusimos, incluso empleando una arquitectura similar con redes siamesas, lo que nos da a entender que se realizó el mismo enfoque del aprendizaje

con pocas sesiones. Estos hallazgos resaltan la importancia de no solo considerar la arquitectura, sino también otros elementos como el proceso de recolección de datos y la manipulación de datos.

Por otro lado, en una investigación desarrollada por Zhang et al. (2023) se logró una precisión de clasificación del 92.2% al abordar la clasificación de imágenes. Este resultado fue lo más parecido al resultado de precisión que obtuvimos con nuestro algoritmo propuesto, el cual fue del 92.5%, similar al obtenido por nuestro algoritmo, lo que sugiere la influencia positiva de la arquitectura utilizada para este escenario, en particular, el empleo de redes siamesas y algoritmos de cálculo de distancias. Este hallazgo refuerza la idea de que la elección cuidadosa de la arquitectura puede ser determinante en el rendimiento final del algoritmo.

En conclusión, la investigación en reconocimiento de imágenes se caracteriza por la diversidad de enfoques y técnicas empleadas. Aunque algunos estudios han alcanzado precisiones notables, la elección de la arquitectura, el método de entrenamiento y el manejo de datos son aspectos críticos que impactan significativamente en los resultados. Considerar estas variables de manera integral es esencial para avanzar en la mejora continua de los algoritmos de reconocimiento de imágenes.

Por otro lado, en el trabajo llevado a cabo por Kim (2022) donde se utilizó el algoritmo Single Shot Multibox Detector (SSD) de aprendizaje en una sesión. Los resultados mostraron una precisión del 100% cuando se probaron con 20 fotografías aleatorias de rostros humanos y una alta precisión del 90% cuando se probó a través de una cámara web en tiempo real, mientras que a diferencia de los resultados de precisión de los autores en nuestra investigación se obtuvo el 92.5% de precisión con 40 imágenes de rostros humanos probadas aleatoriamente.

Por otro lado, la investigación aplicada sobre el desarrollo de un algoritmo de reconocimiento facial presenta fortalezas en su enfoque práctico y colaborativo. Sin embargo, también debe abordar desafíos potenciales, como la posible limitación en la generalización. La relevancia de esta investigación en el contexto científico-social se manifiesta en diversos aspectos. En primer lugar, el impacto en la seguridad y la privacidad de las personas es innegable. Además, el avance tecnológico que

representa este tipo de investigación contribuye al progreso general en el campo de la inteligencia artificial y la informática.

La contribución al desarrollo tecnológico es otro aspecto relevante, ya que la implementación exitosa de un algoritmo de reconocimiento facial puede tener aplicaciones prácticas en diversos sectores, desde la mejora de la accesibilidad hasta la optimización de procesos en entornos empresariales. Esta aplicación práctica demuestra la conexión directa entre la investigación científica y las soluciones tangibles que pueden beneficiar a la sociedad en su conjunto.

V. CONCLUSIONES

Las conclusiones de la investigación fueron las siguientes:

1. El algoritmo propuesto obtuvo una precisión del 92.5%, frente a la precisión obtenida por el algoritmo de pérdida del triplete, que fue del 95%.
2. El algoritmo propuesto obtuvo un tiempo promedio de reconocimiento facial de 0.96525 milisegundos, frente al tiempo obtenido por el algoritmo de pérdida del triplete, que fue de 0.7025 milisegundos.
3. En resumen, el algoritmo propuesto no obtuvo mejores resultados en cuanto al tiempo de reconocimiento facial respecto al algoritmo de pérdida del triplete. De igual forma, el algoritmo propuesto no alcanzó una precisión mayor que el algoritmo de pérdida del triplete.

VI. RECOMENDACIONES

Las recomendaciones para futuras investigaciones son las siguientes:

- Contar con herramientas para evaluar el desempeño del modelo como una matriz de confusión para poder tener una idea de cómo va nuestro entrenamiento.
- Cuando se esté utilizando redes siamesas para el reconocimiento facial lo conveniente será tener una cantidad de variaciones y que no tenga un fondo de oclusión.
- Si se trabaja con un propio dataset es conveniente cambiar los colores RGB para trabajar solo con blanco y negro y aumentar las variaciones de brillo para resaltar las formas de la imagen ya que si se trabaja con imágenes de colores al momento de realizar la convolución se requerirá 3 veces más poder de procesamiento que trabajando con imágenes en blanco y negro.
- Al momento de diseñar la arquitectura, debemos de tener en cuenta la cantidad de capas repetidas y sobre todo el tipo de activación final ya que este será el encargado de enviar las actualizaciones a los pesos de capas anteriores.
- Cuando se trabaje con redes densas considerar aplicar dropout para desactivar los nodos que no estén en el flujo del correcto funcionamiento de la red.
- Cuando se trabaje con imágenes faciales y se entrene con imágenes verticales el modelo tendrá dificultad para predecir resultados en los que el individuo se encuentre con el rostro hacia un lado, para solucionar este

problema es conveniente usar la normalización y aplicarlo a la última capa de nuestra red.

- Cuando no se disponga de una gran cantidad de datos y solo contemos con nuestro conjunto de datos de pocas imágenes, será recomendable utilizar el generador de Keras. Esto nos permitirá generar nuevos datos al pasar un kernel por cada imagen que tengamos.
- Cuando se tenga que llevar el modelo entrenado para producción se recomienda generar un archivo en formato “h5”, ya que no se tendrá que agrupar los pesos y la arquitectura para poner el modelo en funcionamiento.
- Cuando se tenga cargado el conjunto de datos, es conveniente exportarlo en formato “npy”, ya que nos permite tener un conjunto de datos portable y sin la necesidad de realizar el pre-procesado de la imagen.
- Finalmente se recomienda utilizar el diseño de tipo pre experimental ya que son relativamente rápidos de implementar, lo que puede ser beneficioso cuando se necesita obtener resultados preliminares de manera rápida o cuando hay limitaciones de tiempo.

REFERENCIAS

- Adithama, S. P., Maslim, M. P., & Gunawan, R. P. (2023). Implementation of Face Recognition for Patient Identification Using the Transfer Learning Method. *TEM Journal*, 12(2), 775-784. <https://doi.org/10.18421/TEM122-22>
- Ahmed, D. M., & Mahmood, B. S. (2023). Integration of Face and Gait Recognition via Transfer Learning: A Multiscale Biometric Identification Approach. *Traitement du Signal*, 40(5), 2179-2190.
- Ali, M. A. S., Eltoukhy, M. M., Fathimathul Rajeena, P. P., Gaber, T. (2023). Efficient thermal face recognition method using optimized curvelet features for biometric authentication. *PLoS ONE*, 18(6 June), Article No. e0287349. Cited 1 time.
- Anil, J. A., & Padma Suresh, L. (2023). A novel fast hybrid face recognition approach using Convolutional Kernel Extreme Learning Machine with HOG feature extractor. *Measurement: Sensors*, 30, Article No. 100907.
- AT&T Laboratories Cambridge. (2001). Dataset faces from <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>
- Aufar, Y., & Sitanggang, I. S. (2022). Face recognition based on Siamese convolutional neural network using Kivy framework. *Indonesian Journal of Electrical Engineering and Computer Science*, 26(2), 764-772. Cited 4 times.
- Benradi, H., Chater, A., & Lasfar, A. (2023). A hybrid approach for face recognition using a convolutional neural network combined with feature extraction techniques. *IAES International Journal of Artificial Intelligence*, 12(2), 627-640. Cited 7 times.
- Berrahal, M., & Azizi, M. (2023). Toward Multi-label Attribute Estimation on Multiple Faces Using CNN. *Lecture Notes in Electrical Engineering*, 954, 99-107. Cited 1 time.
- Bigun, J., Hernandez-Diaz, K., Alonso-Fernandez, F. (2023). One-Shot Learning for Periocular Recognition: Exploring the Effect of Domain Adaptation and Data Bias on Deep Representations. *IEEE Access*, 11, 100396-100413.
- Bunge, M. (2007). *La ciencia. Su Método y su filosofía*. Buenos Aires, Argentina: Laetoli

- Cadena, J. (2021). Técnica eficiente para reconocimiento facial global utilizando wavelets y máquinas de vectores de soporte en imágenes 3D. [Tesis de doctorado, Universidad Nacional Mayor de San Marcos, Facultad Ingeniería de Sistemas e Informática, Unidad de Posgrado]. Repositorio institucional Cybertesis UNMSM.
- Cardaioli, M., Conti, M., Orazi, G., Tricomi, P. P., & Tsudik, G. (2023). BLUFADER: Blurred face detection & recognition for privacy-friendly continuous authentication. *Pervasive and Mobile Computing*, 92, Article No. 101801.
- Chatterjee, R., Chatterjee, A., Islam, S.K.H., & Khan, M.K. (2023). An object detection-based few-shot learning approach for multimedia quality assessment. *Multimedia Systems*, 29(5), 2899-2912.
- Chiang, H.-H., Deng, Z.-Y., Kang, L.-W., & Li, H.-C. (2023). A lightweight deep learning model for real-time face recognition. *IET Image Process.*, 17, 3869–3883. <https://doi.org/10.1049/ipr2.12903>
- Conti, M., Orazi, G., Tricomi, P. P., & Tsudik, G. (2023). BLUFADER: Blurred face detection & recognition for privacy-friendly continuous authentication. *Pervasive and Mobile Computing*, 92, Article No. 101801.
- Djamaluddin, M., Munir, R., Utama, N. P., & Kistijantoro, A. I. (2023). Open- Set Profile-to-Frontal Face Recognition on a Very Limited Dataset. *IEEE Access*, 11, 65787-65797.
- El Madmoune, Y., El Ouariachi, I., Zenkouar, K., & Zahi, A. (2023). Robust face recognition using convolutional neural networks combined with Krawtchouk moments. *International Journal of Electrical and Computer Engineering*, 13(4), 4052-4067.
- Fathian, R., Phan, S., Ho, C., & Rouhani, H. (2023). Face touch monitoring using an instrumented wristband using dynamic time warping and k-nearest neighbours. *PLoS ONE*, 18(2), Article No. e0281778.
- Frochte, J., & Mohr, J. (2023). One-Shot Identification with Different Neural Network Approaches. *Studies in Computational Intelligence*, 1119, 205-222.

- Gunawan, R. P., Maslim, M. P., & Adithama, S. P. (2023). Implementation of Face Recognition for Patient Identification Using the Transfer Learning Method. *TEM Journal*, 12(2), 775-784. <https://doi.org/10.18421/TEM122-22>
- Hayale, W., Negi, P. S., & Mahoor, M. H. (2023). Deep Siamese Neural Networks for Facial Expression Recognition in the Wild. *IEEE Transactions on Affective Computing*, 14(2), 1148-1158. Cited 9 times.
- Hernández, R., Fernández, C., & Baptista, P. (2014). *Metodología de la investigación* (6.a ed.). México D.F.: McGraw-Hill.
- Hernandez-Diaz, K., Alonso-Fernandez, F., & Bigun, J. (2023). One-Shot Learning for Periocular Recognition: Exploring the Effect of Domain Adaptation and Data Bias on Deep Representations. *IEEE Access*, 11, 100396-100413.
- Hindy, H., Tachtatzis, C., Atkinson, R., Bayne, E., & Bellekens, X. (2021). Developing a Siamese Network for Intrusion Detection Systems. In *Proceedings of the 1st Workshop on Machine Learning and Systems, EuroMLSys 2021* (pp. 120-126). Cited 8 times.
- Holkar, A., Walambe, R., & Kotecha, K. (2022). Few-Shot learning for face recognition in the presence of image discrepancies for limited multi-class datasets. *Image and Vision Computing*, 120, Article No. 104420.
- Jiménez-Bravo, D. M., Murciego, Á. L., Mendes, A. S., Silva, L. A., & De La Iglesia, D. H. (2022). Edge Face Recognition System Based on One-Shot Augmented Learning. *International Journal of Interactive Multimedia and Artificial Intelligence*, 7(6), 31-44.
- Kamble, V., Bhargav, R. T., Teja, K. A., Satpute, V., Naveen, C., & Kumar, K. V. (2023). One-Shot Face Recognition. In *2023 2nd International Conference on Paradigm Shifts in Communications Embedded Systems, Machine Learning and Signal Processing, PCEMS 2023*. Cited 1 time.
- Kim, B.-H. (2022). Implementation of Access Control System based on Face Prediction and Face Tracking. *Journal of System and Management Sciences*, 12(2), 373-383.

- Kumar, C., N, Saranya, Priyadharshini, M., E, Derrick, & M, Kaleel. (2023). Face recognition using CNN and Siamese network. *Measurement: Sensors*, 27, 100800. <https://doi.org/10.1016/j.measen.2023.100800>
- Kumar, K. V., Teja, K. A., Bhargav, R. T., Satpute, V., Naveen, C., & Kamble, V. (2023). One-Shot Face Recognition. In *2023 2nd International Conference on Paradigm Shifts in Communications Embedded Systems, Machine Learning and Signal Processing, PCEMS 2023*. Cited 1 time.
- López, P. L. (2004). Población muestra y muestreo. *Punto cero*, 9(08), 69-74.
- Mohr, J., & Frochte, J. (2023). One-Shot Identification with Different Neural Network Approaches. *Studies in Computational Intelligence*, 1119, 205-222.
- Moguel, E. A. R. (2005). *Metodología de la Investigación*. Tabasco, México: Universidad Juárez Autónoma de Tabasco
- Nieto, E. (2018). *Tipos de Investigación*. Universidad Santo Domingo de Guzmán.
- O'Mahony, N., Campbell, S., Carvalho, A., Krpalkova, L., Hernandez, G. V., Harapanahalli, S., Riordan, D., & Walsh, J. (2019). One-shot learning for custom identification tasks: A review. *Procedia Manufacturing*, 38, 186-193
- Rajani, P. K., Patil, K., Marathe, B., Mhaisane, P., & Tundalwar, A. (2023). Heart Disease Prediction using Different Machine Learning Algorithms. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11, 354-359.
- Sahito, A., Frank, E., & Pfahringer, B. (2020). Transfer of pretrained model weights substantially improves semi-supervised image classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12576 LNAI, 433-444.
- Saxena, N., & Varshney, D. (2021). Smart Home Security Solutions using Facial Authentication and Speaker Recognition through Artificial Neural Networks. *International Journal of Cognitive Computing in Engineering*, 2, 154-164.
- Sun, B., Luo, A., Rong, B., He, J., & Yu, L. (2022). One-Shot Face Recognition through a Region Inception ResNet with Modified Triplet Loss. *Journal of Physics: Conference Series*, 2320(1), Article No. 012022.

- Sun, W., Jiang, X., Dou, S., Li, D., Miao, D., Deng, C., & Zhao, C. (2023). Invisible Backdoor Attack With Dynamic Triggers Against Person Re- Identification. *IEEE Transactions on Information Forensics and Security*, 19, 307-319.
- Suryawanshi, Y., Meshram, V., Meshram, V., Patil, K., & Chumchu, P. (2023). Face mask wearing image dataset: A comprehensive benchmark for image-based face mask detection models. *Data in Brief*, 51, Article No. 109755.
- Tsai, T.-H., Tsai, C.-E., & Chi, P.-T. (2023). A one-shot face detection and recognition using deep learning method for access control system. *Signal, Image and Video Processing*, 17(4), 1571-1579.
- Tsudik, G., Tricomi, P. P., Orazi, G., Conti, M., & Cardaioli, M. (2023). BLUFADER: Blurred face detection & recognition for privacy-friendly continuous authentication. *Pervasive and Mobile Computing*, 92, Article No. 101801.
- Wijaya, I. G. P. S., Mulyana, H., Kadriyan, H., Yudhanto, D., & Fa'rifah, R.Y. (2023). The Design of Convolutional Neural Networks Model for Classification of Ear Diseases on Android Mobile Devices. *International Journal on Informatics Visualization*, 7(1), 84-91.
- Wright, C., & Stewart, D. W. (2020). Understanding visual lip-based biometric authentication for mobile devices. *Eurasip Journal on Information Security*, 2020(1), Article No. 3. Cited 9 times.
- Xu, M., Zhang, X., & Zhou, X. (2023). Confidence-Calibrated Face and Kinship Verification. *IEEE Transactions on Information Forensics and Security*, 19(pp. 372-384).
- Yan, H., Hu, Z., & Liu, Y. (2023). Study on the Influence of PCA Pre-Treatment on Pig Face Identification with KNN [PCA 前处理对 KNN 识别猪脸的影响研究]. *INMATEH - Agricultural Engineering*, 70(2), 86-95.
- Zhang, Q., Yi, X., Guo, J., Tang, Y., Feng, T., & Liu, R. (2023). A few-shot rare wildlife image classification method based on style migration data augmentation. *Ecological Informatics*, 77, Article No. 102237.

ANEXOS

Anexo 1 . Detalle de operacionalización de las variables

Variable	Definición Conceptual	Definición Operacional	Dimensiones	Indicador	Instrumento	Escala de Medición
Impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-	El uso del reconocimiento facial se ha convertido en un gran problema, principalmente debido al aumento de situaciones en	Medición del tiempo y precisión del reconocimiento de imágenes	Reconocimiento de imágenes. (Cadena, 2021; Tsai et al., 2023; Lin y Yen, 2023; Deng et al., 2023; Aufar y Sitanggang, 2022; Holkar y Walambe 2022; Cardaioli et al., 2023)	Tiempo de reconocimiento facial. (Adithama et al., 2023; Kim, 2022)	Ficha de registro	Razón

vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales.	las que existen datos nuevos en tiempo real. (Hernandez- Diaz et al., 2023; Xu et al.,2023)	faciales. (Hernandez- Diaz et al., 2023; Xu et al.,2023)	Precisión (Zhang et al., 2023; Chatterjee et al., 2023; AUFAR y Sitanggang, 2022; Holkar et al., 2022)	Precisión de reconocimiento facial. (Djamaluddin et al., 2023; Sun et al., 2022; Saxena y de Varshney., 2021; Hindy et al., 2021)	Ficha de registro	Razón
--	---	--	--	---	-------------------	-------

Anexo 2 . Ficha de recolección de precisión del algoritmo del triplete vs precisión del algoritmo propuesto

Clase	Precisión Triplete (VP=1, FP=0)	Precisión Algoritmo propuesto (VP=1, FP=0)
1	1	1
2	1	1
3	1	1
4	1	1
5	1	1
6	1	1
7	1	0
8	1	1
9	1	0
10	1	1
11	1	1
12	1	1
13	1	1
14	1	1
15	1	1
16	1	0
17	1	1
18	1	1
19	1	1
20	1	1
21	1	1
22	1	1
23	1	1
24	1	1
25	0	1
26	1	1
27	1	1
28	1	1
29	1	1
30	1	1
31	0	1
32	1	1
33	1	1
34	1	1
35	1	1
36	1	1
37	1	1
38	1	1
39	1	1
40	1	1
TOTAL	95%	92.5%

Anexo 3 . Ficha de recoleccion de tiempo de reconocimiento facial del algoritmo del triplete vs el tiempo de reconocimiento del algoritmo propuesto

Clase	Tiempo(Algoritmo del triplete)	Tiempo(Algoritmo Propuesto)
1	0.69	1.16
2	0.66	1.17
3	0.65	1.16
4	0.65	1.17
5	0.66	0.83
6	0.65	0.83
7	0.64	0.84
8	0.68	0.84
9	0.64	0.87
10	0.64	0.86
11	0.82	0.90
12	0.83	1.44
13	0.76	0.84
14	0.66	0.84
15	0.66	1.18
16	0.66	1.16
17	0.66	1.15
18	0.68	1.18
19	0.65	0.87
20	0.67	0.82
21	0.68	0.85
22	0.66	0.83
23	0.81	0.82
24	0.83	0.81
25	0.82	0.82
26	0.73	0.81
27	0.69	0.82
28	0.69	0.85
29	0.69	0.82
30	0.70	1.15
31	0.71	1.97
32	0.67	1.18
33	0.68	0.97
34	0.67	0.84
35	0.68	0.81
36	0.66	0.83
37	0.69	0.84
38	0.80	0.83
39	0.82	0.83
40	0.81	0.82
TOTAL	0.7025	0.96525

Anexo 4. Reporte de similitud en software Turnitin

Tesis de investigación

INFORME DE ORIGINALIDAD

11 %	10 %	2 %	2 %
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	repositorio.ucv.edu.pe Fuente de Internet	4 %
2	hdl.handle.net Fuente de Internet	1 %
3	www.coursehero.com Fuente de Internet	1 %
4	repositorio.une.edu.pe Fuente de Internet	<1 %
5	www.researchgate.net Fuente de Internet	<1 %
6	dspace.unach.edu.ec Fuente de Internet	<1 %
7	pesquisa.bvsalud.org Fuente de Internet	<1 %
8	Submitted to Universidad Cesar Vallejo Trabajo del estudiante	<1 %
9	Submitted to Universidad Rey Juan Carlos Trabajo del estudiante	<1 %

10	Submitted to uaq Trabajo del estudiante	<1 %
11	Adriano di Paco, Bruno-Pierre Dubé, Pierantonio Laveneziana. "Cambios en la respuesta ventilatoria al esfuerzo en deportistas entrenados: efectos beneficiosos sobre la fisiología respiratoria más allá del rendimiento cardiovascular", Archivos de Bronconeumología, 2017 Publicación	<1 %
12	repositorio.continental.edu.pe Fuente de Internet	<1 %
13	www.fulp.ulpgc.es Fuente de Internet	<1 %
14	Millones, Jonathan Andre Manrique Vasquez, Victor Steve Ramos Maguina, Milagros Mercedes Uriol. "Gestion de la Innovacion y Tecnologia en Empresas Peruanas de la Industria Quimica Pertenecientes al Sector Manufactura.", Pontificia Universidad Catolica del Peru - CENTRUM Catolica (Peru), 2021 Publicación	<1 %
15	www.eye-on.ai Fuente de Internet	<1 %
16	Submitted to Consorcio CIXUG Trabajo del estudiante	<1 %

17	Submitted to Universidad de Almeria Trabajo del estudiante	<1 %
18	repositorio.upn.edu.pe Fuente de Internet	<1 %
19	baixardoc.com Fuente de Internet	<1 %
20	elreferente.es Fuente de Internet	<1 %
21	export.arxiv.org Fuente de Internet	<1 %
22	lareferencia.info Fuente de Internet	<1 %
23	rstudio-pubs-static.s3.amazonaws.com Fuente de Internet	<1 %
24	whotalking.com Fuente de Internet	<1 %
25	www.grafiati.com Fuente de Internet	<1 %
26	www.maec.es Fuente de Internet	<1 %
27	zenodo.org Fuente de Internet	<1 %
28	americanae.aecid.es Fuente de Internet	<1 %

29	docplayer.es Fuente de Internet	<1 %
30	text-id.123dok.com Fuente de Internet	<1 %
31	www.acenologia.com Fuente de Internet	<1 %
32	www.armis.com Fuente de Internet	<1 %
33	www.reduc.cl Fuente de Internet	<1 %
34	www.scoop.it Fuente de Internet	<1 %
35	www.scribd.com Fuente de Internet	<1 %
36	www.slideshare.net Fuente de Internet	<1 %
37	Submitted to Universidad Nacional Federico Villarreal Trabajo del estudiante	<1 %
38	developers.google.com Fuente de Internet	<1 %
39	navegandaluz.com Fuente de Internet	<1 %
40	rmetropolitanachile.blogspot.com Fuente de Internet	<1 %

<1 %

41 **stanford.edu**
Fuente de Internet

<1 %

42 **tr-ex.me**
Fuente de Internet

<1 %

43 **www.dspace.espol.edu.ec**
Fuente de Internet

<1 %

44 **www.quaibranly.fr**
Fuente de Internet

<1 %

Excluir citas Apagado

Excluir coincidencias Apagado

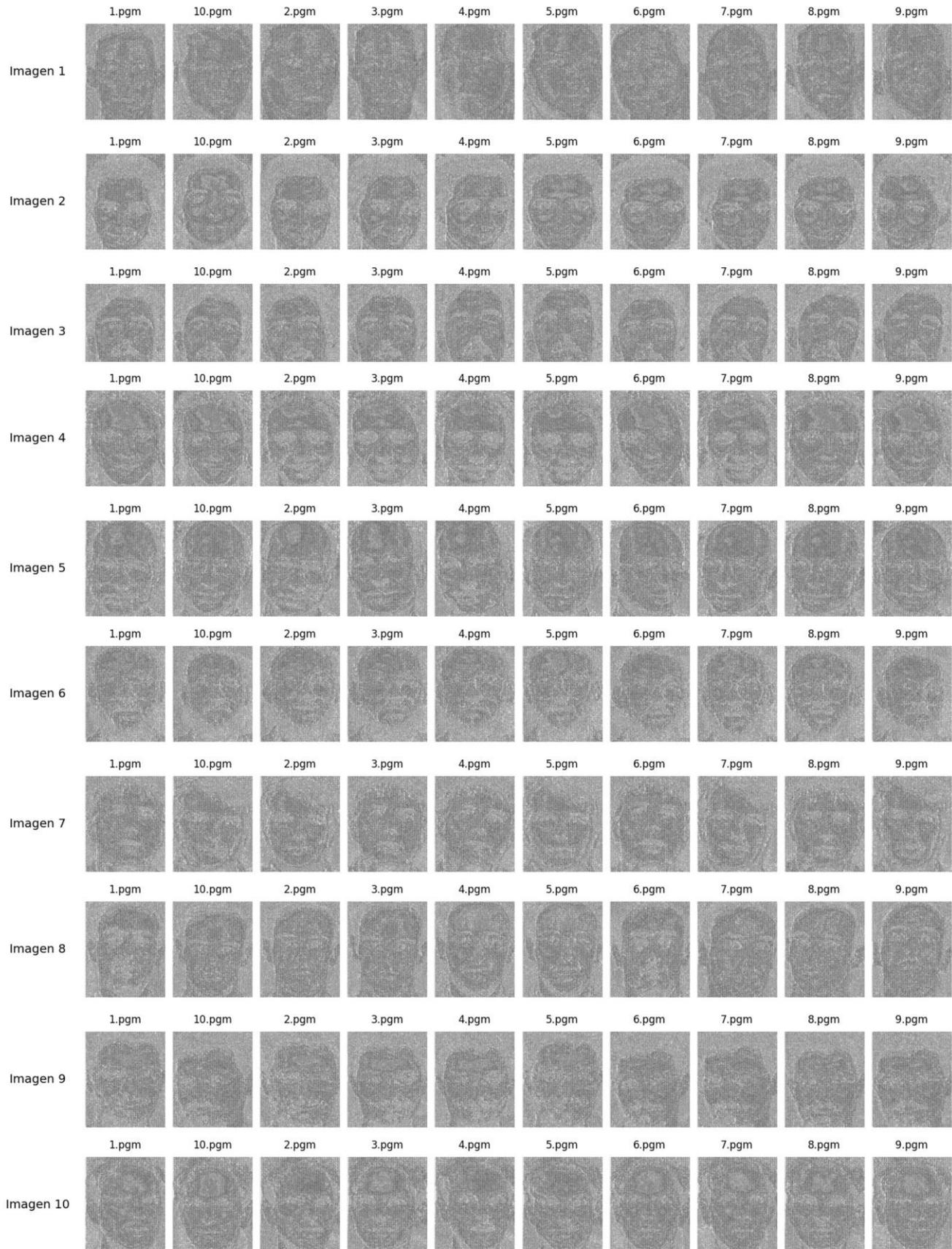
Excluir bibliografía Apagado

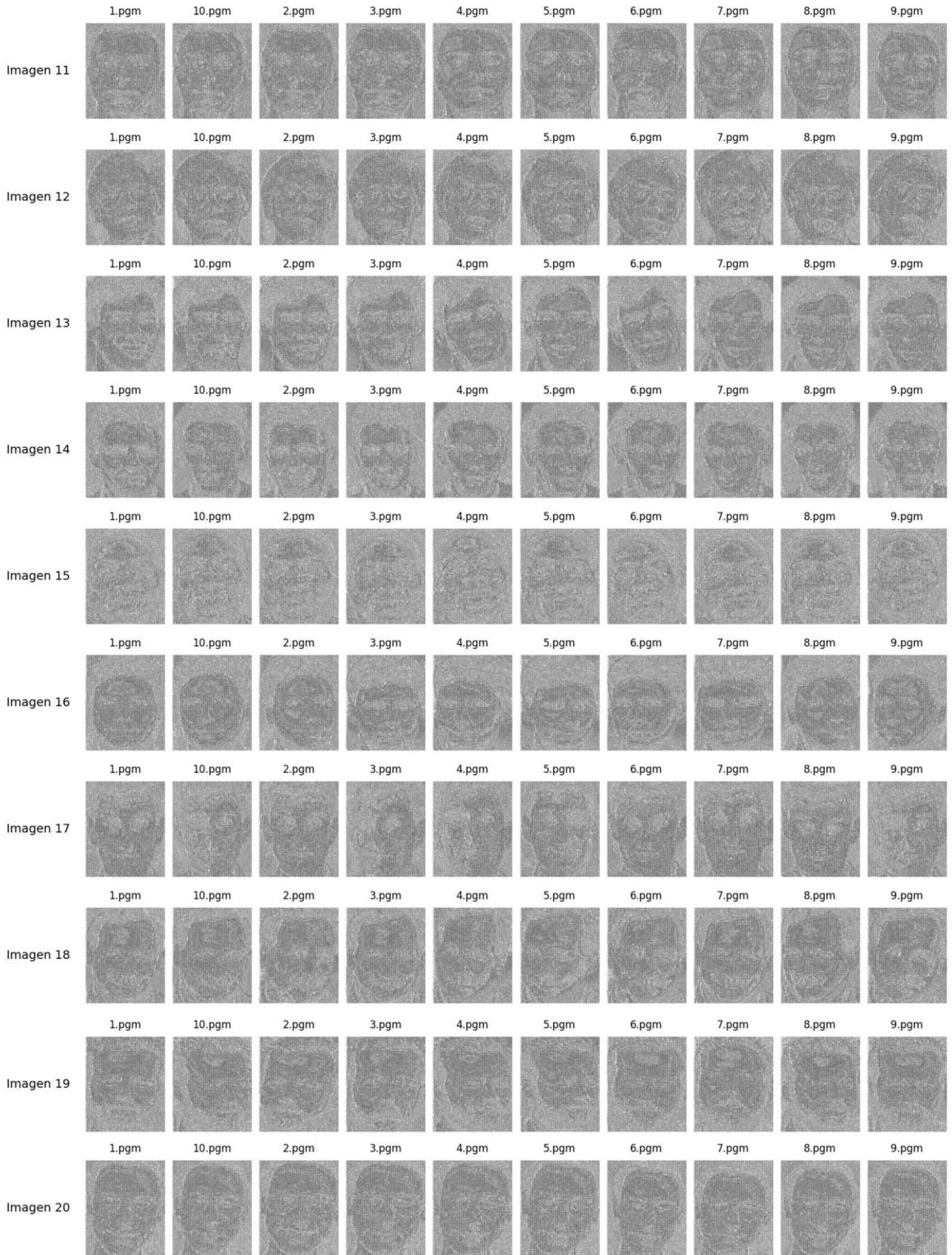
Anexo 5. Matriz de consistencia

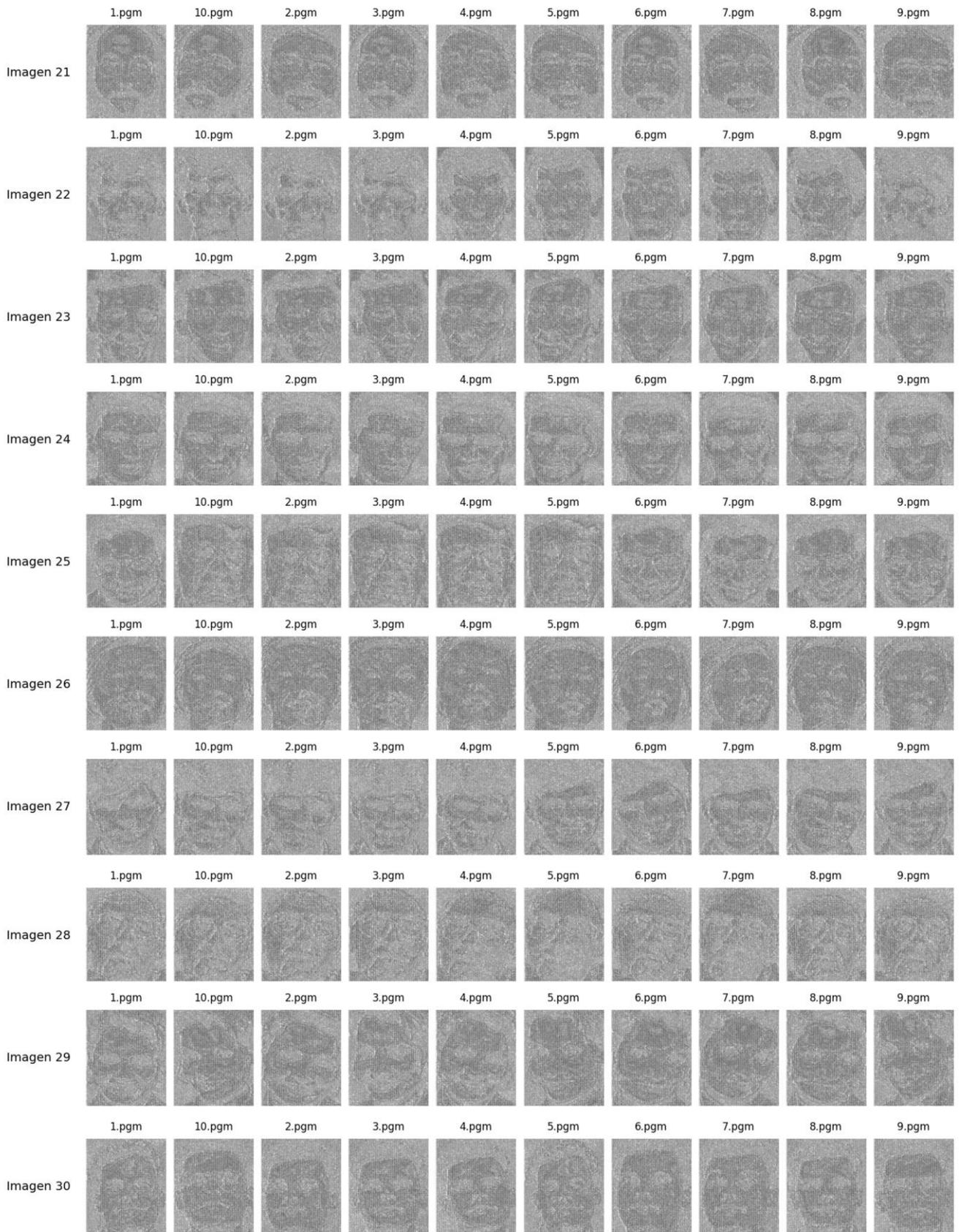
PROBLEMAS	OBJETIVOS	HIPOTESIS	VARIABLE	DIMENSIONES	INDICADORES
GENERAL	GENERAL	GENERAL			
¿Cuál será el impacto que tendrá la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales?	Determinar el impacto que tendrá la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales.	La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión incrementó la precisión y redujo el tiempo de reconocimiento de imágenes faciales.(Bhogale et al. 2018)	-	-	-
ESPECÍFICOS	ESPECÍFICOS	ESPECÍFICOS			INDICADORES
¿Cuál será el impacto de la implementación de un Algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más	Determinar el impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos	La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión redujo el tiempo de		Reconocimiento de imágenes. (Cadena, 2021; Tsai et al.,	Tiempo de reconocimiento facial.

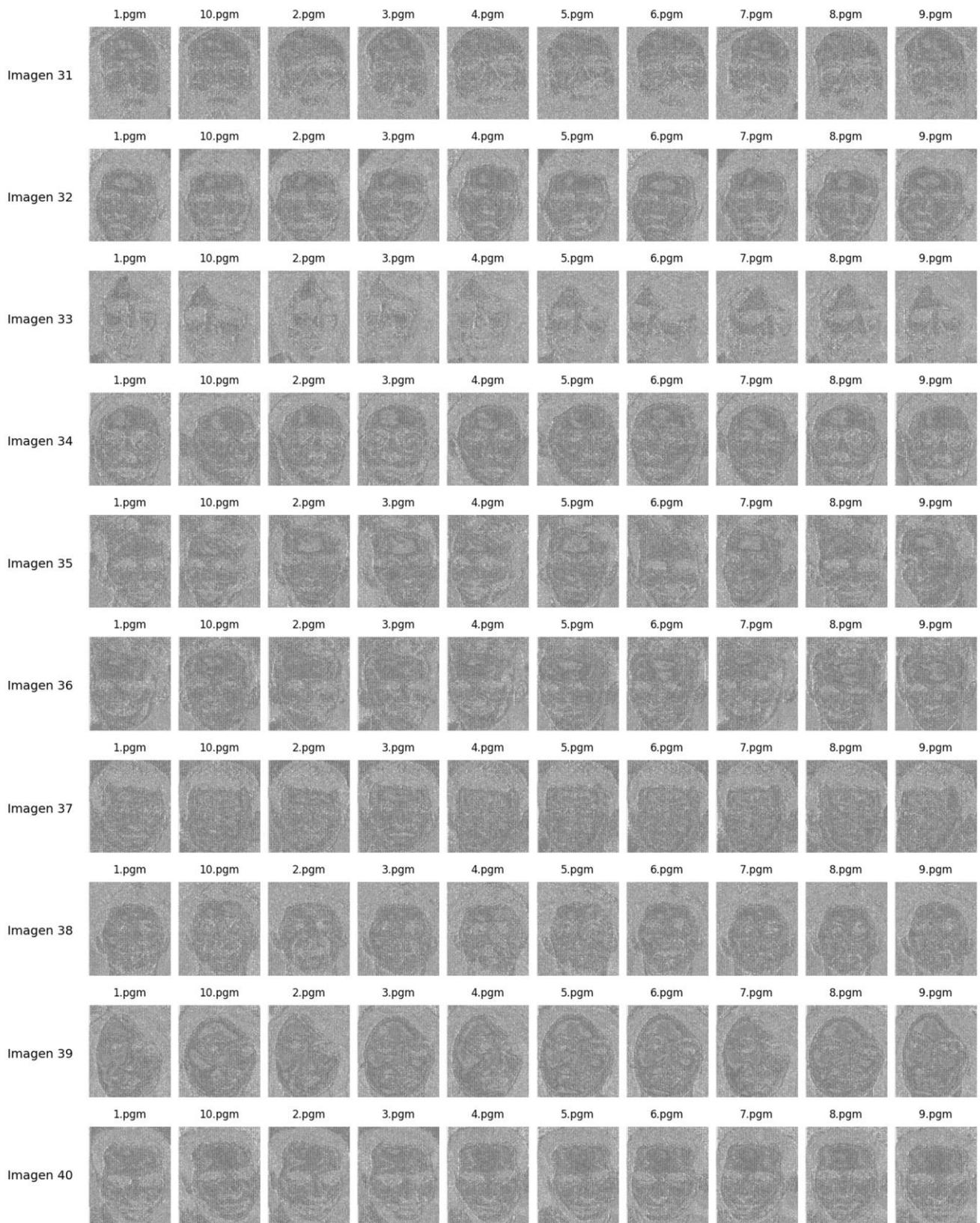
<p>cercanos y en el aprendizaje de una sesión para la reducción del tiempo de reconocimiento de imágenes faciales?</p>	<p>y el aprendizaje de una sesión para la reducción del tiempo de reconocimiento de imágenes faciales.</p>	<p>reconocimiento de imágenes faciales. (Ding, Guo, Zhang y Fu, 2018).</p>	<p>Impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales.</p>	<p>2023; Lin y Yen, 2023; Deng et al., 2023; AUFAR y Sitanggang, 2022; Holkar y Walambe 2022; Cardaioli et al., 2023)</p>	<p>(Adithama et al., 2023; Kim, 2022)</p>
<p>¿Cuál será el impacto de la implementación de un Algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y en el aprendizaje de una sesión para el incremento de precisión en el reconocimiento de imágenes faciales?</p>	<p>Determinar el impacto de la implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión para la precisión del reconocimiento de imágenes faciales</p>	<p>La implementación de un algoritmo basado en una red siamesa, pérdida del triplete, k-vecinos más cercanos y el aprendizaje de una sesión incrementó la precisión en el reconocimiento facial. (Bhogale et al.,2018; Jadhav, Namboodiri y Venkatesh,2016)</p>	<p>Precisión (Zhang et al., 2023; Chatterjee et al., 2023; AUFAR y Sitanggang, 2022; Holkar et al., 2022)</p>	<p>Precisión de reconocimiento facial. (Djamaluddin et al., 2023; Sun et al., 2022; Saxenay de Varshney., 2021; Hindy et al., 2021)</p>	

Anexo 6. Imágenes de la colección de datos ORL





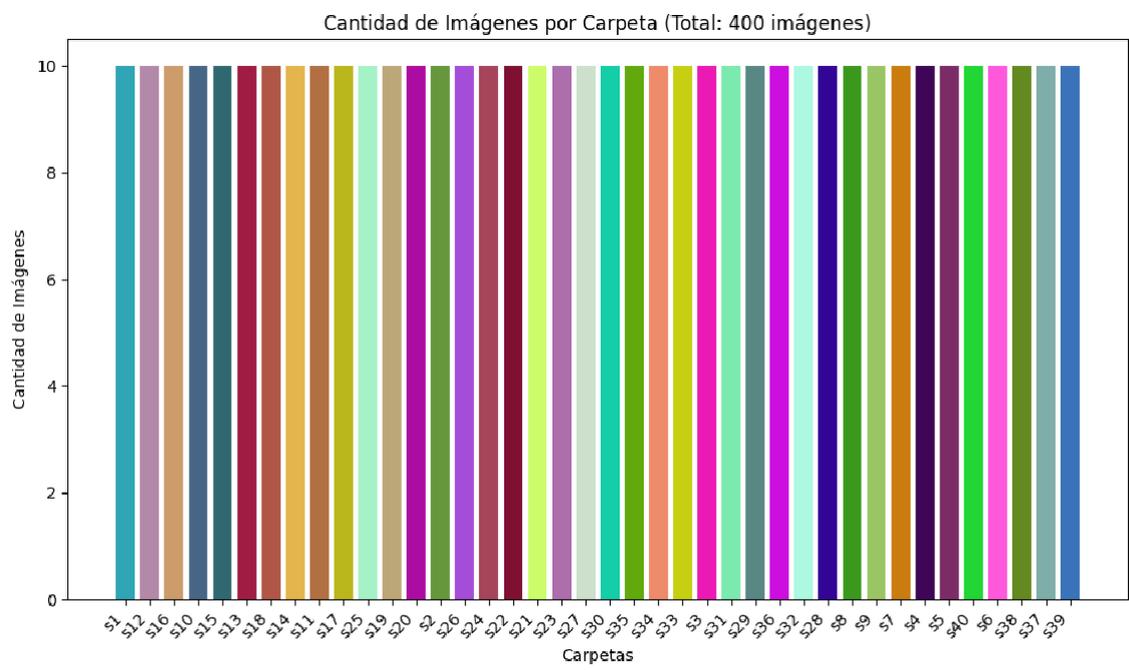




Fuente: Las imágenes fueron recopiladas de AT&T Laboratories Cambridge (2001).

Anexo 7. Distribución de la colección de datos ORL

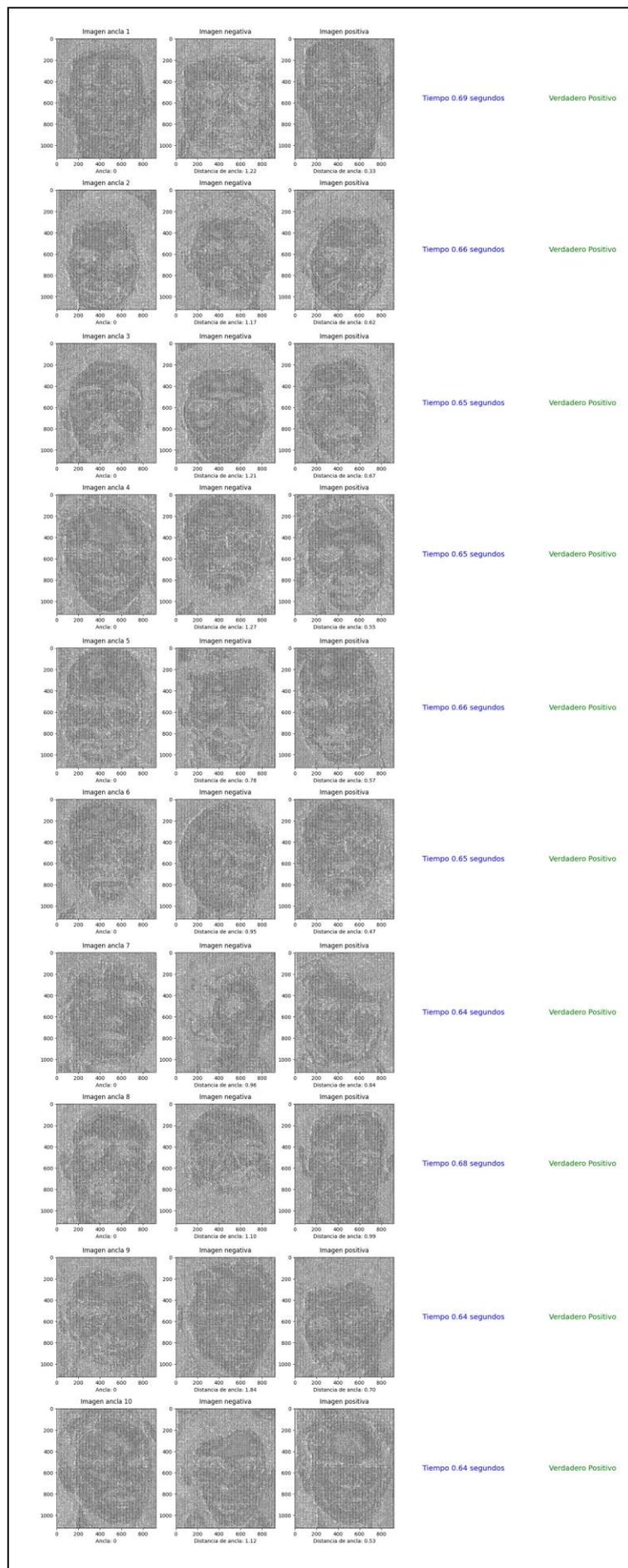
En el siguiente gráfico se puede visualizar como están distribuidas las imágenes que se utilizó para entrenamiento y reconocimiento en esta investigación.



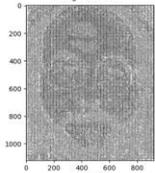
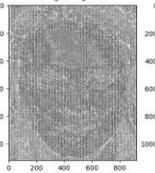
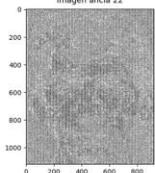
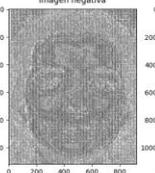
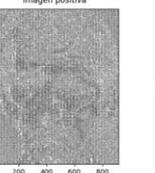
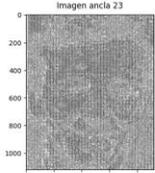
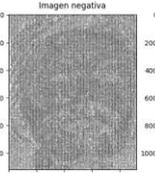
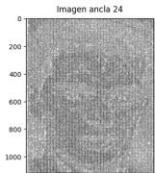
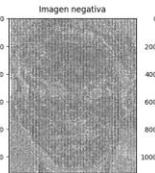
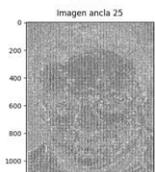
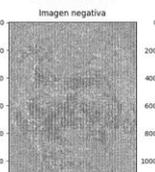
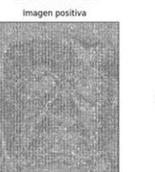
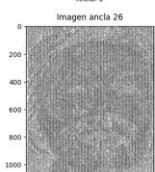
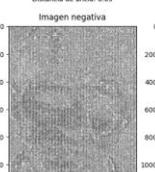
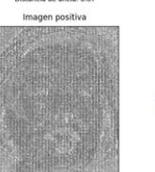
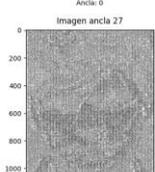
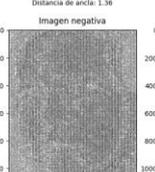
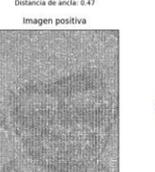
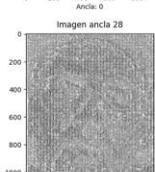
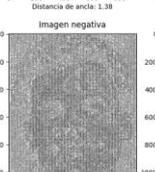
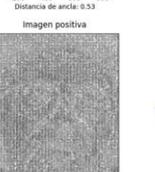
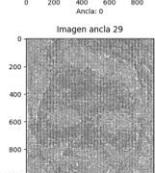
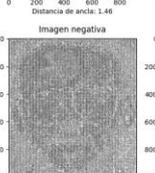
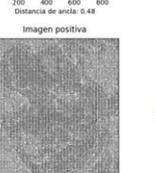
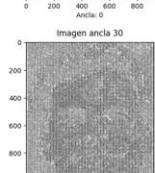
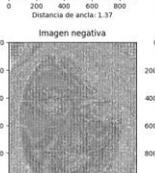
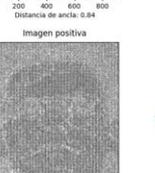
Clases	Cantidad imágenes por clase
1	10
2	10
3	10
4	10
5	10
*	*
*	*
*	*
38	10
39	10
40	10
TOTAL	400

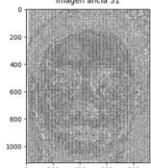
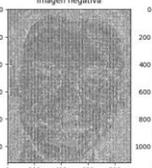
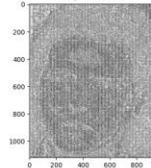
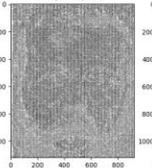
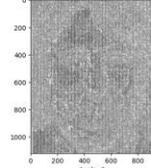
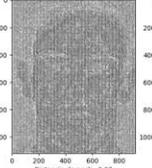
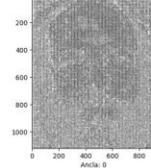
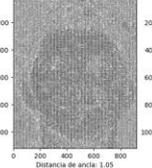
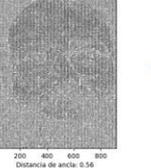
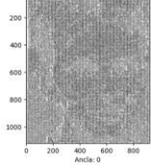
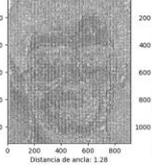
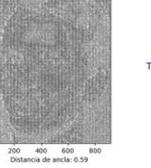
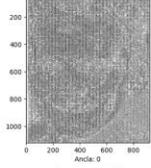
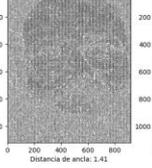
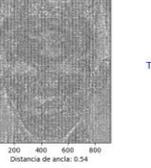
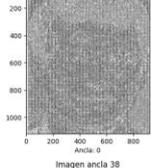
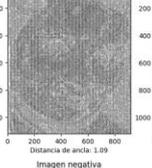
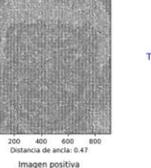
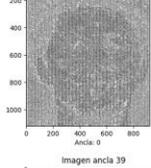
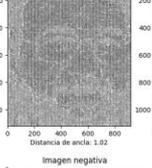
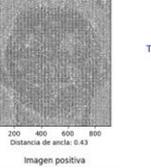
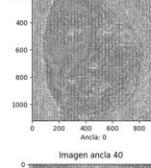
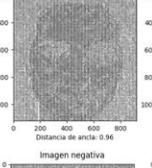
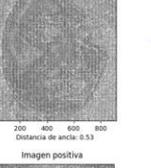
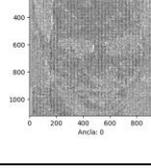
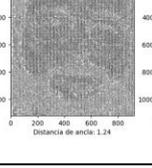
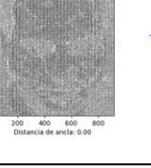
Fuente: Elaboración propia

Anexo 8 . Resultados de test del algoritmo de pérdida del triplete

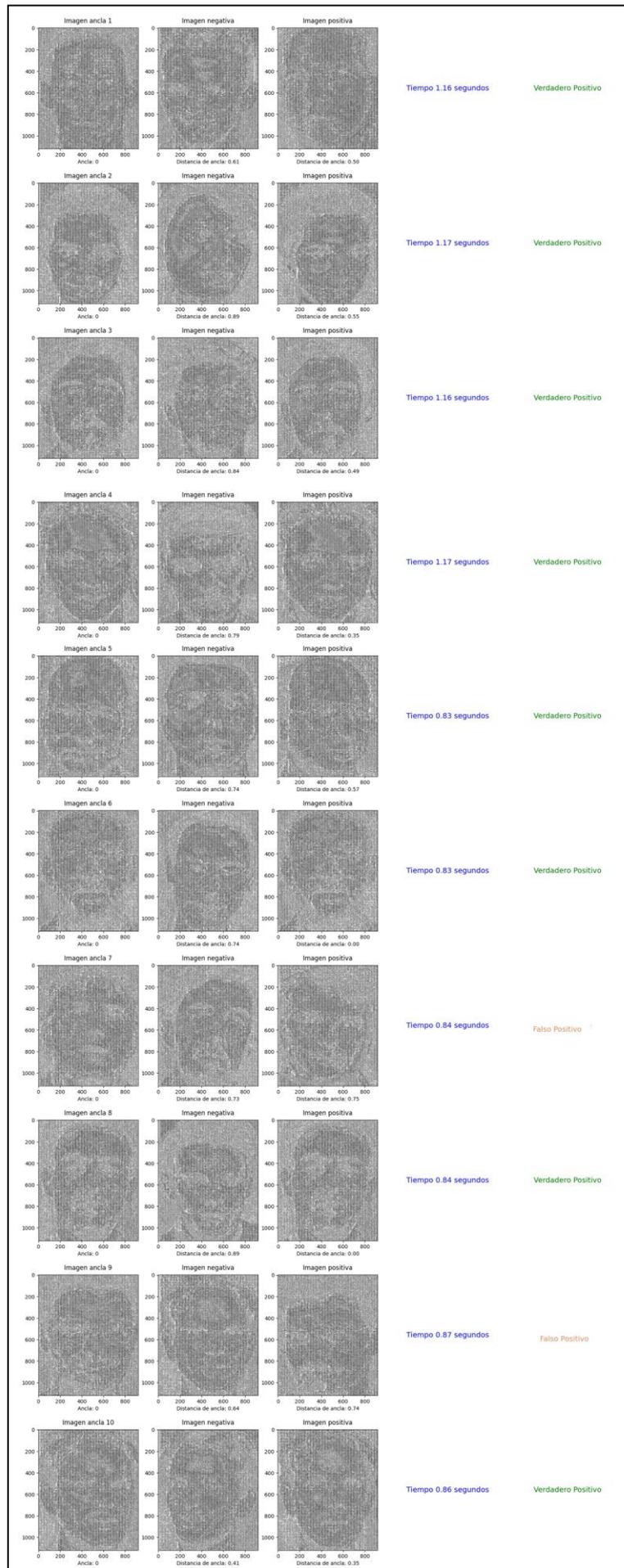


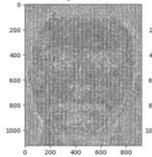
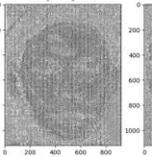
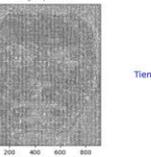
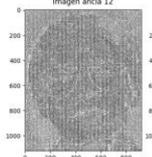
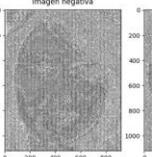
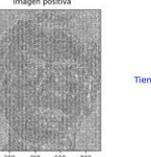
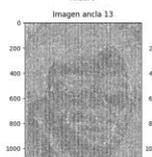
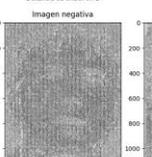
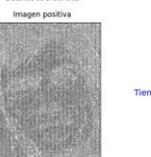
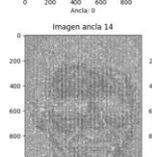
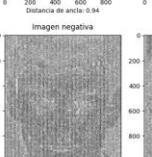
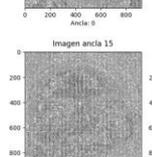
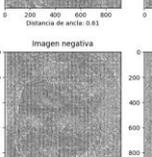
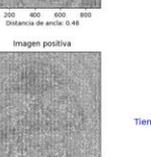
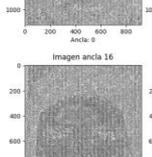
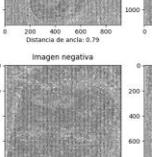
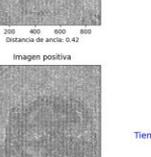
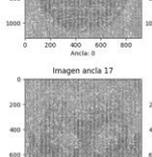
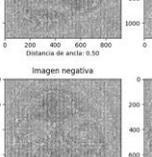
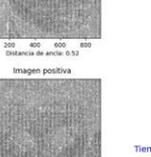
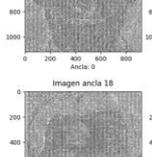
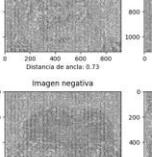
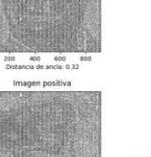
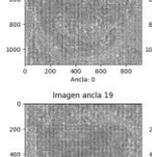
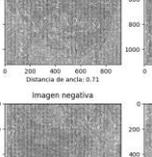
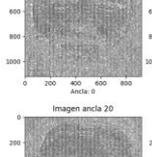
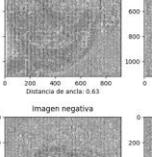
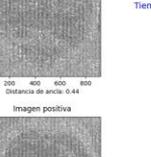
<p>Imagen ancla 11</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 0.95</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.56</p>	Tiempo 0.82 segundos	Verdadero Positivo
<p>Imagen ancla 12</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 0.88</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.46</p>	Tiempo 0.83 segundos	Verdadero Positivo
<p>Imagen ancla 13</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 1.23</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.79</p>	Tiempo 0.76 segundos	Verdadero Positivo
<p>Imagen ancla 14</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 1.23</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.42</p>	Tiempo 0.66 segundos	Verdadero Positivo
<p>Imagen ancla 15</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 0.72</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.50</p>	Tiempo 0.66 segundos	Verdadero Positivo
<p>Imagen ancla 16</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 1.22</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.42</p>	Tiempo 0.66 segundos	Verdadero Positivo
<p>Imagen ancla 17</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 1.04</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.21</p>	Tiempo 0.66 segundos	Verdadero Positivo
<p>Imagen ancla 18</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 0.79</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.50</p>	Tiempo 0.68 segundos	Verdadero Positivo
<p>Imagen ancla 19</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 1.34</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.35</p>	Tiempo 0.65 segundos	Verdadero Positivo
<p>Imagen ancla 20</p> <p>Ancla 0</p>	<p>Imagen negativa</p> <p>Distancia de ancla: 1.38</p>	<p>Imagen positiva</p> <p>Distancia de ancla: 0.00</p>	Tiempo 0.67 segundos	Verdadero Positivo

<p>Imagen ancla 21</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.12</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.35</p>	<p>Tiempo 0.68 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 22</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.18</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.37</p>	<p>Tiempo 0.67 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 23</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.04</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.45</p>	<p>Tiempo 0.81 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 24</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 0.97</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.37</p>	<p>Tiempo 0.83 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 25</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 0.69</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.37</p>	<p>Tiempo 0.82 segundos</p>	<p>Falso Positivo</p>
<p>Imagen ancla 26</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.36</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.47</p>	<p>Tiempo 0.73 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 27</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.38</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.53</p>	<p>Tiempo 0.69 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 28</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.46</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.48</p>	<p>Tiempo 0.69 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 29</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.37</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.84</p>	<p>Tiempo 0.69 segundos</p>	<p>Verdadero Positivo</p>
<p>Imagen ancla 30</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.15</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.19</p>	<p>Tiempo 0.70 segundos</p>	<p>Verdadero Positivo</p>

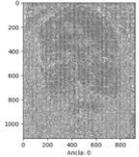
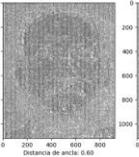
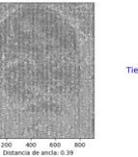
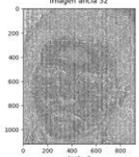
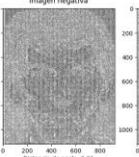
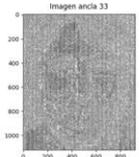
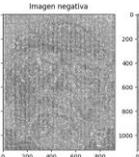
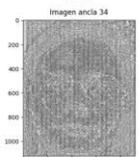
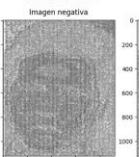
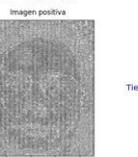
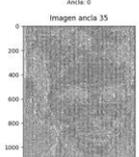
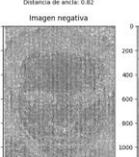
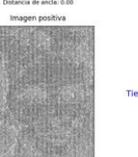
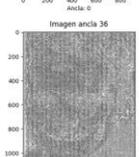
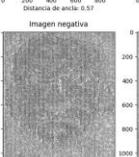
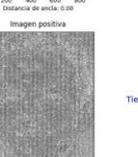
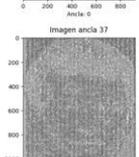
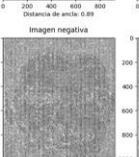
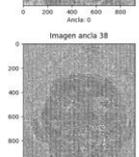
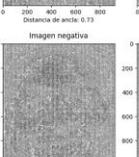
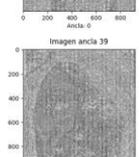
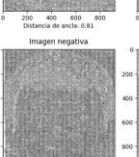
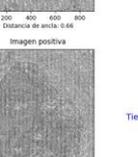
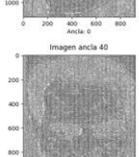
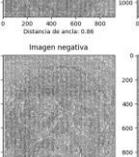
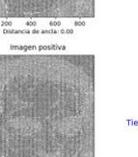
<p>Imagen ancla 31</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 0.79</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.69</p>	Tiempo 0.71 segundos	Falso Positivo
<p>Imagen ancla 32</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 0.84</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.47</p>	Tiempo 0.67 segundos	Verdadero Positivo
<p>Imagen ancla 33</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 0.87</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.37</p>	Tiempo 0.68 segundos	Verdadero Positivo
<p>Imagen ancla 34</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.05</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.56</p>	Tiempo 0.67 segundos	Verdadero Positivo
<p>Imagen ancla 35</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.26</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.59</p>	Tiempo 0.68 segundos	Verdadero Positivo
<p>Imagen ancla 36</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.41</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.54</p>	Tiempo 0.66 segundos	Verdadero Positivo
<p>Imagen ancla 37</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.09</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.47</p>	Tiempo 0.69 segundos	Verdadero Positivo
<p>Imagen ancla 38</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.02</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.43</p>	Tiempo 0.80 segundos	Verdadero Positivo
<p>Imagen ancla 39</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 0.96</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.53</p>	Tiempo 0.82 segundos	Verdadero Positivo
<p>Imagen ancla 40</p>  <p>Ancla: 0</p>	<p>Imagen negativa</p>  <p>Distancia de ancla: 1.24</p>	<p>Imagen positiva</p>  <p>Distancia de ancla: 0.00</p>	Tiempo 0.81 segundos	Verdadero Positivo

Anexo 9. Resultados de test de algoritmo propuesto

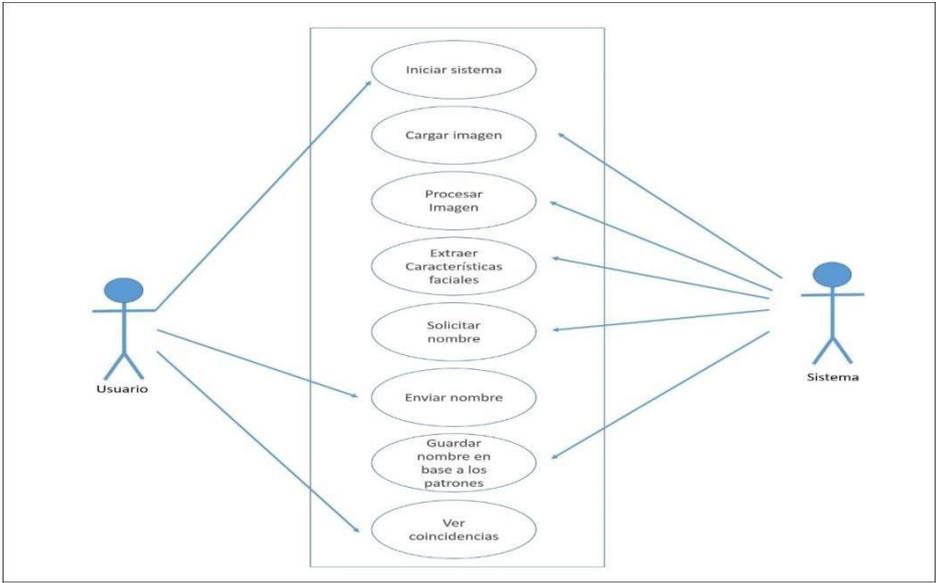


<p>Imagen ancla 11</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 0.90 segundos	Verdadero Positivo
<p>Imagen ancla 12</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 1.44 segundos	Verdadero Positivo
<p>Imagen ancla 13</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 0.84 segundos	Verdadero Positivo
<p>Imagen ancla 14</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 0.84 segundos	Verdadero Positivo
<p>Imagen ancla 15</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 1.18 segundos	Verdadero Positivo
<p>Imagen ancla 16</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 1.16 segundos	Falso Positivo
<p>Imagen ancla 17</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 1.15 segundos	Verdadero Positivo
<p>Imagen ancla 18</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 1.18 segundos	Verdadero Positivo
<p>Imagen ancla 19</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 1.87 segundos	Verdadero Positivo
<p>Imagen ancla 20</p>  <p>Imagen negativa</p>  <p>Imagen positiva</p> 	Tiempo 0.82 segundos	Verdadero Positivo

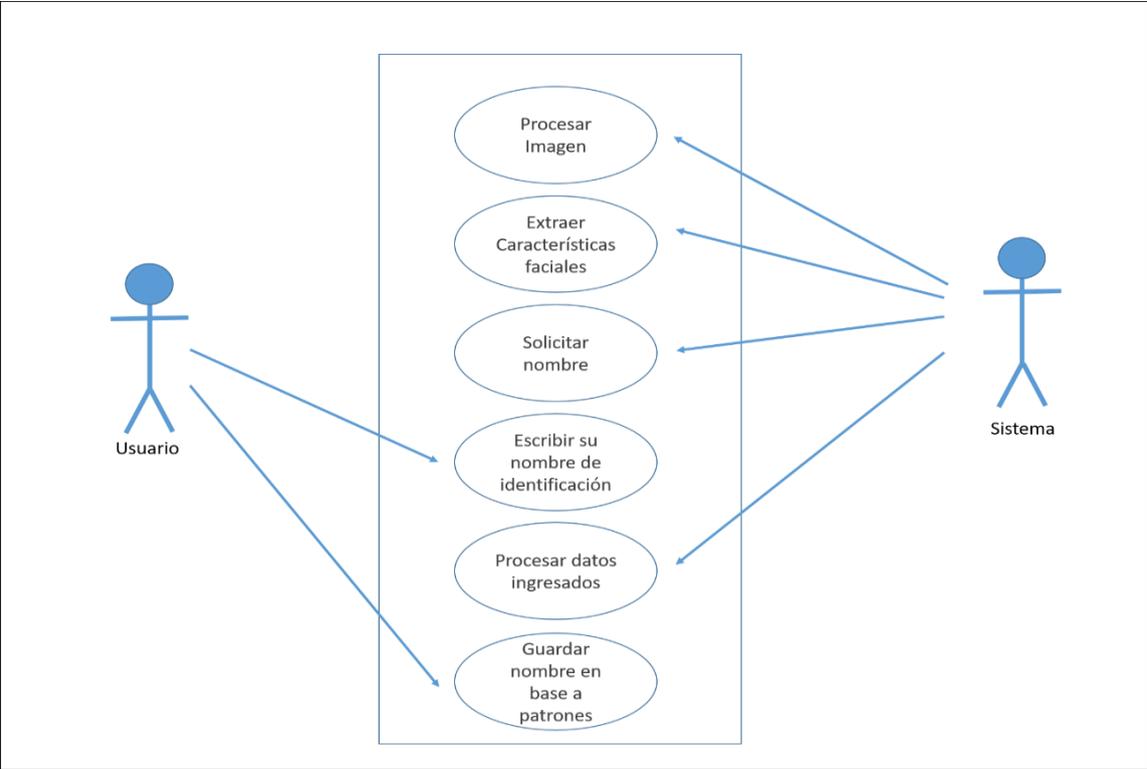
<p>Imagen ancla 21 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.91</p>	<p>Imagen positiva Distancia de ancla: 0.49</p>	Tiempo 0.85 segundos	Verdadero Positivo
<p>Imagen ancla 22 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.74</p>	<p>Imagen positiva Distancia de ancla: 0.59</p>	Tiempo 0.83 segundos	Verdadero Positivo
<p>Imagen ancla 23 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.74</p>	<p>Imagen positiva Distancia de ancla: 0.58</p>	Tiempo 0.82 segundos	Verdadero Positivo
<p>Imagen ancla 24 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.85</p>	<p>Imagen positiva Distancia de ancla: 0.41</p>	Tiempo 0.81 segundos	Verdadero Positivo
<p>Imagen ancla 25 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.76</p>	<p>Imagen positiva Distancia de ancla: 0.41</p>	Tiempo 0.82 segundos	Verdadero Positivo
<p>Imagen ancla 26 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.91</p>	<p>Imagen positiva Distancia de ancla: 0.73</p>	Tiempo 0.81 segundos	Verdadero Positivo
<p>Imagen ancla 27 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.64</p>	<p>Imagen positiva Distancia de ancla: 0.45</p>	Tiempo 0.82 segundos	Verdadero Positivo
<p>Imagen ancla 28 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.77</p>	<p>Imagen positiva Distancia de ancla: 0.54</p>	Tiempo 0.85 segundos	Verdadero Positivo
<p>Imagen ancla 29 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 0.79</p>	<p>Imagen positiva Distancia de ancla: 0.79</p>	Tiempo 0.82 segundos	Verdadero Positivo
<p>Imagen ancla 30 Ancla: 0</p>	<p>Imagen negativa Distancia de ancla: 1.06</p>	<p>Imagen positiva Distancia de ancla: 0.74</p>	Tiempo 1.15 segundos	Verdadero Positivo

  	Tiempo 1.97 segundos	Verdadero Positivo
  	Tiempo 1.18 segundos	Verdadero Positivo
  	Tiempo 0.97 segundos	Verdadero Positivo
  	Tiempo 0.84 segundos	Verdadero Positivo
  	Tiempo 0.81 segundos	Verdadero Positivo
  	Tiempo 0.83 segundos	Verdadero Positivo
  	Tiempo 0.84 segundos	Verdadero Positivo
  	Tiempo 0.83 segundos	Verdadero Positivo
  	Tiempo 0.83 segundos	Verdadero Positivo
  	Tiempo 0.82 segundos	Verdadero Positivo

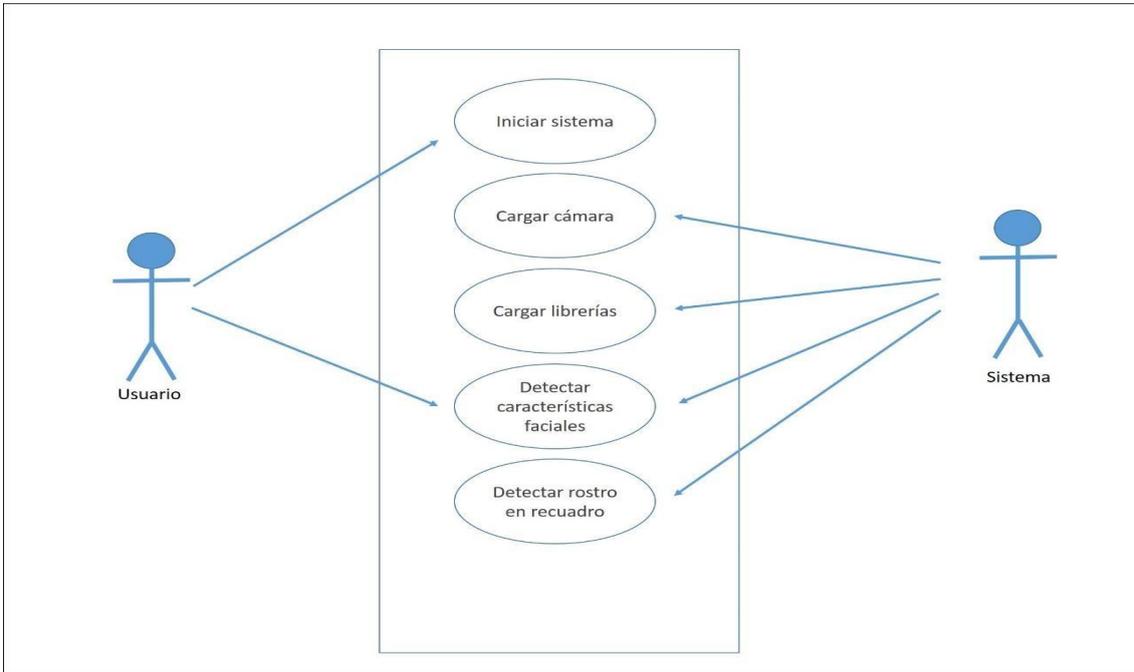
Anexo 10. Diagrama de caso de uso general de aplicación



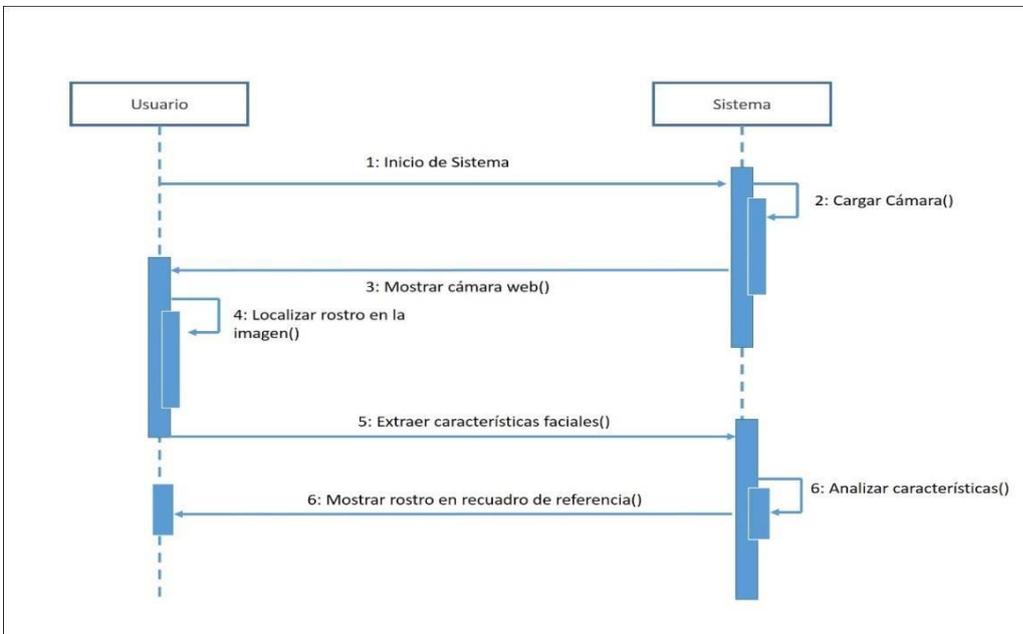
Anexo 11. Diagrama de caso de uso “Guardar imagen” de aplicación



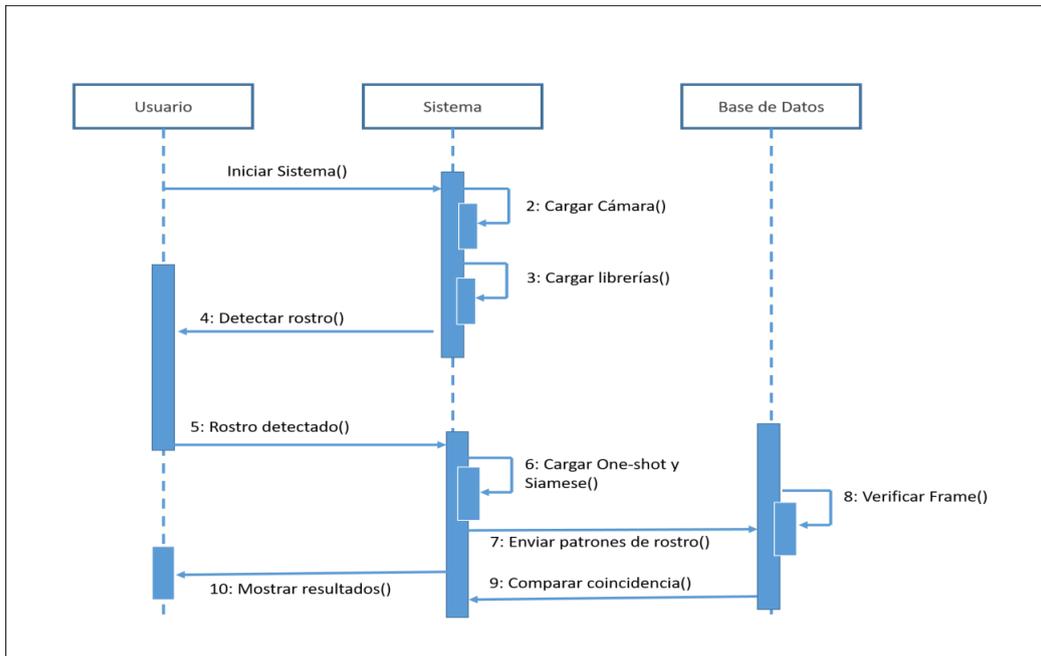
Anexo 12. Diagrama de caso de uso "Detectar Rostro" de aplicación



Anexo 13. Diagrama de secuencia "Detectar Rostro en aplicación"



Anexo 14 . Diagrama de secuencia "Reconocer Rostro" de aplicación



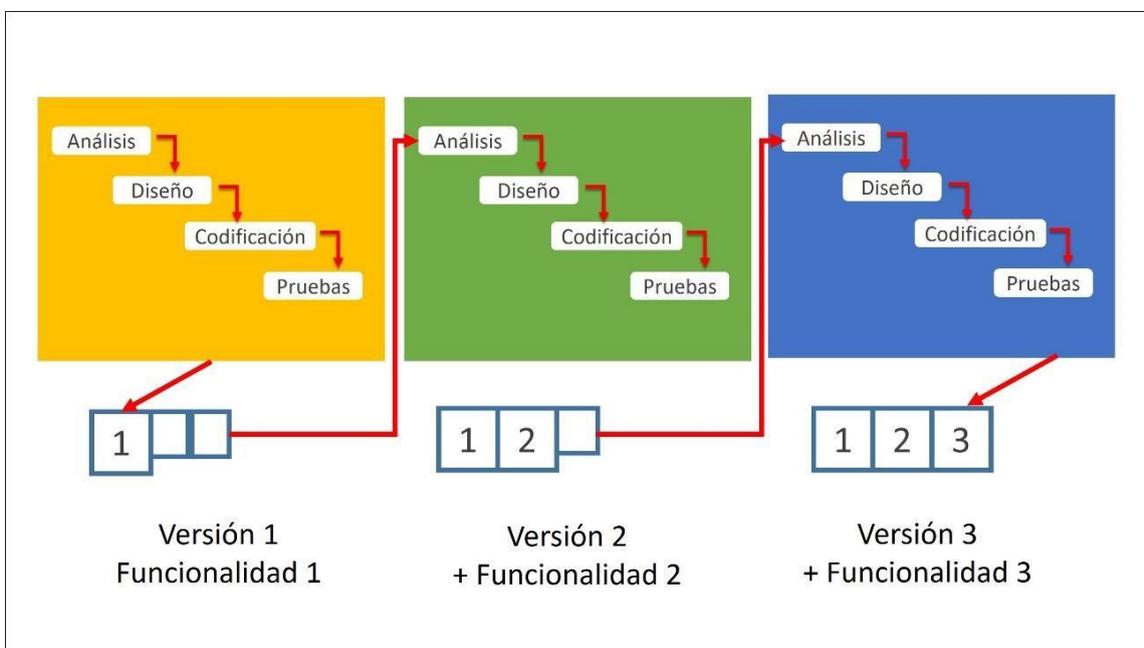
Anexo 15. Herramientas utilizadas para el desarrollo de aplicación

Algunas de las herramientas que hemos utilizado para el desarrollo del proyecto han sido los siguientes:

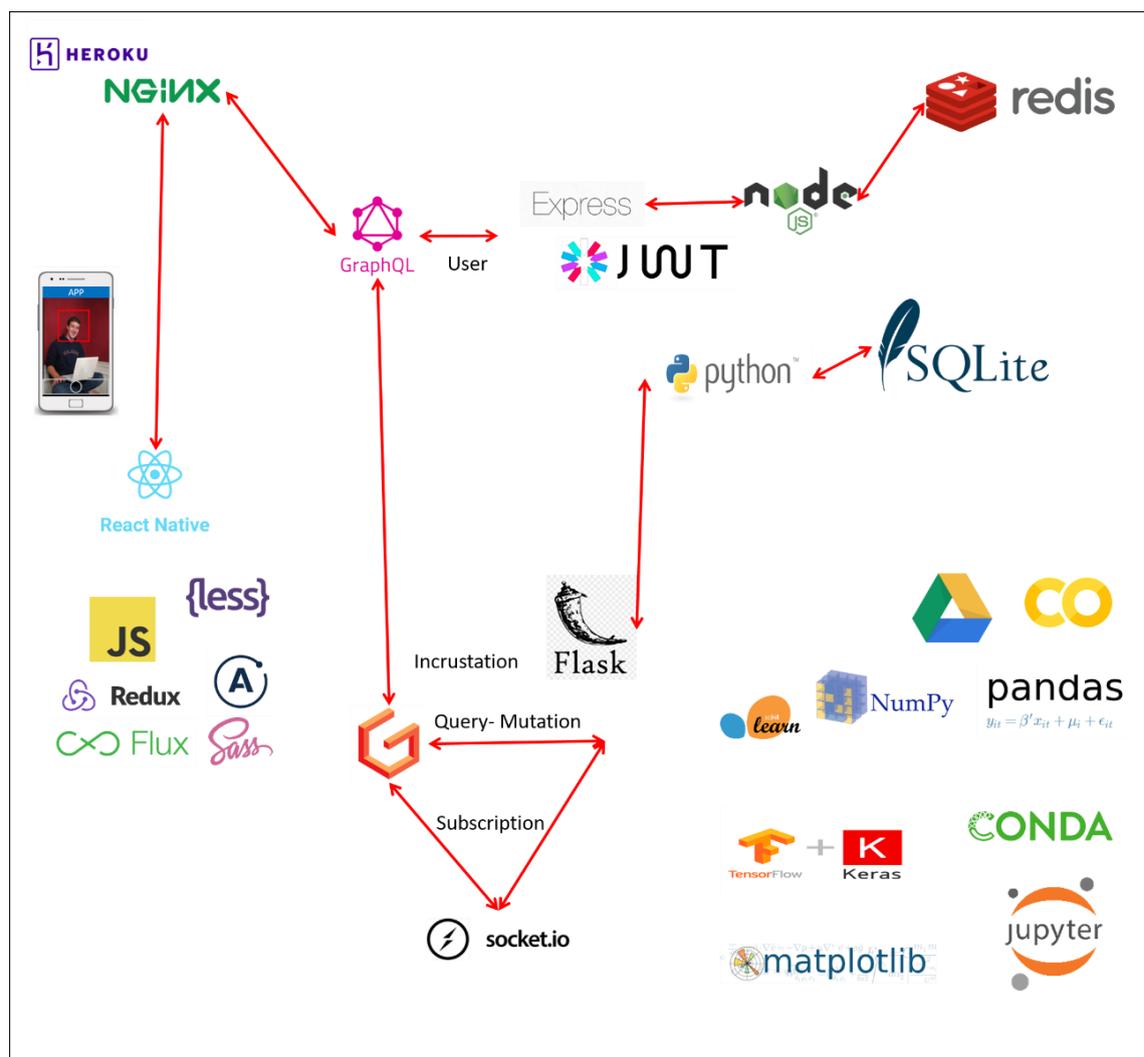
Herramientas de desarrollo	
Python	El lenguaje de programación utilizado.
Anaconda	Para realizar la instalación del entorno virtual utilizado durante el desarrollo del trabajo.
Pillow	Se utilizó para tratar y modificar imágenes.
Jupyter	Editor de código.
Stackoverflow	Se ha obtenido respuestas a problemas que se desconoce en el proceso de investigación.
Keras	Un marco de desarrollo para el desarrollo de código en base a inteligencia artificial
matplotlib	Se usa para mostrar gráficos en Jupyter
sklearn	Se usa para extraer una muestra aleatoria de dataset
Planificación y diseño	
Github	Para llevar el control de versiones.
Draw.io	Se utilizó para generar esquemas y diagramas.
Tensorboard	Se utilizó para visualizar la arquitectura y generar gráficos.

Anexo 16. Metodología de desarrollo de aplicación

La metodología que se utilizará para el desarrollo del software será la del modelo iterativo incremental, debido a que utiliza las ventajas del modelo en cascada y el modelo de desarrollo evolutivo, este modelo se adapta de forma más eficiente para el desarrollo de este software en comparación con otros modelos ya que está basado en la filosofía de desarrollo en incrementos, en la que cada uno (incremento) proporciona un subconjunto de funcionalidades requeridas por el cliente



Anexo 17. Arquitectura de aplicación



Para poder visualizar los datos del algoritmo propuesto, se desarrolló una aplicación que consta de backend y frontend. En el desarrollo del backend, se utilizó Python con Flask como framework para exponer una API. La aplicación y dicha API estarán compuestas por GraphQL, que manejará las consultas a la base de datos SQLite, donde se almacenarán los pesos generados por Python. Previamente, el algoritmo fue entrenado utilizando Conda con Jupyter, Keras, TensorFlow, NumPy y Pandas para tareas que requieren características computacionales no exigentes. Para la tarea de entrenamiento, se usó Colab, y para los gráficos, se utilizó Matplotlib.

Una vez entrenado nuestro modelo y expuesto a la API de GraphQL, en el frontend se tiene una aplicación React compuesta por Flux, Redux, LESS, SASS y, lo más importante, Apollo, que se utilizó para conectar con la API mediante un servidor con Nginx en su interior. Para las validaciones, se utilizó JSON Web Token corriendo en Node.js, que a su vez utiliza Redis para mejorarla velocidad de respuesta desde la memoria

Anexo 18. Preparación de datos

Se obtuvo el conjunto de datos de las imágenes ya ordenadas y procesadas en escala de grises. En el conjunto de datos existieron 10 imágenes por cada clase, de modo que se pasó a ordenar las imágenes para que fueran utilizadas desde una ruta raíz, evitando así tener que dividir los datos manualmente. El conjunto de datos pre procesado fue utilizado tanto con el algoritmo propuesto como también se usó con el algoritmo del triplete.

Entrenamiento de modelos

Se realizó un análisis con respecto a los modelos generados a partir de la implementación de los algoritmos con el objetivo de tener la certeza de que los datos registrados en nuestra ficha de recolección de datos provengan de una fuente que haya sido generada en igualdad de condiciones para cada algoritmo. Esto incluye la cantidad de épocas de entrenamiento, la arquitectura de red neuronal, la arquitectura de hardware, el conjunto de datos a entrenar y los mismos márgenes, para que los resultados obtenidos reflejen características exclusivas del impacto de la implementación de cada algoritmo.

En el presente análisis, las interpretaciones de los resultados obtenidos de los gráficos de entrenamiento de los modelos se llevaron a cabo siguiendo las directrices establecidas por Santosh et al. (2022). Este grupo de investigadores ha contribuido significativamente al campo de modelos de aprendizaje profundo, proporcionando una base teórica sólida y metodologías relevantes para la evaluación y comprensión de la dinámica de entrenamiento en modelos avanzados.

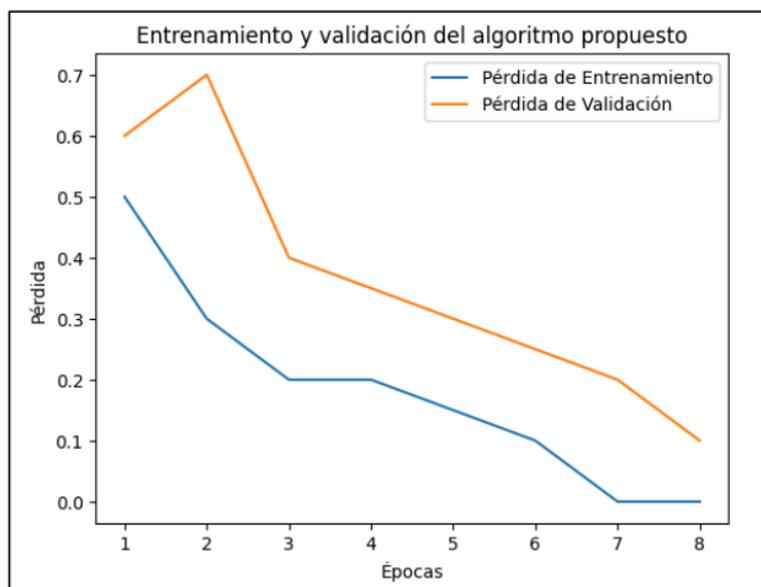
Por otro lado, el libro de Hastie et al. (2009) fue crucial en nuestro enfoque analítico, ya que su trabajo ofreció perspectivas valiosas sobre las mejores

prácticas en el diseño y evaluación de modelos de aprendizaje profundo. Se abordaron aspectos clave del proceso de entrenamiento, incluidas las estrategias de optimización, la gestión de la complejidad del modelo, la interpretación de las métricas de rendimiento, así como conceptos de regularización y sobreajuste.

Entrenamiento de algoritmo propuesto

Resultados obtenidos a partir del entrenamiento usando el algoritmo propuesto:

Figura 10. Gráfico de curvas de aprendizaje y pérdida de algoritmo propuesto



Fuente: Elaboración propia

Épocas 1-2 (Entrenamiento y Validación Iniciales):

La pérdida de entrenamiento disminuyó, indicando que el modelo está aprendiendo de manera efectiva del conjunto de entrenamiento.

La pérdida de validación también disminuyó, lo cual es positivo, haciendo que el modelo generalice bien.

Épocas 3-4 (Posible Sobreajuste):

La pérdida de entrenamiento continuó disminuyendo, pero la pérdida de validación se estancó o incluso aumentó.

Esto podría indicar un posible sobreajuste, donde el modelo tiene dificultades para generalizar.

Épocas 5-8 (Regularización y Mejora):

Se aplicaron técnicas de regularización para mejorar la generalización y evitar el sobreajuste.

La pérdida de validación volvió a disminuir, indicando que las técnicas de regularización resultaron ser efectivas.

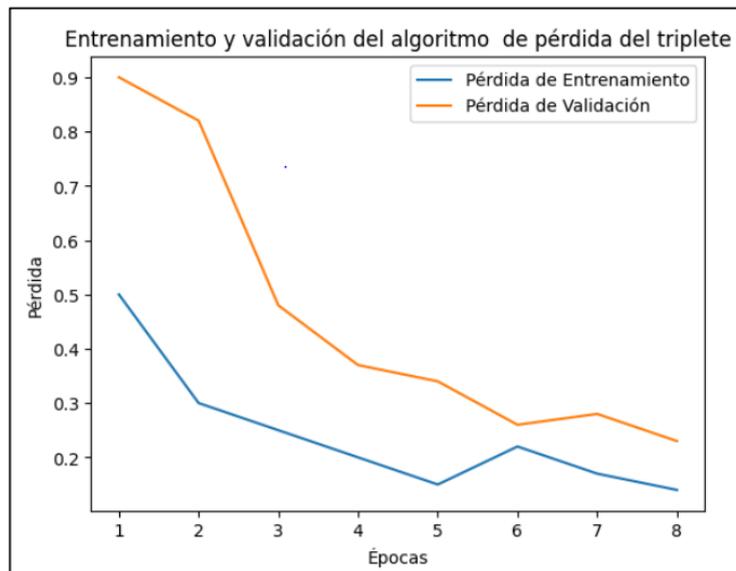
Consideración Adicional:

Aunque la pérdida de entrenamiento alcanzó valores cercanos a cero en las últimas épocas, esto podría ser indicativo de un ligero sobreajuste."

Entrenamiento de algoritmo de pérdida del triplete

Resultados obtenidos a partir del entrenamiento usando el algoritmo de pérdida del triplete.

Figura 11. Gráfico de curvas de aprendizaje y pérdida de algoritmo del triplete



Fuente: Elaboración propia

Épocas 1-2 (Entrenamiento y Validación Iniciales):

La pérdida de entrenamiento disminuyó, indicando que el modelo estuvo aprendiendo de manera efectiva del conjunto de entrenamiento.

La pérdida de validación también disminuyó, lo cual es positivo, sugiriendo que el modelo generalizó bien a los datos.

Épocas 3-4 (Regularización):

La pérdida de entrenamiento siguió disminuyendo, pero la pérdida de validación tuvo una disminución significativa, indicando un ligero sobreajuste.

Ambas pérdidas continuaron disminuyendo, lo que es positivo.

Épocas 5-8 (Regularización y Mejora):

Las pérdidas siguieron disminuyendo, pero la pérdida de validación comenzó a estabilizarse en la época 5.

Aunque la pérdida de entrenamiento aumentó ligeramente en la época 6, la pérdida de validación siguió disminuyendo.

Ambas pérdidas tuvieron pequeñas variaciones en la época 7, y la pérdida de validación aumentó ligeramente.

Ambas pérdidas disminuyeron nuevamente en la época 8, mostrando una mejora general.

Resultado de entrenamiento de modelo

Precisión:

Algoritmo	Imágenes	Épocas	Hardware	VP	FP	Precisión
Propuesto	400	8	NVIDIA Tesla T4	37	3	92.5%
Algoritmo de pérdida del triplete	400	8	NVIDIA Tesla T4	38	2	95%

Tiempo:

Algoritmo	Imágenes	Épocas	Hardware	Tiempo de reconocimiento facial(ms)
Propuesto	400	8	NVIDIA Tesla T4	0.96525
Algoritmo de pérdida del triplete	400	8	NVIDIA Tesla T4	0.7025

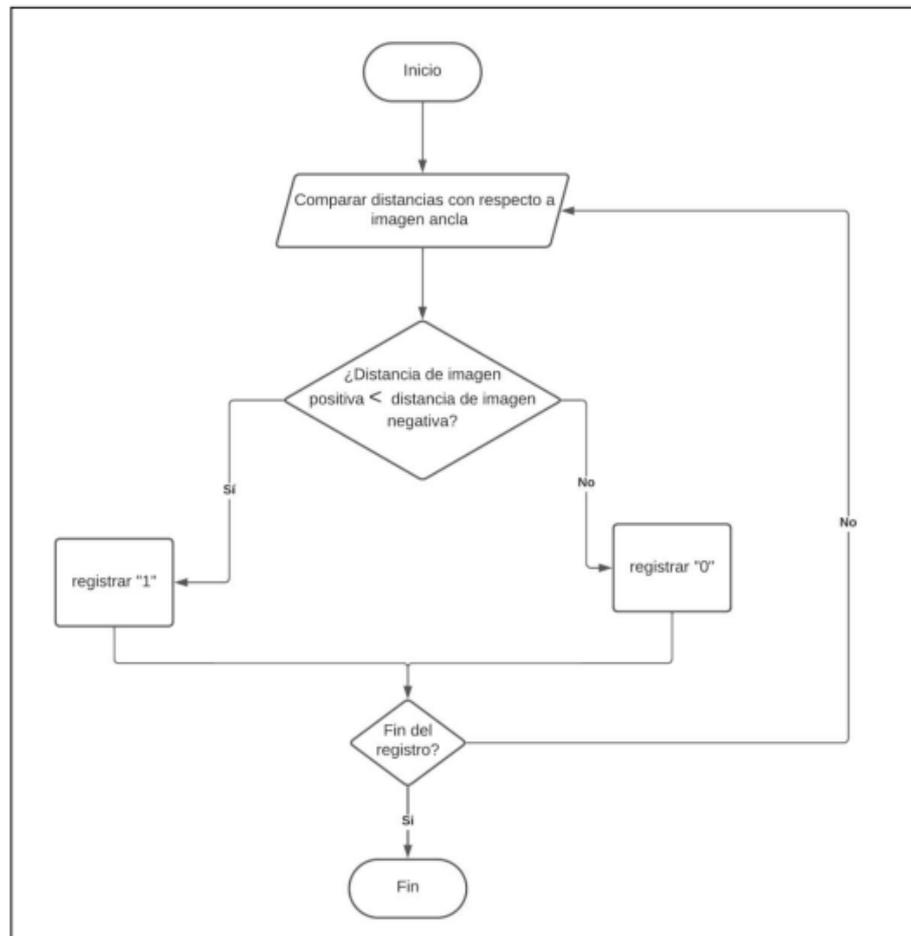
Anexo 19. Proceso de registro en instrumentos de recolección de datos

En lo que respecta a la recolección de información, se dispone de dos instrumentos en forma de ficha de recolección de datos para registrar los resultados del algoritmo propuesto y algoritmo del triplete en base al tiempo y precisión.

Para la primera ficha que, se incluyeron las siguientes columnas: La columna 'Clase' que identifica a cada representación facial en el cual se realizarán las pruebas. La columna 'Tiempo (Algoritmo del triplete)' hace referencia al transcurso del tiempo en milisegundos desde que comenzó el proceso de reconocimiento facial hasta el reconocimiento usando el algoritmo de pérdida del triplete. La columna 'Tiempo (Algoritmo propuesto)' hace referencia al transcurso del tiempo en milisegundos desde que comenzó el proceso de reconocimiento facial hasta el reconocimiento usando el algoritmo propuesto.

Para la segunda ficha se incluyeron las siguientes columnas: La columna 'Precisión Triplete' y 'Precisión algoritmo propuesto' en las cuales se registra un valor de '1' si la distancia de la imagen positiva es menor que la distancia de la imagen negativa, acercándose al valor del ancla que siempre será cero, ya que se usa como base. Las distancias tienden a ser más bajas cuanto más se parecen a la imagen ancla. Teniendo en cuenta que la imagen con la distancia más baja será la que el algoritmo reconozca como la imagen de identificación final. Por ejemplo, si se tiene una imagen del rostro de una persona 'X' y, al generarse la imagen positiva y negativa, en la imagen negativa se encuentra el rostro de una persona 'Y' con un valor de distancia más bajo que la imagen positiva, entonces el sistema estaría reconociendo, pero daría un falso positivo al indicar que la persona 'Y' es igual a la persona 'X'. Por lo tanto, se registra con valor '0', que indica un falso positivo en la precisión de la ficha. Por otro lado, si la distancia de una imagen negativa tiene una distancia mayor a la distancia que tiene el rostro de la imagen positiva, entonces el sistema estaría indicando que la imagen con la distancia más baja es el rostro de la imagen positiva, lo que indica que el rostro de la imagen positiva es igual al rostro 'X' de la imagen ancla, siendo correcto ya que se busca que la imagen positiva tenga una distancia menor. En este caso, se procede a registrar un valor de '1', que indicaría verdadero positivo en la columna de precisión de la ficha."

Diagrama de flujo del registro de precisión



Fuente: Elaboración propia

Según O'Mahony et al. (2019), se indica que, por un lado, se tiene una imagen ancla, una positiva y una imagen negativa, y que se tiene como finalidad encontrar un espacio de incrustación donde las clases o diferencias entre imágenes sean mínimas. Basado en ello, este diagrama de flujo ofrece una representación visual y sistemática del proceso de evaluación, permitiendo una comprensión clara de cómo se registran los datos para luego proceder a calcular la precisión en el reconocimiento facial.

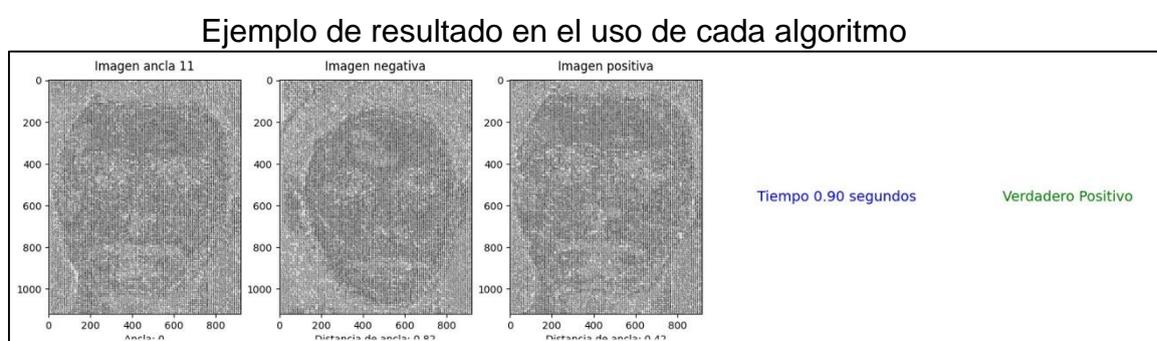
La captura de datos y la creación de una ficha de datos manualmente después de realizar el entrenamiento del modelo con un conjunto de datos como ORL (utilizado en el código proporcionado) involucraron los siguientes pasos:

Paso 1. Obtención del conjunto de datos: Se obtuvo el conjunto de datos también llamado ORL, que contenía imágenes faciales de diferentes personas, y se organizó el

conjunto de datos en una estructura adecuada, donde cada carpeta representaba a una persona y contenía múltiples imágenes de esa persona.

Paso 2. Entrenamiento del modelo: El modelo fue entrenado utilizando el conjunto de datos. Esto generó embeddings (vectores de características) para cada imagen.

Paso 3. Generación de resultados: Después del entrenamiento, se generaron tripletes aleatorios a partir de los embeddings. Cada triplete consistió en tres imágenes aleatorias para realizar las pruebas de reconocimiento, utilizando un ancla que representaba una imagen con la que se compararía la imagen positiva y negativa. También se incluyeron el tiempo, la clase y si era un verdadero positivo o un falso positivo.



Fuente: Elaboración propia

Paso 4. Captura de datos manual: Los resultados obtenidos (verdaderos positivos, falsos positivos, tiempo y clase) fueron registrados en una ficha de datos de manera manual, considerando que al final del registro se realizarían los cálculos para obtener la precisión y el tiempo final. Se siguió el ejemplo de Suryawanshi et al. (2023), quienes registraron los datos de imágenes reconocidas correctamente (verdaderos positivos) y también los datos de imágenes reconocidas incorrectamente (falsos positivos). Paso 5. Repetición para un siguiente registro: Se repitió el proceso de generación de tripletas y captura de datos manual para los siguientes resultados.

Los pasos mencionados se enfocaron en recopilar información relevante sobre el impacto de la implementación de un algoritmo basado en una red siamesa, la pérdida del triplete, el uso de k-vecinos más cercanos y el aprendizaje de una sesión para el reconocimiento de imágenes faciales.

El método de análisis de datos utilizado fue de carácter cuantitativo; de esta forma, se pudo analizar datos numéricamente. Se utilizó la recolección de datos para probar la

hipótesis mediante los siguientes indicadores: 'Tiempo de reconocimiento facial' y 'Precisión de reconocimiento facial'. Posteriormente, se compararon los resultados obtenidos del algoritmo propuesto con el resultado del algoritmo del triplete. Para esta comparación, tanto el algoritmo del triplete como el algoritmo propuesto se sometieron a las pruebas en igualdad de condiciones, tanto en la arquitectura de red neuronal, hardware y épocas de entrenamiento, de modo que los resultados puedan reflejar diferencias únicamente entre los algoritmos y no arrojar datos inexactos por agentes externos.

Se utilizaron las siguientes fórmulas para los indicadores:

$$Tiempo = \frac{\sum x}{n}$$

Dónde:

$\sum x$ = *Es la sumatoria de todos los registros de tiempo.*

n = *El número total de registros*

(Ali et al., 2023, p.15; Haeussler et al., 2003, p.1)

$$Precisión = \frac{VP}{(VP + FP)} \times 100$$

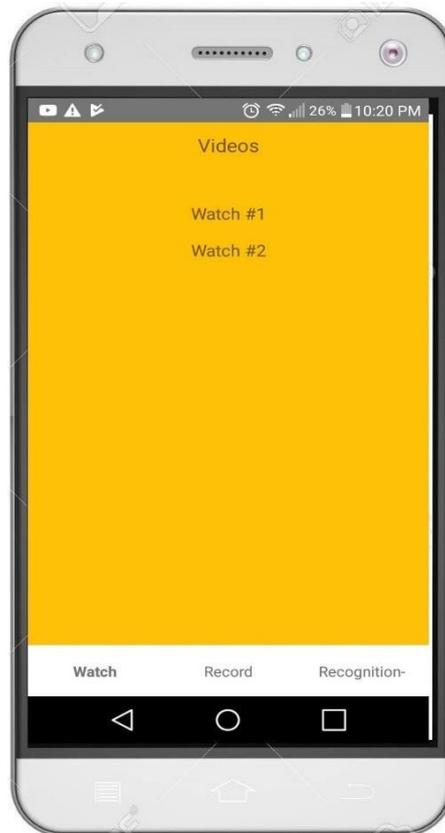
Dónde:

VP = Verdadero positivo

FP = Falso positivo

(Wijaya et al., 2023, p.87; Ali et al., 2023, p.15)

Anexo 20 . Pantalla de inicio de aplicación



En la presente imagen se pueden visualizar las opciones que se muestran al inicio de la aplicación, como la opción de visualizar todos los resultados obtenidos, así como la opción para grabar un nuevo rostro y otra opción para comenzar a reconocer a partir de los frames en tiempo real.

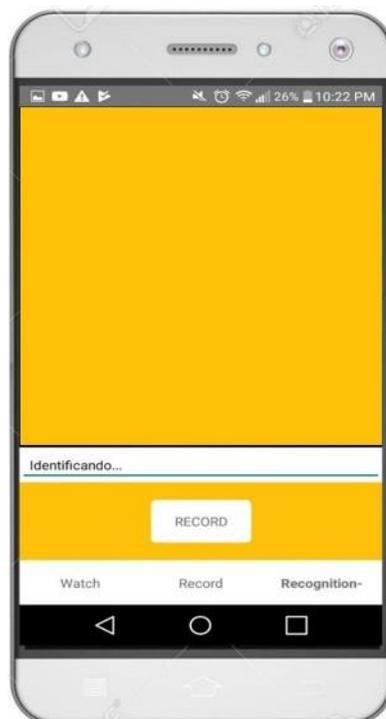
Anexo 21. Pantalla de registro de nuevo rostro

En la siguiente imagen se puede visualizar un ejemplo del registro de rostro.



Anexo 22. Pantalla de identificación

En la imagen que se visualiza se puede apreciar el procesamiento de reconocimiento



Anexo 23. Algoritmo del triplete

Input: Tensor
Output: Tensor

Algoritmo *perdida_del_triplete*

$$ancla = \{a_1, a_2, a_3, a_4, a_5, \dots, a_m\}$$

$$positivo = \{p_1, p_2, p_3, p_4, p_5, \dots, p_z\}$$

$$negativo = \{ne_1, ne_2, ne_3, ne_4, ne_5, \dots, ne_s\}$$

$$margen \leftarrow 0.2$$

$$disPos = \sum_{i=1}^n (ancla_i - positivo_i)^2$$

$$disNeg = \sum_{i=1}^n (ancla_i - negativo_i)^2$$

$$perdida = \begin{cases} disPos - disNeg + margen; & \text{si } disPos - disNeg + margen \geq 0 \\ 0 & ; \text{si } disPos - disNeg + margen < 0 \end{cases}$$

$$media = \frac{1}{N} \sum_{i=1}^N perdida_i$$

$$retornar \leftarrow media$$

FinAlgoritmo

El algoritmo tiene como finalidad agrupar los embeddings de las imágenes. El algoritmo presentado es una representación del algoritmo del triplete expuesto en la investigación de (Hastie et al., 2009, p. 14; Hernández-Díaz et al., 2023; O'Mahony et al., 2019; Kumar et al., 2023).

Anexo 24. Algoritmo propuesto

Input: Tensor

Output: Tensor

Funcion $tensorResultado \leftarrow ordenVector(tensor)$

$tensor \in \mathbb{R}^{p,q,r}$

$t_p \leftarrow tensor_p$

$t_q \leftarrow tensor_q$

$t_r \leftarrow tensor_r$

$valor_minimo \leftarrow [1,1,1]$

$pos_minimo[1,1,1]$

Para $i \leftarrow 1$ **Hasta** $t_p - 1$ **Hacer**

Para $j \leftarrow 1$ **Hasta** $t_q - 1$ **Hacer**

Para $k \leftarrow 1$ **Hasta** $t_r - 1$ **Hacer**

Si $tensor[i, j, k] < valor_minimo$ **Entonces**

$valor_minimo \leftarrow tensor[i, j, k]$

$pos_minimo \leftarrow [i, j, k]$

FinSi

FinPara

FinPara

FinPara

$tensorResultado \leftarrow [valor_minimo, pos_minimo]$

FinFuncion

Funcion $k \leftarrow calcularK(tensor)$

$k \leftarrow \sum_{i=1}^p \sum_{j=1}^q \sum_{k=1}^r ordenVector(tensor_{ijk})$

FinFuncion

Algoritmo propuesto

$ancla = \{a_1, a_2, a_3, a_4, a_5, \dots, a_m\}$

$positivo = \{p_1, p_2, p_3, p_4, p_5, \dots, p_z\}$

$kVecinosNeg = \{kn_1, kn_2, kn_3, kn_4, kn_5, \dots, kn_d\}$

$kVecinosPos = \{kp_1, kp_2, kp_3, kp_4, kp_5, \dots, kp_g\}$

$margen \leftarrow 0.2$

$disPos = \sum_{v=1}^n (ancla_v - positivo_v)^2$

$disNeg = \sum_{h=1}^n (ancla_h - positivo_h)^2$

$posKnnIndices \leftarrow calcularK(disPos)$

$negKnnIndices \leftarrow calcularK(disNeg)$

$prPosDis = \frac{1}{N} \sum_{i=1}^N positivo_{posKnnIndices}$

$prNegDis = \frac{1}{N} \sum_{i=1}^N negativo_{negKnnIndices}$

$$perdida = \begin{cases} prPosDis - prNegDis + margen; & \text{si } prPosDis - prNegDis + margen \geq 0 \\ 0; & \text{si } prPosDis - prNegDis + margen < 0 \end{cases}$$

$$media = \frac{1}{N} \sum_{i=1}^N perdida_i$$

retornar \leftarrow *media*

FinAlgoritmo

El algoritmo presentado tiene como base el algoritmo expuesto por (Hastie et al., 2009, p. 14).