UNIVERSIDAD TÉCNICA DE MACHALA

FACULTAD DE INGENIERÍA CIVIL

MAESTRÍA EN SOFTWARE

MODELO DE RECONOCIMIENTO DE ROSTROS UTILIZANDO INTELIGENCIA ARTIFICIAL.

CASO DE ESTUDIO: SUPERVISIÓN REMOTA DE EXÁMENES EN LÍNEA

ING. CARLOS MANUEL QUEZADA CENTENO

TUTOR: ING. WILMER RIVAS ASANZA

MACHALA - ECUADOR

2023

PENSAMIENTO

"Siempre parece imposible hasta que se hace"

Nelson Mandela

DEDICATORIA

A mi amada esposa Marcia Maribel, quien es mi compañera de vida y mi mayor apoyo en mi carrera profesional. Su amor incondicional, paciencia, apoyo y su motivación han sido la luz que me ha guiado en este camino de crecimiento y superación. Como dijo Alan Kay, "la mejor manera de predecir el futuro es inventarlo", y gracias a ti, he tenido la confianza para perseguir mis sueños y alcanzar mis metas, te amo.

A mi Madre, Gloria Centeno, por inculcarme y formarme en el camino de la rectitud, sé que estarías orgullosa de mí, todo lo que soy, soy gracias a ti, eres mi estrella en el cielo que me guía en este mundo de incertidumbres y retos

A mi hijo Carlos Alejandro, mi superación es un tributo a ti, para que algún día veas que, con trabajo duro y perseverancia, los sueños se pueden hacer realidad. Gracias por enseñarme lo que significa la "felicidad +1", porque mi responsabilidad contigo me motiva a ser mejor cada día. Espero que este pequeño peldaño profesional te inspire a ti también a perseguir tus sueños y sobre todo superarte cada día.

Carlos Q.

AGRADECIMIENTOS

A los docentes de la Universidad Técnica de Machala que durante mi estancia en pregrado y ahora en posgrado han compartido sus conocimientos de forma íntegra y profesional. De manera especial a la Mg. Jennifer Célleri Pacheco, por invitarme a participar de esta Maestría y por ser la líder y pionera para su desarrollo en nuestra provincia.

Al Dr. Wilmer Rivas, tutor de esta investigación por acompañarme en el proceso y brindarme su experiencia en este campo tecnológico. A mis compañeros de clase por compartir sus experiencias profesionales de forma desinteresada y honesta.

RESPONSABILIDAD DE AUTORÍA

Yo, Carlos Manuel Quezada Centeno con CI. **0704195635**, declaro que el trabajo: "**MODELO DE RECONOCIMIENTO DE ROSTROS UTILIZANDO INTELIGENCIA ARTIFICIAL. CASO DE ESTUDIO: SUPERVISIÓN REMOTA DE EXÁMENES EN LÍNEA**", en opción al título de Magister en Software, es original y auténtico; cuyo contenido: conceptos, definiciones, datos empíricos, criterios, comentarios y resultados son de mi exclusiva responsabilidad.

Carlos Manuel Quezada Centeno

Firmado digitalmente por Carlos Manuel Quezada Centeno
Fecha: 2023.06.05 00:49:40 -05'00'

CARLOS MANUEL QUEZADA CENTENO

C.I. **0704195635**

Machala, 2023/05/04

REPORTE DE SIMILITUD TURNITIN

Q28042023

INFORME DE ORIGINALIDAD

5%	4%	2%	1%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Universidad de Santiago de Chile Trabajo del estudiante	<1%
2	LUCERO VERONICA LOZANO VAZQUEZ, ALBERTO JORGE ROSALES SILVA, EDUARDO RAMOS DIAZ, JEAN MARIE VIANNEY KENANI. "MODELO DE MÉTODOS COMBINADOS CON LÓGICA DIFUSA PARA UN SISTEMA DE RECONOCIMIENTO FACIAL", DYNA NEW TECHNOLOGIES, 2018 Publicación	<1%
3	repobib.ubiobio.cl Fuente de Internet	<1%
4	Submitted to Universidad Pontificia Bolivariana Trabajo del estudiante	<1%
5	dialnet.unirioja.es Fuente de Internet	<1%

CERTIFICACIÓN DEL TUTOR

Por medio de la presente apruebo que el trabajo de titulación, titulado **"MODELO DE RECONOCIMIENTO DE ROSTROS UTILIZANDO INTELIGENCIA ARTIFICIAL"**, del autor **Quezada Centeno Carlos Manuel**, en opción al título de Master en Software, sea presentado al Acto de Defensa.

Firmado electrónicanente por:
WILMER BRAULIO
RIVAS ASANZA

Ing. Wilmer Rivas Asanza, Phd.

C. I. 0702580192

CESIÓN DE DERECHOS DE AUTOR

Yo, CARLOS MANUEL QUEZADA CENTENO con CI. 0704195635, autor del trabajo de titulación ***"MODELO DE RECONOCIMIENTO DE ROSTROS UTILIZANDO INTELIGENCIA ARTIFICIAL. CASO DE ESTUDIO: SUPERVISIÓN REMOTA DE EXÁMENES EN LÍNEA"***, en opción al título de Magister en Software, declaro bajo juramento que:

- El trabajo aquí descrito es de mi autoría, que no ha sido presentado previamente para ningún grado o calificación profesional. En consecuencia, asumo la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

- Cede a la Universidad Técnica de Machala de forma exclusiva con referencia a la obra en formato digital los derechos de:

  a. Incorporar la mencionada obra en el repositorio institucional para su demostración a nivel mundial, respetando lo establecido por la Licencia *Creative Commons Atribution-NoCommercial* – Compartir Igual 4.0 Internacional (CC BY NCSA 4.0); la Ley de Propiedad Intelectual del Estado Ecuatoriano y el Reglamento Institucional.

  b. Adecuarla a cualquier formato o tecnología de uso en INTERNET, así como correspondiéndome como Autor la responsabilidad de velar por dichas adaptaciones con la finalidad de que no se desnaturalice el contenido o sentido de la misma.

Carlos Manuel Quezada Centeno

Firmado digitalmente por Carlos Manuel Quezada Centeno
Fecha: 2023.06.05 00:50:04 -05'00'

CARLOS MANUEL QUEZADA CENTENO

C.I. 0704195635

Machala, 2023/05/04

RESUMEN

En el contexto de la pandemia de COVID-19, la virtualización de la educación y las evaluaciones en línea se volvieron esenciales, para garantizar la continuidad de los procesos educativos. Sin embargo, la suplantación de identidad en entornos remotos, planteó desafíos en cuanto a la validez y confiabilidad en el proceso de toma de evaluaciones. Esta investigación, propone un modelo de reconocimiento facial basado en inteligencia artificial para mejorar la seguridad en las evaluaciones en línea. Se comparan tres algoritmos de reconocimiento facial: *Eigenfaces, Face-recognition y FaceNet*, este último un modelo de Redes Neuronales Convolucionales. Se realizaron experimentos utilizando 5000 rostros que conforman la muestra del *dataset*, bajo diversas circunstancias de iluminación y poses, para evaluar la precisión y eficiencia de cada algoritmo. Los resultados indicaron que el algoritmo *FaceNet* presentó altas métricas de eficiencia en la identificación de rostros y es capaz de identificar a personas de manera precisa, superando a los otros dos algoritmos comparados, demostrando ser el adecuado en la toma de evaluaciones en línea para evitar la suplantación de identidad. El prototipo de reconocimiento facial se desarrolló utilizando el lenguaje de programación Python, la librería *OpenCV*, *Dlib, Tensorflow y el framework CaffeModel. Este* trabajo de titulación contribuye al desarrollo de soluciones tecnológicas para la mejora de la seguridad en la toma de evaluaciones en línea. Además, abre la posibilidad de futuras investigaciones para su implementación en otros contextos, como la identificación de personas en sistemas de seguridad y vigilancia en tiempo real.

PALABRAS CLAVE: Reconocimiento facial, Inteligencia artificial, Suplantación de identidad, Evaluación en línea, Seguridad, Prototipo.

ABSTRACT

In the context of the COVID-19 pandemic, the virtualization of education and online assessments became essential to ensure the continuity of educational processes. However, impersonation in remote environments posed challenges in terms of validity and reliability in the process of taking assessments. This research proposes a facial recognition model based on artificial intelligence to improve security in online assessments. Three face recognition algorithms are compared: Eigenfaces, Face-recognition and FaceNet, the latter a Convolutional Neural Networks model. Experiments were performed using 5000 faces that make up the dataset sample, under different lighting and pose circumstances, to evaluate the accuracy and

efficiency of each algorithm. The results indicated that the FaceNet algorithm presented high efficiency metrics in face identification and is able to identify people accurately, outperforming the other two compared algorithms, proving to be the appropriate one in taking online evaluations to avoid impersonation. The facial recognition prototype was developed using the Python programming language, the OpenCV library, Dlib, Tensorflow and the CaffeModel framework. This degree work contributes to the development of technological solutions for the improvement of security in taking online evaluations. In addition, it opens the possibility of future research for implementation in other contexts, such as the identification of people in security systems and real-time surveillance.

KEYWORDS: Facial recognition, Artificial intelligence, Identity fraud, Online assessment, Security, Prototype.

ÍNDICE GENERAL

PENSAMIENTO ii
DEDICATORIA iii
AGRADECIMIENTOS iv
RESPONSABILIDAD DE AUTORÍA v
REPORTE DE SIMILITUD TURNITIN vi
CERTIFICACIÓN DEL TUTOR vii
CESIÓN DE DERECHOS DE AUTOR viii
RESUMEN ix
ABSTRACT ix
ÍNDICE GENERAL xi
ÍNDICE DE FIGURAS xiv
ÍNDICE DE TABLAS xvi
INTRODUCCIÓN 1
CAPÍTULO I 6
1. MARCO TEÓRICO 6
1.1 ANTECEDENTES HISTÓRICOS 6
1.2 Antecedentes Conceptuales y referenciales 8
1.2.1 Hipótesis de la investigación. 8
1.3 Fundamentación teórica de la variable dependiente. 8
1.4 Antecedentes Contextuales 8
1.5 TRABAJOS PREVIOS 9
1.6 Fundamentación teórica de la variable independiente. 13
1.6.1 Biometría 13
1.6.2 Inteligencia artificial 13
1.6.3 Aprendizaje de máquina vs Aprendizaje profundo 15
1.6.4 Visión Computacional 16
1.6.5 Aprendizaje automático (*Machine Learning*) 16
1.6.6 Reconocimiento facial 19
1.6.7 Detección de rostros 19
1.6.8 Métodos, algoritmos y bibliotecas 20
1.6.8.1 Detección usando descriptores Haar 21
1.6.8.2 Detección usando *deep learning* 25
1.6.8.3 Detección usando el método de los HOG 29
1.6.8.4 Random Forest 38
1.6.8.5 K-Nearest Neighbours 39
1.6.8.6 Support Vector Machines 40

1.6.8.7 Redes Neuronales 40
1.6.8.8 Deep Learning 41
1.6.8.9 Biblioteca Face-recognition 42
1.6.8.10 Biblioteca *Dlib* 43
1.6.8.11 Biblioteca FaceNet 43
1.6.8.12 Framework CaffeModel 44
1.7 Metodología CRISP DM 44
CAPÍTULO II 46
2. Materiales y métodos 46
2.1. Diseño de la investigación 46
2.1.1 Diseño experimental 46
2.3 Paradigma o Enfoque 46
2.3.1 Enfoque cuantitativo 47
2.4. Cálculo de población y muestra 47
2.5 Técnicas estadísticas 48
2.6 Métodos teóricos 49
2.6.1 Método Inductivo 49
2.7 Metodología CRISP-DM 49
2.7.1 Fase de comprensión del negocio 50
2.7.1.2 Situación Actual 50
2.7.1.3 Objetivos 50
2.7.1.4 *Stakeholders* 50
2.7.1.5 Alcance 51
2.7.1.6 Requerimientos Funcionales 52
2.7.1.7 Criterios para los experimentos 57
2.7.2 Fase de comprensión de los datos 58
2.7.2.1 Desarrollo de los experimentos 58
2.7.2.1.1 Detección usando el Algoritmo: *HaarCascade* 58
2.7.2.1.2 Detección de rostros usando el algoritmo: Dlib 60
2.7.2.1.3 Detección usando el algoritmo Facenet 62
2.7.3 Fase de preparación de datos 63
2.7.3.1 Extracción de rostros algoritmo haarCascade 64
2.7.3.2 Extracción de rostros framework CaffeModel 65
2.7.3.3 Resultados en la etapa de extracción de rostros 66
2.7.4 Fase de modelado 66
2.7.4.1 Entrenamiento del modelo mediante la técnica de Eigenface 66
2.7.4.2 Entrenando el modelo usando Face_recognition 66
2.7.3.3 Entrenando el modelo usando FaceNet 67

2.7.5 Evaluación 68
2.8 Métodos empíricos 68
CAPÍTULO 3 70
3.1 RESULTADOS 70
3.1.1 Fase de evaluación 70
3.1.1.1 Métricas 70
3.1.4.2 Índice o coeficiente de Kappa 70
3.1.4.3 Resultados Obtenidos Experimento 1 - *HaarCascade - Eigenface* 72
3.1.4.4 Resultados Obtenidos Experimento 2 - *Dlib, FaceRecognition* y Knn 73
3.1.4.5 Resultados Obtenidos Experimento 3 - FaceNet - Tensorflow 75
3.1.4.6 Resultados generales de los experimentos 76
3.1.4.7 Resultados en base a la Matriz de confusión 77
Contrastación de Hipótesis 79
3.1.5 Fase de Despliegue 81
3.1.5.1 Desarrollo del prototipo 81
CAPÍTULO 4 82
4. Discusión de resultados 82
4.1 Hallazgos significativos 82
4.2 Ventajas y desventajas de cada algoritmo 82
4.3 Resultados de la experimentación 83
4.3.1 Precisión 83
4.3.2 Sensibilidad (Recall) 85
4.3.3 Índices *Kappa* 85
4.4 Limitaciones 86
4.5 Trabajos futuros 87
4.6 Conclusiones 88
4.7 Recomendaciones 89
Bibliografía 91
Anexos 105
Anexo 1 105

ÍNDICE DE FIGURAS

Figura 1. Ramas de la IA. 14
Figura 2. Ingresos globales de la I.A de 2016 a 2025 (en mill. Euros) 15
Figura 3. Disciplinas de la visión computacional 16
Figura 4. Ciclo de vida del Machine Learning 17
Figura 5. Algoritmos de aprendizaje supervisado 18
Figura 6. Algoritmos de aprendizaje no supervisado 18
Figura 7. Etapas del reconocimiento facial 19
Figura 8. Detección de un rostro 20
Figura 9. Características o kernels convolucionales 22
Figura 10. Ejemplo de la aplicación de los kernels convolucionales a través de la imagen y su aplicación en múltiples escalas 22
Figura 11. Cálculo de una región S dentro de la Imagen Integral, para lo cual son necesarios los puntos A, B, C y D (Ecuación 2). 23
Figura 12. Ejemplo de clasificador en cascada. Las detecciones falsas(F) no pasan al siguiente clasificador (Cn). 24
Figura 13. Características Haar 24
Figura 14. código básico en Python para la detección de un rostro usando HaarCascade – OpenCv 25
Figura 15. Resultado del Algoritmo 25
Figura 16. Ejemplo de convolución. 26
Figura 17. Average Pooling con filtro 2x2 y stride = 2 27
Figura 18. Max Pooling con filtro 2x2 y stride = 2 27
Figura 19. Fully Conected Layer 28
Figura 20. Diagrama de una red neuronal convolucional. 28
Figura 21. Diagrama de una red neuronal convolucional identificando un animal 29
Figura 22. Localización de rostros 29
Figura 23. Aplicación de los descriptores HOG. Izquierda: Imagen bajo test dividida en varias celdas. Derecha: Visualización de los descriptores HOG sobre la imagen. 31
Figura 24. Operación de alineación y recorte de un rostro utilizando como referencia la línea de los ojos 32
Figura 25. Clasificación de Técnicas de Extracción de Características 33
Figura 26. Ejemplos de vecindad de pixeles en LBP, se muestra el número de pixeles (P) y el radio utilizado (R) Fuente: Tomada de [89]. 34
Figura 27. Ejemplo de LBP Multi-escala, en la izquierda se muestra el LBP tradicional, en la derecha la variación para bloques LBP Multi-escala 35
Figura 28. Eigenfaces de un conjunto de imágenes de la base Extended Yale Face Database B 35
Figura 29. Eigenfaces de un conjunto de imágenes de la base Extended Yale Face Database B. 36
Figura 30. Imágenes de entrenamiento 36
Figura 31. Conjunto de Eigenfaces 37
Figura 32. Diagrama Random Forest 39
Figura 33. Para K=3, la muestra se clasifica como clase B, pero para K=5 como clase A .. 40
Figura 34. Red neuronal con una capa oculta 41
Figura 35. Red neuronal con varias capas ocultas. 42
Figura 36. Metodología CRISP-DM. 45

Figura 39. Fases metodología CRISP-DM 50
Figura 40. Código fuente detección rostro haarCascade..... 59
Figura 41. Resultado de la detección de rostro usando EigenFace..... 59
Figura 42. Código en fuente para detectar rostros usando HaarCascade 60
Figura 43. Resultado del algoritmo HaarCascade 60
Figura 44. Resultado del algoritmo HaarCascade 60
Figura 45. Código en python para detectar rostros usando Framework: Caffemodel 61
Figura 46. Resultado de la detección de rostro usando Código Framework: Caffemodel.... 62
Figura 47. Código en python para detectar y guardar rostros usando TensorFlow..... 63
Figura 48. Entrenamiento de imágenes capturadas 63
Figura 49. Imágenes extraídas de una secuencia de video..... 64
Figura 50. Detección de rostros en diversas posiciones..... 64
Figura 51. Rostros extraídos. 65
Figura 52. Rostros extraídos del framework CaffeModel. 66
Figura 53. Resultado del entrenamiento 67
Figura 54. Código fuente entrenamiento del modelo 67
Figura 55. Modelos obtenidos 68
Figura 56. Resultados en base a la Matriz de confusión precisión 77
Figura 57. Resultados en base a la Matriz de confusión sensibilidad (Recall)..... 78
Figura 66. Métricas 83
Figura 67. Precisión 84
Figura 68. Resultados generales de precisión en base a la Matriz de confusión precisión.. 84
Figura 69. Resultados de la sensibilidad (Recall)..... 85
Figura 70. Índices general Kappa 86
Figura 58. Perfil del estudiante..... 105
Figura 59. Captura del rostro del estudiante. 106
Figura 60. Captura y entrenamiento del modelo..... 106
Figura 61. Ingreso a la evaluación 107
Figura 62. Entorno del examen en línea 107
Figura 63. Panel administrativo - Calificación, tiempos, revisión 108
Figura 64. Panel Administrativo eventos y resultados 109
Figura 65. Panel Administrativo eventos y resultados 109

ÍNDICE DE TABLAS

Tabla 1. Análisis de trabajos existentes actualmente. 12
Tabla 2. Matriz de stakeholder 51
Tabla 3. Requerimientos funcionales 53
Tabla 4. REQ001 Registro de usuario 54
Tabla 5. REQ002 Verificación de identidad 54
Tabla 6. REQ003 Captura de imágenes 55
Tabla 7. REQ004 Análisis de imágenes 55
Tabla 8. REQ005 Interfaz de usuario 55
Tabla 9. REQ006 Notificaciones 56
Tabla 10. REQ007 Reportes 56
Tabla 11. REQ008 Seguridad de la información. 56
Tabla 12. Algoritmos seleccionados. 57
Tabla 13. Matriz de confusión 70
Tabla 14. Valoración de concordancia del coeficiente Kappa 71
Tabla 15. Experimento 1 - HaarCascade – Eigenface 72
Tabla 16. Resultados obtenidos en las métricas de evaluación experimento 1 72
Tabla 17. Experimento 2 - Dlib, FaceRecognition y Knn 73
Tabla 18. Resultados obtenidos en las métricas de evaluación experimento 2 74
Tabla 19. Experimento 3 - FaceNet – Tensorflow 75
Tabla 20. Resultados obtenidos en las métricas de evaluación experimento 3 75
Tabla 21. Resultados generales de los experimentos 76
Tabla 22. Resultados en base a la Matriz de confusión precisión 77
Tabla 23. Resultados en base a la Matriz de confusión sensibilidad (Recall) 78
Tabla 24. Ventajas y desventajas de cada algoritmo 83

INTRODUCCIÓN

En Ecuador y en algunos países latinoamericanos, las autoridades gubernamentales decidieron suspender las clases presenciales en todas las instituciones educativas debido al brote del virus SARS-CoV-2, también conocido como COVID-19, que apareció en 2020. La Organización Mundial de la Salud (OMS) declaró la enfermedad como pandemia el 11 de marzo de 2020.

Los países decidieron implementar medidas de bioseguridad para reducir la propagación del virus en la población. En el caso del Gobierno ecuatoriano, algunas de las acciones tomadas incluyeron la declaración del estado de emergencia por parte del Ministerio de Salud Pública el 12 de marzo de 2020, y medidas como el aislamiento, las medidas adoptadas incluyeron la obligatoriedad del confinamiento en el hogar, el cierre de fronteras y aeropuertos, la prohibición de eventos masivos y la suspensión de clases [2]. Con esto se dio apertura al despliegue de modalidades de aprendizaje virtual [3] y con ello, la escuela tradicional tuvo que cambiar y hacer uso de las *TICs* [4], trayendo consigo la necesidad de que los docentes se adapten a nuevas modalidades de enseñanza e implementaciones tecnológicas [5], a través del uso de plataformas virtuales, que permitan de una forma ágil y confiable virtualizar el entorno del aula física en un ambiente virtual desde los hogares. De repente, el modelo educativo al que la comunidad educativa estaba acostumbrada cambió completamente en cuestión de días.

De acuerdo con la información proporcionada por la UNESCO [6], para el 31 de marzo de 2020, la pandemia había afectado a más de 1.500 millones de estudiantes, lo que representa el 89,4% de la población estudiantil mundial, en un total de 185 países que habían cerrado sus instalaciones educativas, incluyendo escuelas y universidades [6].

Los cambios de paradigmas educativos provocado por la pandemia, exigieron que las instituciones educativas, reconstruyan sus modelos pedagógicos tradicionales de enseñanza aprendizaje[5], implicando transformaciones significativas; las cuales tuvieron que adaptarse sobre la marcha, ya que, se requerían que estos sean implementadas de forma inmediata y muchos procesos involucrados en el ámbito educativo debían evolucionar y continuar a través de medios virtuales; tal es el caso de la evaluación de conocimientos, la cual según [7], permite al docente y por ende a la institución, medir el nivel educativo con respecto al conocimiento adquirido por el estudiante en clases. Los modelos educativos actuales consideran que la evaluación del conocimiento adquiere un nuevo sentido. Ya no se trata simplemente de recopilar datos, sino que se ha convertido en una pieza clave e imprescindible para que el

profesor pueda brindar al alumno la ayuda necesaria y valorar las transformaciones en el conocimiento que hayan ocurrido durante el proceso de enseñanza y aprendizaje.[7]

El uso de plataformas virtuales (*Zoom, Skype, Google Meets, Teams*), sin duda dieron una solución eficiente e inmediata para transformar el aula tradicional a virtual, así mismo, fueron virtualizados algunos procesos inmersos en la enseñanza aprendizaje, como por ejemplo: las evaluaciones de conocimientos, las cuales se efectuaron en la mayoría de los casos, a través del uso de formularios en línea y el uso de plataformas *LMS* como: *MOODLE, Chamilo, Sakai, Edmodo, Google Classroom, Almagesto, Teams*, entre otras; pero más sin embargo, a través de su uso cotidiano, se fueron encontrando necesidades que quizás antes no fueron tan notorias o requeridas y no fueron determinadas como urgentes.

En estos escenarios virtuales, es necesario mantener el control y la seguridad para preservar la integridad y credibilidad de los procesos educativos, tal es el caso de: las aprobaciones y certificaciones, en donde se requiere medir el conocimiento a través de una evaluación, pero, al no existir el contacto físico con el evaluado, ante la necesidad de garantizar la integridad del seguimiento académico, resulta crucial conocer y autenticar la verdadera identidad de la persona que se somete a la evaluación, así como controlar el entorno físico en el que se realiza dicha evaluación. El incumplimiento de estos controles podría aumentar el riesgo de suplantación de identidad y, en consecuencia, conducir a actos deshonestos en el ámbito académico.

Es precisamente esto, uno de los grandes obstáculos a los que se enfrenta este tipo de modalidades de enseñanza virtual [8][9], ya que, es algo a lo que las instituciones educativas de modalidad presencial no se han enfrentado desde su concepción tradicional y desde una perspectiva institucional y tecnológica, muchas no estaban preparadas. Ante este problema, es indispensable integrar nuevas tecnologías que agilicen y aseguren la integridad de tan importante y delicado proceso, como lo es, la toma de evaluación de conocimientos [10] [11].

Durante la pandemia, muchas plataformas que ofrecen servicios de videoconferencia y sistemas de gestión del aprendizaje (LMS) añadieron nuevas funciones a sus soluciones para adaptarse a las exigencias y demandas de la educación virtual. Esto permitió satisfacer en cierta medida las necesidades de enseñanza y aprendizaje en línea.

En el mercado tecnológico global, existen soluciones comerciales y *open source* que usan técnicas de reconocimiento facial para la supervisión remota, tales es el caso de: el complemento para *Moodle E-Learning, Faceidentity, Smowl y FaceAuthentic*, entre otras, que para las instituciones educativas, en algunos casos son inaccesibles, ya que su precio comercial es elevado y requieren de licencias que muy difícilmente pueden ser solventadas por las mismas, al menos para las instituciones públicas, ya que su presupuesto es limitado.

De acuerdo con Noguera [12], el reconocimiento facial es una de las áreas clave en la informática y representa la técnica biométrica más eficiente para la identificación de personas. Además, este tipo de tecnología es considerado como un método pasivo y no invasivo, ya que utiliza características faciales comunes y no requiere de software especializado para la captura de fotografías, sino que solamente se necesita una cámara web.

Este trabajo de investigación se enfoca en la aplicación de algoritmos de detección de rostros y modelos de reconocimiento facial utilizando una cámara web, con el objetivo de detectar suplantaciones de identidad en procesos remotos de evaluación. El análisis de imágenes y las técnicas de detección y reconocimiento facial han sido ampliamente investigadas en los últimos años [13] [14] [15], y se han aplicado en diversas áreas como la identificación biométrica para la clasificación de individuos [16] [17] [18], sistemas de seguridad y procesamiento de imágenes y video para mejorar la calidad y facilitar la búsqueda de información [20] [21], entre otras. Es un área en la que los investigadores han elaborado trabajos investigativos y aportes significativos, desarrollando nuevas soluciones y algoritmos cuyo tiempo de respuestas se acortan en cada aporte, ya se podría decir que, el tiempo de respuesta en algunos casos son medidos en milisegundos a diferencia de que hace unos pocos años el reconocimiento facial tardaban segundos [22].

FORMULACIÓN DEL PROBLEMA

¿Cómo se puede desarrollar un modelo de reconocimiento facial basado en inteligencia artificial que garantice una identificación precisa y segura de los estudiantes en la supervisión remota de exámenes en línea?

La supervisión remota de exámenes en línea es una práctica cada vez más común en el ámbito educativo, pero presenta desafíos en cuanto a la seguridad y confiabilidad del proceso. En este contexto, el reconocimiento de rostros mediante inteligencia artificial puede ser una herramienta útil para la identificación de los estudiantes durante el examen y la prevención de

fraudes. Sin embargo, la implementación de un modelo de reconocimiento de rostros efectivo y confiable para la supervisión remota de exámenes en línea plantea desafíos en cuanto a la precisión, la privacidad y la seguridad de los datos. De esta manera, la formulación del problema se enfoca en los desafíos y las incertidumbres que surgen al tratar de responder la pregunta científica. En este caso, se plantea la necesidad de diseñar un modelo de reconocimiento de rostros que sea efectivo y confiable para la supervisión remota de exámenes en línea, y se identifican los desafíos en cuanto a la precisión, privacidad y seguridad de los datos que deben ser abordados en la investigación.

Sistematización del problema

- ¿Cómo se desempeñan los modelos de detección y reconocimiento facial según las métricas de evaluación de algoritmos de aprendizaje automático?
- ¿Cómo puede el reconocimiento facial ayudar a solucionar el problema de la suplantación de identidad en exámenes en línea?

OBJETIVOS

Objetivo general

Desarrollar un prototipo de reconocimiento facial seleccionando algoritmos que apliquen IA para mitigar la suplantación de identidad durante el proceso de evaluación en entornos virtuales.

Objetivos específicos

- Investigar los fundamentos teóricos el reconocimiento facial.
- Evaluar el desempeño de tres algoritmos de reconocimiento facial mediante la realización de experimentos para medir precisión y robustez en la detección y verificación de rostros.
- Seleccionar el algoritmo adecuado para el proyecto a partir de la comparación de los resultados obtenidos en cada experimento.
- Determinar de manera precisa y eficiente si la persona presentada ante el sistema es o no quien afirma ser, identificando posibles suplantaciones de identidad mediante el análisis de características biométricas únicas del rostro y la comparación con registros previos.

Delimitación del campo de acción

La presente investigación se centra en el análisis comparativo de algoritmos de reconocimiento facial, para determinar cuál es el apropiado para el desarrollo de la propuesta a través de la experimentación. Los algoritmos que se comparan son: *HaarCascade, Eigenfaces, Face-recognition* y un modelo de Redes Neuronales Convolucionales llamado *FaceNet*. Así mismo se analizan las librerías de Python: *Dlib, Tensorflow y el framework CaffeModel*. Se realizan experimentos con cada uno de ellos, cada algoritmo fue entrenado con un *dataset* de 500 rostros. No se pretende plantear nuevos algoritmos, sino construir un prototipo utilizando los mejores resultados obtenidos durante la ejecución real de los algoritmos en un ambiente controlado que simula un entorno del proceso de evaluación en línea

Estructura del trabajo

La investigación se divide en los siguientes capítulos: el primero se enfoca en el estado del conocimiento de las herramientas empleadas en el trabajo de titulación; el segundo describe las metodologías y materiales utilizados; el tercer capítulo estudia los modelos, algoritmos y resultados de investigaciones previas para determinar el más adecuado para el cumplimiento del objetivo del presente trabajo; En el cuarto capítulo se realiza la discusión sobre los resultados obtenidos ya en la etapa de prueba con el algoritmo seleccionado y finalmente se presentan las conclusiones y recomendaciones.

CAPÍTULO I

1. MARCO TEÓRICO

1.1 ANTECEDENTES HISTÓRICOS

En 1883, Alphonse Bertillon sentó las bases del sistema de reconocimiento facial al utilizar medidas antropométricas como la distancia entre los ojos, la simetría y los rasgos faciales de un individuo. En la actualidad, este sistema es una práctica común en el ámbito forense, llegando incluso a ser utilizado en los tribunales para acusar o exonerar a individuos con antecedentes penales [23].

Desde los años 1950 se han llevado a cabo investigaciones sobre algoritmos y técnicas de detección de rostros. En aquel entonces, se realizaron los primeros experimentos en este campo, los cuales se aplicaron en el área de la psicología. A partir de estos primeros estudios, se realizaron otros que buscaban reconocer las distintas expresiones faciales, interpretar emociones o gestos percibidos en el rostro [24].

En 1960, se logró desarrollar un sistema que clasificaba rostros a partir de fotografías, el mismo que registraba las coordenadas de la nariz, los ojos, la boca y la línea del cabello. El matemático Woodrow Wilson Bledsoe, se valió de un dispositivo llamado tableta rand, la cual fue una de las primeras tabletas gráficas que existieron. Estas métricas fueron incorporadas a una base de datos, luego el dispositivo era capaz de devolver la imagen que más se parecía a una solicitada [25].

Durante las décadas de los 70 y 80, se emplearon plantillas y mediciones de características geométricas de partes del rostro para la detección y el reconocimiento de caras [26]. Es así que, en los setenta Toshiyuki Sakai, Makoto Nagao y Takeo Kanade[27], desarrollaron un método que utilizaba de igual forma técnicas heurísticas y antropométricas. Pero, a diferencia de las anteriores, estas dependían mucho de la calidad del entorno y estaba condicionado a los cambios y variaciones de luz y posición de la imagen, por lo que, significaba reajustar el modelo o en muchas ocasiones el rediseño completo del algoritmo. Durante la década de los setenta, investigadores como Goldstein, Harnon y Lesk [28] lograron mejorar la precisión del reconocimiento facial al utilizar técnicas que incluían características como el grosor del labio y el color del cabello para identificar caras de forma automática. En 1981, Lucas-Kanade [29], aplicaron una característica de suavidad, utilizando un algoritmo que tenía un ajuste de

mínimos cuadrados, el cual se aplicaba a una pequeña parte de la imagen. Dicho algoritmo puede ser aplicado en un contexto denso ya que se basa solamente en información específica que es derivada de algunas ventanas vecinas en cada punto de interés.

En el año de 1988, Sirovich y Kirby [30], desarrollaron la técnica de *Eigenfaces*, la cual requería al menos 100 conjuntos de vectores para determinar un rostro alineado y normalizado. En los siguientes años, la precisión de las técnicas de detección facial evolucionó, pero no fue hasta la década de los 90 que se dieron los primeros pasos hacia el reconocimiento automático. Turk y Pentland [31] utilizaron las técnicas de Eigenfaces, lo que permitió la realización de sistemas de detección de rostros en tiempo real y generó un interés significativo en posteriores desarrollos de este tipo de sistemas. Sin embargo, su enfoque seguía limitado por factores tecnológicos y ambientales, pero fue un avance significativo en la prueba de la viabilidad del reconocimiento facial automático [32].

Durante la década de 1990 a 2000, Sirovich y Kirby aplicaron el álgebra lineal al problema del reconocimiento facial. Durante este período, la Agencia de Proyectos de Investigación Avanzada de Defensa (DARPA) y el Instituto Nacional de Estándares y Tecnología lanzaron un programa de Tecnología de Reconocimiento Facial llamado FERET para impulsar el mercado comercial del reconocimiento facial. Este programa consistió en la creación de un conjunto de datos de rostros para la prueba, que incluía 2,413 imágenes de rostros representando a 856 individuos. El objetivo era inspirar a futuras investigaciones para mejorar e innovar en las tecnologías de reconocimiento facial existentes y desarrollar soluciones más avanzadas. Además, en 2003, se actualizó el conjunto de datos para incluir versiones en color de alta resolución de las imágenes [33].

En el año 2001, Paul Viola y Michael Jones crearon un sistema de detección de rostros que fue un gran avance en el campo del reconocimiento facial, debido a su rápida capacidad de procesamiento para identificar rostros. A diferencia de sus predecesores, este sistema clasificaba características extraídas en una escala de grises, en lugar de realizar la clasificación pixel por pixel en imágenes a color. En 2003, Gunnar Farneback desarrolló otro algoritmo que calcula el flujo óptico para todos los píxeles de la imagen, utilizando polinomios cuadráticos para aproximar conjuntos de píxeles que pertenecen a una misma región en dos fotogramas consecutivos. Esta técnica difiere del algoritmo de Lucas Kanade [34].

1.2 Antecedentes Conceptuales y referenciales

El estudio presenta las siguientes variables:

Variable dependiente: Desempeño de los algoritmos de reconocimiento facial

Variable independiente: Algoritmos de reconocimiento facial

1.2.1 Hipótesis de la investigación.

H1. Un modelo de reconocimiento facial basado en IA aplicado a la supervisión remota de exámenes en línea, mejorará la detección de suplantación de identidad identificando a una persona con una precisión mayor o igual al 98%.

H2. Existen diferencias significativas en el desempeño de los diferentes algoritmos de reconocimiento facial en términos de precisión y eficiencia.

1.3 Fundamentación teórica de la variable dependiente.

Desempeño de los algoritmos de reconocimiento facial

1.4 Antecedentes Contextuales

Las investigaciones en este campo han evolucionado significativamente en relación al tiempo dando paso a soluciones cada vez más efectivas. Es usado en diversos campos ocupacionales de la sociedad[36][37][38], tal es el caso, de la seguridad informática, área en el cual, con el avance de la tecnología y la proliferación de las aplicaciones, crecen considerablemente los niveles de fraudes de identidad, ocasionando estas, un problema social que provocan afectaciones económicas considerables[39], es por esto, que las organizaciones tienen la necesidad de implementar tecnologías que usen técnicas de reconocimiento facial como medio de autentificación, como medida de seguridad[40].

Durante un partido del Super Bowl en 2001, se emplearon cámaras de vigilancia para recolectar imágenes que fueron comparadas con imágenes digitalizadas de delincuentes en una base de datos [41]. Este evento, junto con el atentado a las torres gemelas ese mismo año, impulsó el uso de sistemas de biometría en aeropuertos y lugares públicos, aumentando la necesidad de sistemas de seguridad más avanzados y confiables asistidos por computadora. De hecho, el reconocimiento facial fue utilizado para confirmar la identidad de Osama Bin Laden después

de su muerte en una operación de los Estados Unidos.

Las técnicas de detección y reconocimiento de rostros se han usado mucho en el campo de la fotografía, siendo esta área la que acercó por primera vez al usuario este tipo de tecnologías. La empresa Fujifilm fue la pionera y por aquel entonces anunciaba la detección de rostros en sus dispositivos, esto en el año 2004, pero fue su competencia, Nikon, en el año 2005, la primera en sacar al mercado una cámara utilizando la detección de rostros [42]. Posteriormente, se dieron significativos avances en el uso de este tipo de tecnologías, ya que, en el 2010, se implementaron sitios web, específicamente en las redes sociales, la primera de ellas fue Facebook, que para ese entonces iniciaba su crecimiento.

Así mismo, en el 2014 se empieza a implementar el sistema ID en los teléfonos móviles de alta gama con distintos fines de seguridad. En año 2017 se implementó el uso del sistema ID en distintas aplicaciones las cuales tienen fines lucrativos (Like, tik tok, editor).

En el año 2017, Apple presentó su iPhone X con la función de reconocimiento facial para la seguridad del dispositivo. Esta característica fue muy bien recibida por los usuarios, lo que llevó a que se agotaran las existencias casi al instante y ahora es un estándar en dispositivos móviles. El reconocimiento facial se utiliza ampliamente en lugares donde se requiere control de acceso y verificación de identidad, como aeropuertos, terminales de transporte, así como para la búsqueda de personas desaparecidas y la prevención del fraude de suplantación de identidad [41]. Además, también se utiliza en sistemas de seguridad de banca electrónica, centros educativos y de investigación (a través de la huella dactilar), bases de datos confidenciales, y otros lugares con alto tráfico de personas [43][44]. Un ejemplo de su utilidad fue la identificación y detención de un individuo buscado por infracciones económicas en un concierto, entre una multitud de 60,000 asistentes [45].

1.5 TRABAJOS PREVIOS

El Banco de Guayaquil en Ecuador ha implementado un sistema de reconocimiento facial en su Banca Virtual Móvil, eliminando el uso de contraseñas y permitiendo a los usuarios realizar transacciones bancarias de manera rápida y segura [46][47]. Además, se han realizado investigaciones sobre el reconocimiento facial en Ecuador, como la realizada en la ciudad de Ambato por [48]. El objetivo de esta investigación fue desarrollar un prototipo de reconocimiento facial con visión artificial para apoyar al ecu-911 en la identificación de personas en la lista de los más buscados. Los resultados fueron aceptables, manteniendo un alto

nivel de confianza gracias al uso del Análisis de Componentes Principales (PCA) a través del algoritmo *Eigenfaces*.

Sin duda este tipo de tecnología se adapta a todo tipo de procesos, según las necesidades tecnológicas, tal es el caso del sector educativo, en donde muchos procesos que se generan en este campo se han respaldado de las tecnologías considerablemente, pero más, sin embargo, se puede hacer más.

El presente trabajo de investigación aborda el proceso vinculado con el control de identidad para mitigar la suplantación de identidad en exámenes en línea, a través del uso de técnicas de reconocimiento facial; para garantizar la tele presencia real de un estudiante mientras desarrolla una evaluación de conocimientos en línea; esto resulta un cambio significativo al compararlo con el proceso actual de toma de evaluaciones en línea, en donde en ciertas ocasiones no se puede llegar al punto de verificar la identidad real del estudiante; todo este proceso sería transparente para el docente y el sistema se encargará del monitoreo de asistencia y la verificación de identidad en cada momento del proceso académico.

Cabe mencionar que, para este tipo de soluciones ya se han realizado varias investigaciones, como, por ejemplo, Chintalapati y Raghunadh [49], propusieron un modelo para el sistema automatizado de asistencia de detección de rostros en forma presencial, el cual se centraba principalmente en los algoritmos de detección y reconocimiento de rostros, donde pudieron evidenciar que funcionaba de manera muy eficiente y pudieron reconocer fácilmente a los estudiantes cuando entraban al salón de clases. De igual forma, Hidalgo V [50], diseñó un prototipo el cual permitió detectar rostros de forma eficaz, bajo un ambiente controlado de factores externos permitiendo con esto la gestión y control en el registro de estudiantes asistentes en un aula. En su investigación se determinó el algoritmo adecuado para la detección facial y reconocimiento de rostros, eligiendo para su proyecto, el algoritmo de Viola-Jones y *Eigenfaces*, estos métodos dieron como resultado una alta tasa de precisión bajo condiciones controladas de iluminación, posición y orientación de la cara.

Dentro del apartado de verificación de identidad, también se han realizado propuestas con base en la investigación sobre el tratamiento de imágenes, tal es el caso de una investigación que se realizó en [51] en donde se propone el reconocimiento facial en tiempo real en videollamadas o *live stream* para autenticar identidades durante una audiencia legal, usaron la herramienta DLIB, la cual es una librería que ayuda a detectar objetos, en este caso el rostro de una persona.

En su investigación concluye que si es posible identificar a una persona en tiempo real, pero en algunos casos se presentan algunas excepciones, por lo que recomienda al usuario ejecutar varias veces el algoritmo durante la videollamada con el fin de corroborar que realmente el reconocimiento facial, así también, recomienda que la resolución de la cámara y de las imágenes almacenadas para el entrenamiento sean las adecuadas, ya que es fundamental para que el sistema de reconocimiento facial pueda identificar de manera acertada el rostro de una persona.

A continuación, se analizan los trabajos previos desarrollados para la detección de rostros (Tabla 1), donde se describen los problemas encontrados de las técnicas ante la aplicación a secuencias de vídeo de baja calidad o si los fotogramas con las que fueron probadas los trabajos fueron extraídos de entornos controlados (interiores) o exteriores.

Investigación	**Técnica**
"A Study on Online Examination using Face Recognition Technology", 2018	Eigenface
"Facial Recognition for Online Exam Proctoring" (2019)	OpenCV
"Remote Proctoring using Face Recognition Technique for Online Examination" (2019)	Eigenface
"Online Remote Proctoring Using Artificial Intelligence Techniques" (2019)	Redes neuronales convolucionales (CNN)
"Face Recognition in Online Examination for Secured Testing" (2019)	Fisherfaces
"Automated Remote Proctoring in Online Examinations" (2020):	Redes neuronales convolucionales (CNN)
"A novel facial recognition approach for online examination monitoring system" de Md. A. Hossain y col. (2019)	Reconocimiento facial basado en características de textura y color - Local Binary Patterns (LBP)
"A supervised learning-based approach for facial recognition using deep learning model for online examination monitoring" de Prateek Agrawal y col. (2018).	Redes neuronales convolucionales (CNN)

"A facial recognition-based system for remote examination supervision" de S. Roy y col. (2020).	Red neuronal convolucional (CNN)
"An Efficient Facial Recognition System for Online Examination Monitoring using Hybrid CNN-LSTM Model" de Anamika Singh y col. (2021).	modelo de reconocimiento facial híbrido basado en redes neuronales convolucionales y de memoria a largo plazo (CNN-LSTM)

Tabla 1. Análisis de trabajos existentes actualmente.

Fuente: Autor

El análisis comparativo de los estudios previos sobre algoritmos de reconocimiento facial en el contexto de la supervisión remota de exámenes en línea revela una serie de relaciones y divergencias relevantes para nuestra investigación. Por un lado, varios estudios, como el de *'A Study on Online Examination using Face Recognition Technology*' (2018) y *'Online Remote Proctoring Using Artificial Intelligence Techniques*' (2019), destacan el potencial de los algoritmos basados en redes neuronales convolucionales (CNN) para lograr altas tasas de precisión en el reconocimiento facial, superando el umbral del 90%. Estos resultados respaldan la elección de los algoritmos basados en CNN que consideramos para nuestra investigación.

Sin embargo, también se observan diferencias significativas en las técnicas utilizadas por los estudios previos. Por ejemplo, algunos estudios han empleado algoritmos como *Eigenface, OpenCV y Fisherfaces*, que se basan en diferentes enfoques y características del rostro. Estos algoritmos han demostrado su eficacia en el reconocimiento facial y han arrojado tasas de precisión prometedoras en el rango del 91% al 94%. Además, se ha explorado el uso de técnicas basadas en características geométricas de la cara, como el estudio de *'Remote Proctoring using Face Recognition Technique for Online Examination*' (2019), que obtuvo una tasa de precisión del 92,1%. Estas divergencias en las técnicas resaltan la diversidad de enfoques en el campo y nos brindan una oportunidad para evaluar y comparar el desempeño de diferentes algoritmos en nuestro propio estudio.

Considerando estas relaciones y divergencias, la presente investigación propone profundizar en el análisis comparativo del desempeño de estos algoritmos de reconocimiento facial. Para lograrlo, se lleva a cabo una metodología experimental rigurosa, que incluirá la recopilación y preparación de un conjunto de datos representativo (dataset), la configuración de los experimentos, la implementación de los algoritmos y la evaluación utilizando métricas

específicas de reconocimiento facial. El objetivo es determinar cuál de estos algoritmos es el más adecuado para la supervisión remota de exámenes en línea, considerando la precisión del reconocimiento facial. Además, se busca identificar las fortalezas y debilidades de cada enfoque y explorar posibles mejoras o combinaciones de técnicas para optimizar el desempeño.

1.6 Fundamentación teórica de la variable independiente.

Debido a la diversidad que tiene un rostro humano y sus rasgos físicos que lo caracterizan y hacen único a un individuo, a lo largo del tiempo, se han realizado significativos avances en la identificación de rostros de personas para poder reconocer su identidad; para lograr este objetivo, se necesita extraer los rasgos biométricos de una persona [52], por ello, a este campo se lo ha denominado biometría informática, la cual, es el uso el uso de técnicas matemáticas y estadísticas que se aplican sobre las características físicas de los individuos para determinar su identidad o verificar su autenticidad. A continuación, se describen las consideraciones iniciales y conceptos previos, detallando el estado del arte dentro de cada concepto, se explica lo básico para la mejor comprensión de la investigación presentada.

1.6.1 Biometría

La biometría es la disciplina que se encarga de identificar a una persona a través de características fisiológicas o de comportamiento únicas. La palabra deriva del griego "bios", que significa vida, y "metron", que significa medida. En lugar de utilizar métodos tradicionales como contraseñas, códigos o firmas que pueden ser copiados o manipulados para fines delictivos, la biometría se basa en la verificación digital y científica de estas características únicas para identificar a una persona, lo que la hace más segura. (53)

Existen diferentes tipos de biometría, que se pueden clasificar según la fisiología, como la huella dactilar, el escaneo de iris, el escaneo de retina, el reconocimiento facial, el escaneo de la geometría de la mano, entre otros; y según el comportamiento, como la firma personal o el comportamiento en el uso del computador, entre otras. (54).

1.6.2 Inteligencia artificial

La Inteligencia Artificial (IA) es un campo de la informática que se dedica a desarrollar sistemas informáticos capaces de llevar a cabo tareas que requieren inteligencia humana, como el reconocimiento de patrones, el aprendizaje y la toma de decisiones. Dentro de este campo,

existen múltiples tecnologías y enfoques, tales como el aprendizaje automático, el procesamiento del lenguaje natural y el análisis de datos, entre otros. Según [55], describe a la IA, como: "*el tratamiento y combinación de algoritmos para desarrollar una variedad de sistemas, máquinas y robots para que estos, puedan de una forma sistemática realizar las mismas tareas que realizan los humanos*".

Figura 1. Ramas de la IA.

Fuente: Tomado de [55]

La IA tiene como objetivo imitar las capacidades cognitivas de los seres humanos, como la percepción, el pensamiento y el razonamiento, para poder realizar tareas de manera autónoma o para mejorar la eficiencia y la precisión de ciertas tareas. Los sistemas de IA pueden ser utilizados en una amplia variedad de aplicaciones, como el reconocimiento de voz, el procesamiento de lenguaje natural, la toma de decisiones y el aprendizaje automático. Aunque la IA ha avanzado mucho en las últimas décadas, sin embargo, todavía hay mucho por descubrir y muchos desafíos por superar. Algunos de los desafíos a los que se enfrenta la IA incluyen la complejidad de imitar la inteligencia humana, la necesidad de grandes cantidades de datos para el entrenamiento y la preocupación por la privacidad y la seguridad de los datos utilizados para el aprendizaje automático [55].

En la actualidad, la inteligencia artificial (IA) continúa en constante desarrollo y evolución, generando cada vez más interés en investigaciones y dedicando más recursos a su estudio, debido a su amplia aplicabilidad en diversos campos de la sociedad. En la figura 2, se puede observar una infografía realizada por Statista (Las aplicaciones más rentables de la inteligencia

artificial | Statista, n.d.), que muestra las aplicaciones de la IA clasificadas según los ingresos generados.

Figura 2. Ingresos globales de la I.A de 2016 a 2025 (en mill. Euros)

Fuente: Tomado de [56]

Es importante mencionar que el campo de la IA cuenta con varias ramas, entre ellas las redes neuronales, el machine learning, el deep learning y la visión por computador, entre otras (ver Figura 1). Estos conceptos están estrechamente relacionados con el tema del proyecto y serán fundamentales para darle forma y contexto a la investigación. A continuación, se describen brevemente cada uno de ellos.

1.6.3 Aprendizaje de máquina vs Aprendizaje profundo

El aprendizaje de máquina es una técnica de inteligencia artificial que se basa en el uso de algoritmos de aprendizaje automático para permitir que una computadora "aprenda" a partir de datos sin ser explícitamente programada. El campo del aprendizaje de máquina se divide en dos categorías principales: el aprendizaje supervisado y el aprendizaje no supervisado. [57]. El aprendizaje supervisado involucra el uso de datos etiquetados que se utilizan para "enseñar" a la máquina a realizar una tarea específica. Por ejemplo, se pueden utilizar imágenes etiquetadas de gatos y perros para "enseñar" a una computadora a reconocer gatos y perros en imágenes no etiquetadas.

El aprendizaje profundo es una técnica de aprendizaje de máquina que se basa en el uso de redes neuronales artificiales de gran profundidad y complejidad para llevar a cabo tareas de aprendizaje automático. Las redes neuronales, inspiradas en el funcionamiento del cerebro

humano, son un tipo de algoritmo utilizado para procesar y analizar grandes cantidades de datos. El aprendizaje profundo ha tenido un impacto significativo en numerosas aplicaciones de IA, como el reconocimiento de patrones, el procesamiento del lenguaje natural y el análisis de imágenes. [58][59].

1.6.4 Visión Computacional

Es una de las disciplinas de la IA, consiste en adquirir, procesar, analizar y comprender las imágenes del mundo real adquiridas por dispositivos digitales, con el fin de obtener información estructurada de forma numérica que pueda ser procesada por una computadora [60]. Existen numerosas aplicaciones en donde esta disciplina ha dado solución a determinados problemas de la sociedad [61].

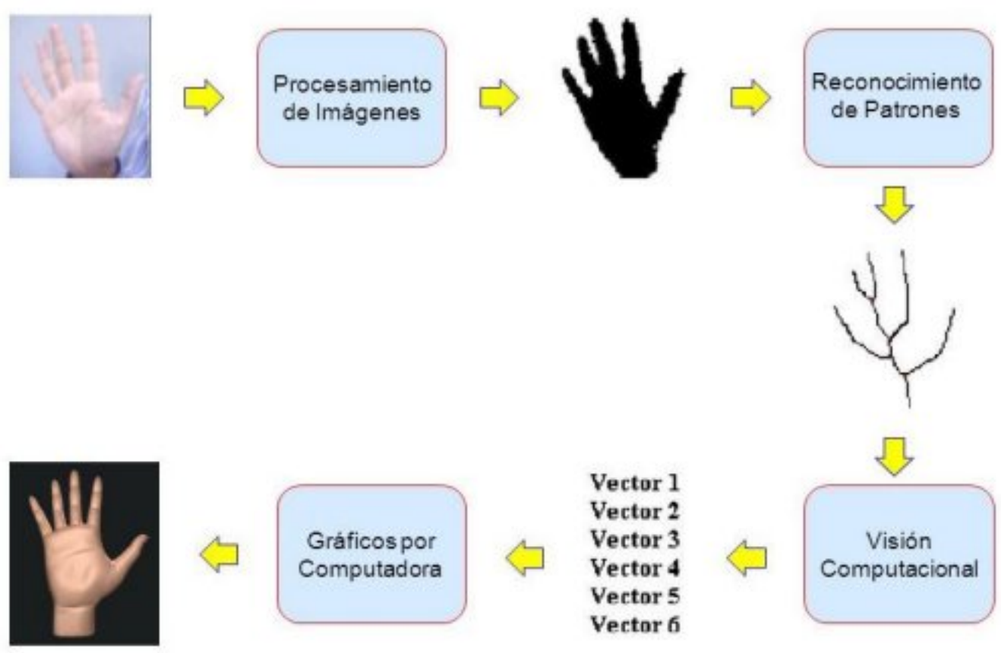

Figura 3. Disciplinas de la visión computacional

Fuente: Autor.

La visión computacional es el estudio de cómo procesar y analizar imágenes y videos digitales utilizando técnicas de informática y algoritmos de procesamiento de imágenes. La visión computacional se utiliza ampliamente en una amplia variedad de aplicaciones, como el reconocimiento de patrones, el análisis de imágenes médicas, la vigilancia de seguridad y el reconocimiento de voz [62][63].

1.6.5 Aprendizaje automático (*Machine Learning*)

El Aprendizaje Automático o *Machine Learning* es un subcampo de las ciencias de la computación y una rama importante de la inteligencia artificial. Consiste en estudiar y diseñar algoritmos que, a partir de datos de ejemplo, puedan realizar predicciones mediante la

aplicación de distintos tipos de reglas y métodos estadísticos para manejar patrones complejos. Este proceso se llama "entrenamiento", donde los datos utilizados para el entrenamiento se conocen como "conjunto de entrenamiento", mientras que los datos sobre los que se realizan las predicciones pero que no formaron parte del entrenamiento se denominan "conjunto de evaluación". En resumen, el Machine Learning se encarga de generar algoritmos que tienen la capacidad de aprender sin necesidad de ser programados de manera explícita, utilizando métodos o técnicas estadísticas. De esta forma, el programador no tiene que desarrollar funciones para todos los escenarios posibles o definir todas las excepciones posibles. En su lugar, solo debe proporcionar datos al algoritmo para que pueda aprender y saber cómo actuar en cada uno de los casos posibles. [64].

Figura 4. Ciclo de vida del Machine Learning

Fuente: Tomado de [64]

Aunque hay muchas categorías dentro del *Machine Learning*, en este trabajo de investigación se enuncian los siguientes:

- Algoritmos de aprendizaje supervisado
- Algoritmos de aprendizaje no supervisado
- Algoritmos de aprendizaje semi supervisado
- Algoritmos de aprendizaje por refuerzo.

Algoritmos de aprendizaje supervisado.

El aprendizaje supervisado es un tipo de aprendizaje de máquina que utiliza una variable de entrada y una variable de salida, y se basa en descubrir la relación existente entre ambas. Este tipo de aprendizaje se produce al enseñar a los algoritmos cuál es el resultado deseado para un valor determinado, mostrándoles numerosos ejemplos. Si se dan las condiciones adecuadas, el algoritmo será capaz de producir resultados precisos incluso para valores que no haya visto antes. En esencia, el aprendizaje supervisado implica la identificación de patrones en los datos de entrenamiento para poder hacer predicciones precisas en nuevos conjuntos de datos.

El objetivo es encontrar una función matemática con base en los datos de entrada y salida procurando obtener la más acertada posible de manera que al tener nuevos datos de entrada se puedan predecir las variables de salida [65].

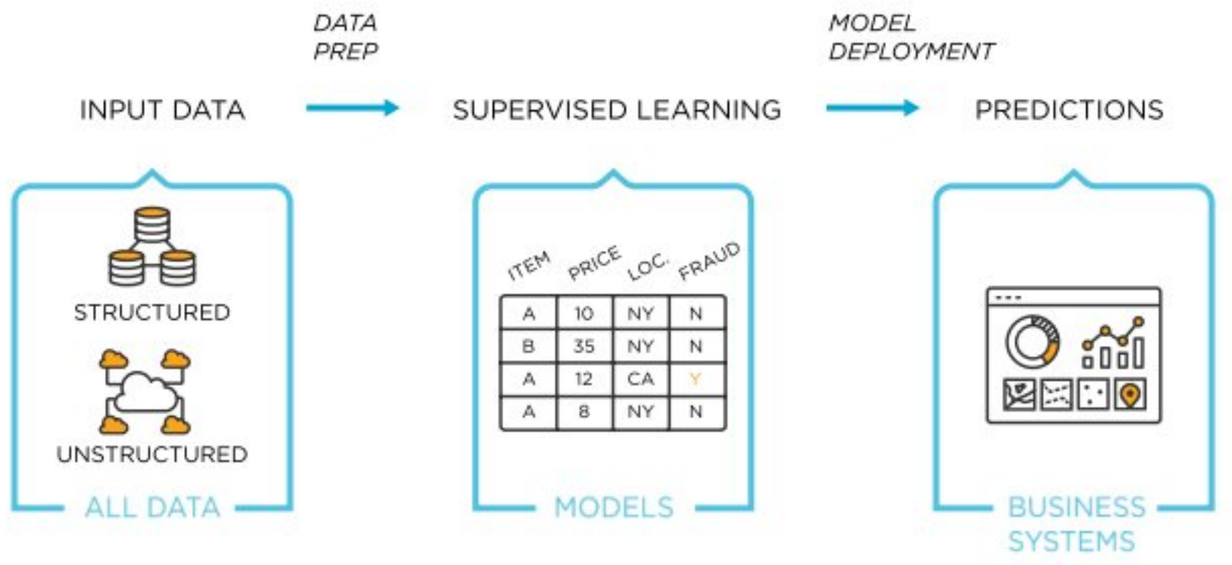

Figura 5. Algoritmos de aprendizaje supervisado

Fuente: Tomado de [65]

Algoritmos de aprendizaje no supervisado

El paradigma al que se hace referencia es capaz de generar conocimiento a partir de los datos que se le proporcionan como entrada, sin requerir una indicación previa del resultado deseado por el usuario. A diferencia del aprendizaje supervisado, en el que el sistema espera una respuesta correcta, aquí no hay tal expectativa. Estos algoritmos suelen emplear técnicas de agrupación y asociación para clasificar los datos.

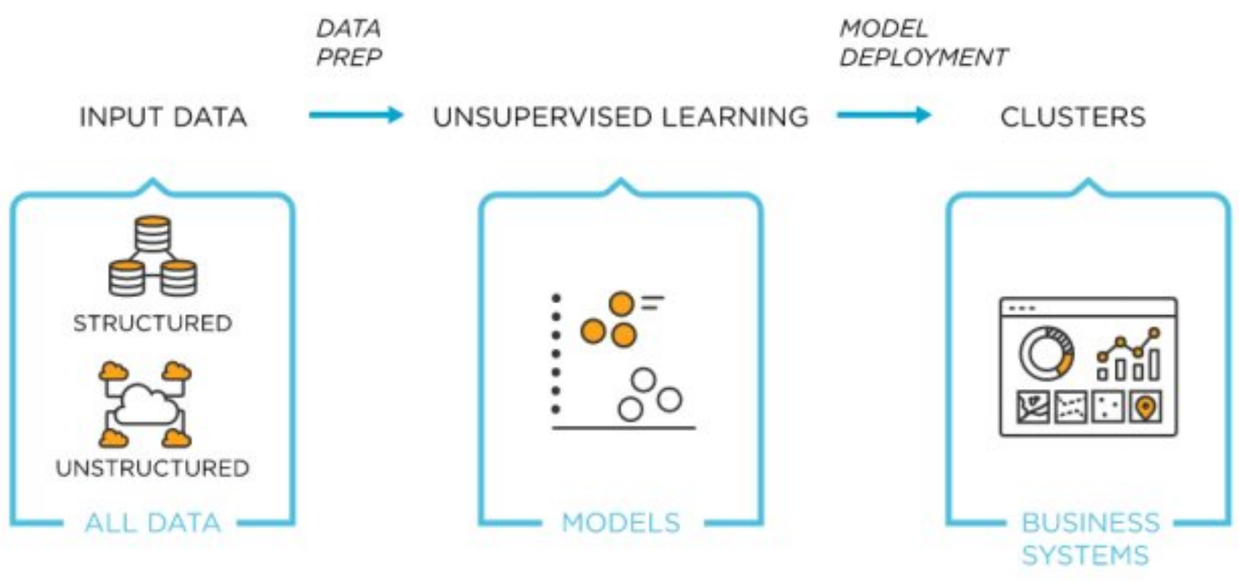

Figura 6. Algoritmos de aprendizaje no supervisado

Fuente: Tomado de [65]

Algoritmos de aprendizaje semi supervisado.

Es una mezcla de aprendizaje supervisado y no supervisado, es empleado cuando se tiene se tiene gran cantidad de datos de entrada y pocos datos etiquetados a la salida [67]. Existen muchos casos de uso que encajan perfectamente con este tipo de algoritmos, debido a al

volumen de información o de datos que amerita el problema, ya que etiquetar una gran cantidad de datos conlleva mucho tiempo mientras que los datos no etiquetados son más fáciles de recopilar y almacenar con dichos algoritmos.

Algoritmos de aprendizaje por refuerzo.

El AR o por sus siglas en inglés RL (*Reinforcement Learning*), es el más común de entre las categorías de ML. El objetivo principal es determinar qué acciones debe escoger el sistema ante un entorno dado con el fin de maximizar una recompensa, es decir no necesita indicaciones o que el programador le indique que hacer, el resultado se verá inducido por una simple señal de recompensa. El sistema aprende de experiencias pasadas para en cada iteración mejorar sus respuestas. El AR, representa lo que se conoce comúnmente como IA por aprendizaje automático [68]. El aprendizaje automático tiene una amplia gama de aplicaciones, incluyendo algoritmos utilizados para filtrar correos no deseados [69], reconocimiento de lenguaje escrito y hablado [70], así como también reconocimiento facial, como se discutirá en esta investigación.

1.6.6 Reconocimiento facial

Cualquier sistema de reconocimiento facial puede ser por lo general dividido en cuatro fases: detección, preprocesado, extracción de características y comparación y clasificación [71].

Figura 7. Etapas del reconocimiento facial

Fuente: Tomada de [71].

1.6.7 Detección de rostros

La detección de rostros es una tarea de visión computacional que se refiere al proceso de encontrar y localizar rostros en imágenes o vídeos, cuando se trata de un video, se habla del seguimiento de rostros. La detección de rostros es un problema importante en una amplia variedad de aplicaciones, como el reconocimiento facial, la vigilancia de seguridad y el análisis

de sentimientos [72]. Existen diferentes enfoques para la detección de rostros, como la detección basada en modelos y la detección basada en aprendizaje automático. La detección basada en modelos utiliza un conjunto predefinido de características faciales para detectar rostros en imágenes, mientras que la detección basada en aprendizaje automático utiliza técnicas de aprendizaje automático para "aprender" a detectar rostros a partir de datos etiquetados. [73]

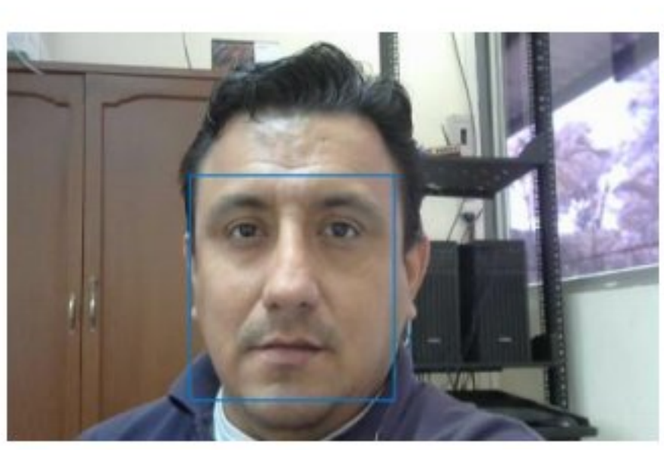

Figura 8. Detección de un rostro

Fuente: Autor.

Algunas investigaciones realizadas en el campo de detección de rostros han realizado clasificaciones para diferenciar entre algunos métodos y otros, según su fundamento. Tal es el caso de Yang et al. [74], el cual distingue los métodos que se basan en el conocimiento, los basados en características invariantes, los que hacen uso de plantillas o modelos y los que se basan en apariencia. Así mismo, con base a las investigaciones realizadas han surgido técnicas que apoyan el proceso de detección mediante la descripción de las características del rostro [75], las cuales se basan en las características propias de las imágenes, tales como: los bordes, cambios en la intensidad de los pixeles, gradientes de bordes, entre otros.

1.6.8 Métodos, algoritmos y bibliotecas

Dentro de los métodos y técnicas de detección de rostros desarrolladas tenemos a:

1) *HaarCascade* [76]: Este es un algoritmo de detección de objetos basado en el uso de una cascada de clasificadores débiles, que se entrenan a partir de un conjunto de imágenes de rostros y no rostros.
2) Detección basada en histograma de gradientes orientados (HOG) [77]: Este algoritmo utiliza una representación basada en gradientes de la imagen para detectar características clave en una imagen y determinar si hay un rostro presente.
3) Detección basada en redes neuronales [78]: Estos algoritmos utilizan redes neuronales para aprender a detectar rostros a partir de un conjunto de imágenes etiquetadas

previamente.

4) Detección basada en aprendizaje profundo [79]: Estos algoritmos utilizan técnicas de aprendizaje profundo, como las redes neuronales convolucionales, para aprender a detectar rostros a partir de imágenes etiquetadas.
5) Detección basada en máscara facial [80]: Este algoritmo utiliza una máscara facial para detectar el contorno del rostro y luego busca características específicas, como ojos, nariz y boca, para confirmar la presencia de un rostro.

Resultaría extenso nombrar y describir todos los métodos y técnicas de detección existentes debido a la gran variedad de algoritmos (que además a su vez presentan variantes), en la presente investigación se listan tres que se han considerado de relevancia para cumplir con el objetivo del proyecto.

1.6.8.1 Detección usando descriptores Haar

El algoritmo de detección facial más conocido es el método de Viola-Jones, basado en los descriptores *Haar*, que se describe en el artículo técnico "*Rapid Object Detection using a Boosted Cascade of Simple Features*" [76]. Este método cuenta con más de 20,000 citas en la actualidad, ya que fue uno de los primeros algoritmos de detección de objetos en ofrecer buenos resultados en tiempo real y sin consumir grandes cantidades de recursos. A pesar de haber sido desarrollado a principios del siglo XXI, su uso sigue siendo muy extendido en aplicaciones cotidianas como en teléfonos móviles o cámaras fotográficas, siendo uno de los algoritmos más populares y efectivos para la detección de rostros en imágenes y videos en tiempo real.

El algoritmo utiliza una técnica de aprendizaje automático llamada "aprendizaje por habilidades" para detectar rostros en una imagen o un video. Para hacer esto, primero se entrena un clasificador utilizando un gran conjunto de imágenes etiquetadas que contienen rostros y no rostros. Luego, el clasificador se utiliza para detectar rostros en imágenes o videos nuevos.

El primer modelo de cascadas de *Haar* se diseñó específicamente para detectar rostros. Además de este modelo específico, también existen otras cascadas que pueden utilizarse para detectar distintos elementos, como:

- Ojos, nariz y boca [81]
- Cuerpo entero (*fullbody*) [82]
- Parte baja del cuerpo (*lowerbody*) [82]
- Parte superior del cuerpo (*upperbody*) [83]

Se pueden llegar a detectar cualquier patrón con las cascadas de Haar, solo se tiene que tener un conjunto de imágenes positivas de detección y un conjunto de imágenes negativas.

Características o *kernels*

Viola y Jones [76] definieron las primeras características utilizadas para la detección de rostros, que consistían en cuatro patrones que se aplicaban en diferentes zonas de la imagen. Estas características se pueden visualizar en la Figura 9, en la que las áreas oscuras se restan de las áreas claras.

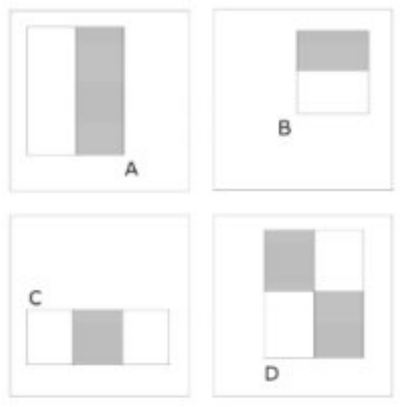

Figura 9. Características o kernels convolucionales

Fuente: Tomada de [76].

Estas características se pueden considerar como *kernels* convolucionales, ya que se aplican a cada píxel de una imagen en diferentes escalas. Un ejemplo de cómo se aplican estos *kernels* a una imagen se muestra en la Figura 10. Para aplicarlos a diferentes escalas (multi-escala), se reduce el tamaño de la imagen y se vuelve a aplicar el *kernel* a toda la imagen. De esta manera, se pueden detectar objetos de interés en diferentes escalas y posiciones dentro de la imagen que se está procesando [81].

Figura 10. Ejemplo de la aplicación de los kernels convolucionales a través de la imagen y su aplicación en múltiples escalas

Fuente: Tomada de [81].

Imágenes Integrales

Una de las principales aportaciones de Viola y Jones [76], fue el uso de imágenes integrales o tablas de sumas de áreas. Estas imágenes integrales son tablas de dos dimensiones del mismo tamaño que la imagen que se está procesando. Cada elemento de la tabla contiene la suma de los valores de todos los píxeles ubicados encima y a la izquierda del elemento de referencia (Ec. 1).

$$ii\,(x,y) = \sum_{x' \le x,\, y' \le y} i\,(x',y') \tag{1}$$

Después de calcular la imagen integral, es posible calcular la suma de los píxeles de cualquier rectángulo en la imagen con solo 3 operaciones y 4 accesos a la imagen integral. La Figura 8 muestra el área S que se calcula utilizando los puntos A, B, C y D con la Ecuación 2.

$$S = I(C) + I(A) - I(B) - I(D) \tag{2}$$

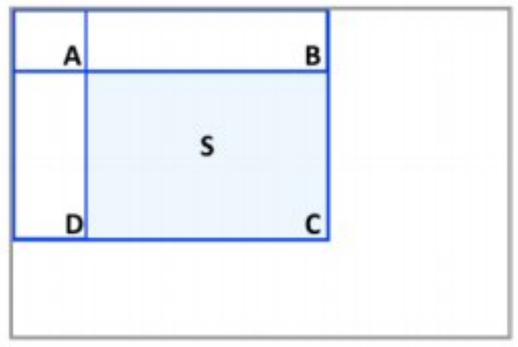

Figura 11. Cálculo de una región S dentro de la Imagen Integral, para lo cual son necesarios los puntos A, B, C y D (Ecuación 2).

Clasificador en cascada

La aplicación de *kernels* a una imagen en diferentes ubicaciones y tamaños resultaría en miles de posibilidades y un gran número de cálculos. Sin embargo, no todos estos valores son relevantes, ya que algunos pueden aplicarse en regiones sin cambios significativos o uniformes. Para solucionar esto, se utiliza la técnica de *AdaBoost* desarrollada por Freund y Schapire [84].

"Durante el proceso de entrenamiento del clasificador en cascada, se utiliza un conjunto de imágenes positivas y un conjunto de imágenes negativas. La técnica *AdaBoost* [84] selecciona las características con el menor error, para lo cual un clasificador débil calcula un umbral adecuado para clasificar correctamente las imágenes positivas y negativas. El clasificador final

(clasificador fuerte) es una combinación de los clasificadores débiles (Figura 12). Los clasificadores débiles son llamados así porque por sí solos no son capaces de clasificar correctamente las imágenes, pero en conjunto pueden actuar como un clasificador fuerte [76].

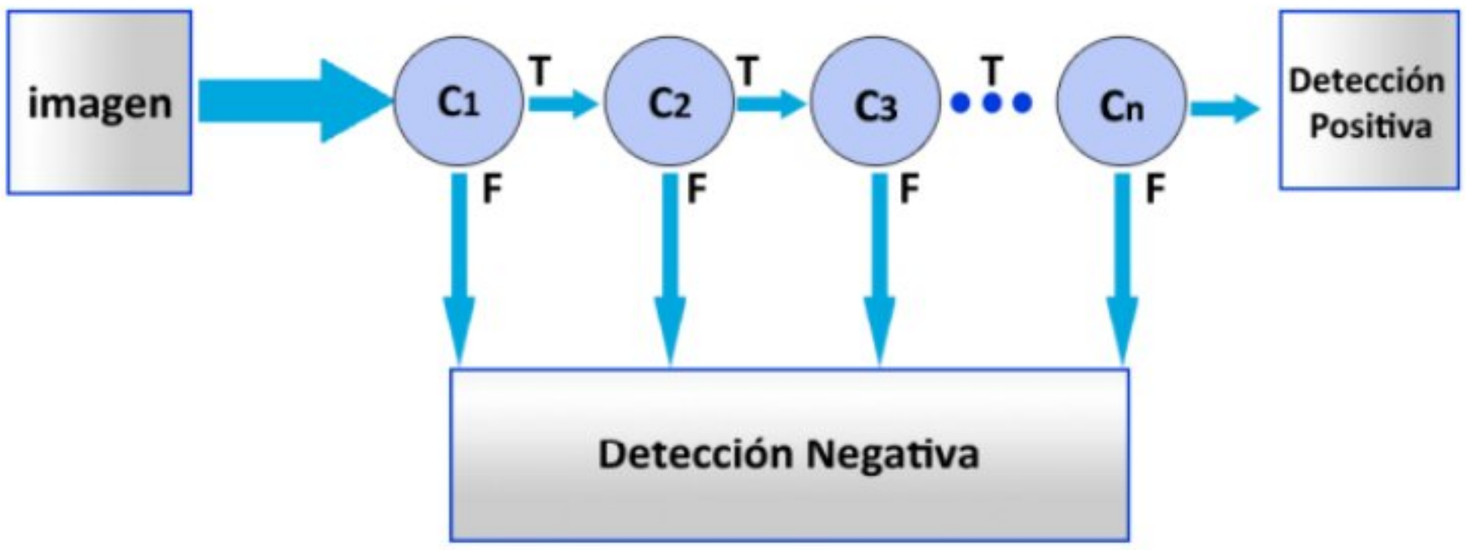

Figura 12. Ejemplo de clasificador en cascada. Las detecciones falsas(F) no pasan al siguiente clasificador (Cn).
Fuente: Tomada de [84].

En cada etapa del clasificador en cascada, se aumenta la complejidad de los clasificadores binarios, lo que permite eliminar rápidamente regiones que no contienen los objetos de interés. Si el objeto de interés no se encuentra en ninguna etapa del clasificador, se elimina de forma inmediata, evitando así la ejecución de clasificadores más intensivos del clasificador en cascada y ahorrando recursos computacionales [76]. El algoritmo de Viola y Jones utiliza distintas características (Haar), tal como las que aparecen representadas en la figura 13:

Figura 13. Características Haar
Fuente: Tomada de [76].

A continuación, se describe un código básico en *Python* para la detección de un rostro usando un modelo pre entrenado de *haarCascade* de la librería *OpenCv*.

```python
import cv2
# Cargar el clasificador de rostro Haar
face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
# Iniciar la cámara web
cap = cv2.VideoCapture(0)
while True:
    # Leer un frame de la cámara
    _, frame = cap.read()
    # Convertir a escala de grises
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    # Detectar rostros en el frame
    faces = face_cascade.detectMultiScale(gray, 1.3, 5)
    # Dibujar un rectángulo alrededor de cada rostro detectado
    for (x,y,w,h) in faces:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(255,0,0),2)
    # Mostrar el frame con los rectángulos dibujados
    cv2.imshow('frame', frame)
    # Salir con la tecla 'q'
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
# Liberar la cámara y cerrar las ventanas
cap.release()
cv2.destroyAllWindows()
```

Figura 14. código básico en Python para la detección de un rostro usando HaarCascade – OpenCv

Fuente: Autor.

El resultado de la detección de rostros usando el algoritmo propuesto usando haar

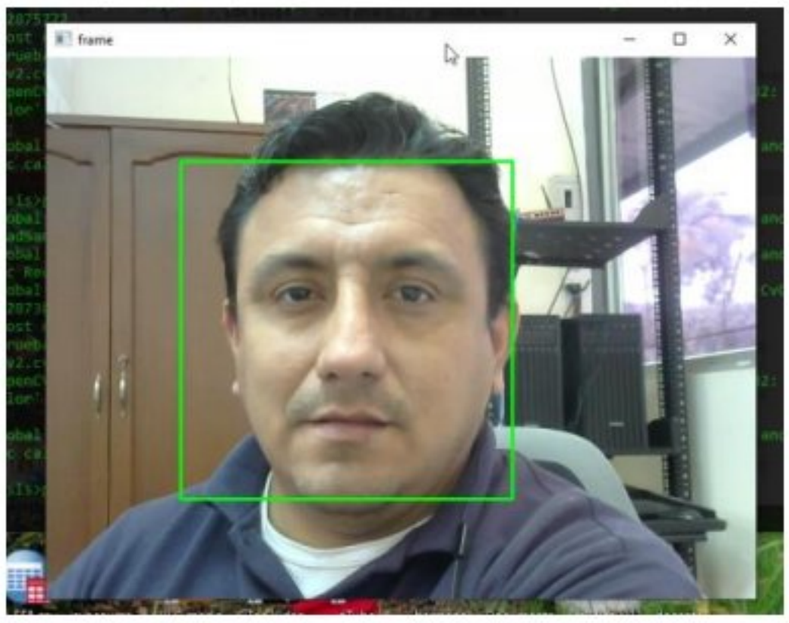

Figura 15. Resultado del Algoritmo

Fuente: Autor

1.6.8.2 Detección usando *deep learning*

1.6.8.2.1 Red neuronal convolucional

La arquitectura de una Red Neuronal Convolucional está formada por diferentes capas que transforman la entrada en un conjunto de salida. A continuación, se describen algunas de las capas más utilizadas:

Capa de Convolución

La capa de convolución es una parte fundamental de la estructura de una red neuronal convolucional. Está compuesta por un conjunto de neuronas que conectan regiones pequeñas de la entrada a una capa anterior. Estas regiones se conocen como filtros y pueden tener diferentes tamaños. Para cada una de estas regiones, se calcula el producto punto entre los pesos y los valores de entrada, a los cuales se suma el valor del bias. El filtro se mueve a través de la entrada tanto vertical como horizontalmente, repitiendo el cálculo del producto punto en cada región (convolución). La cantidad de desplazamiento, tanto vertical como horizontal, se llama "stride". Como resultado, se obtiene una capa de filtros que se activan cuando detectan ciertos tipos de características específicas en una determinada posición de la entrada [85].

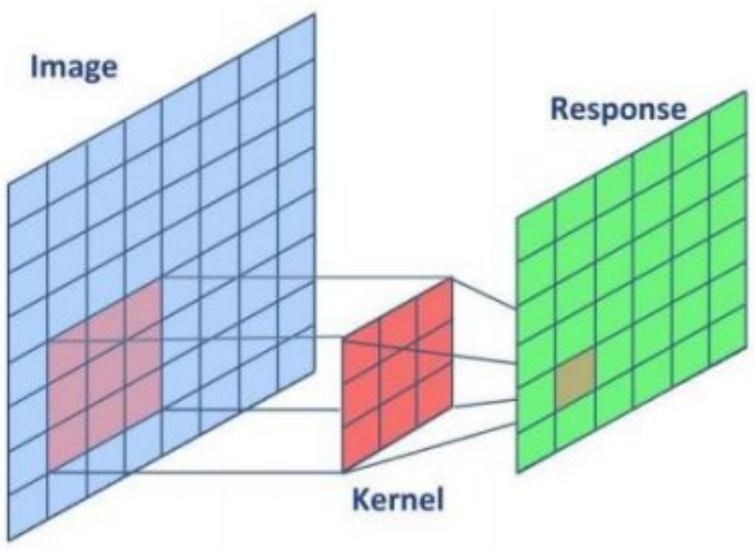

Figura 16. Ejemplo de convolución.

Fuente: Tomada de [85]

Pooling

La función de *Pooling* reduce progresivamente el tamaño espacial de la representación con el fin de reducir la cantidad de parámetros y cálculos en la Red. Hay varias funciones no lineales que se pueden utilizar como funciones de *pooling*, como el promedio (Figura 17) o la norma L2, pero la más utilizada es *max-pooling* (Figura 18). La función de *max-pooling* devuelve los valores máximos de regiones rectangulares no superpuestas de una entrada. Estas regiones rectangulares se delimitan por un filtro, a menudo de tamaño 2x2, que se escanea a través de la entrada tanto vertical como horizontal en pasos específicos (la distancia de desplazamiento se conoce como *stride* y su valor más utilizado es 2). La función de *max-pooling* no realiza ningún aprendizaje, su principal objetivo es reducir la cantidad de parámetros. [85].

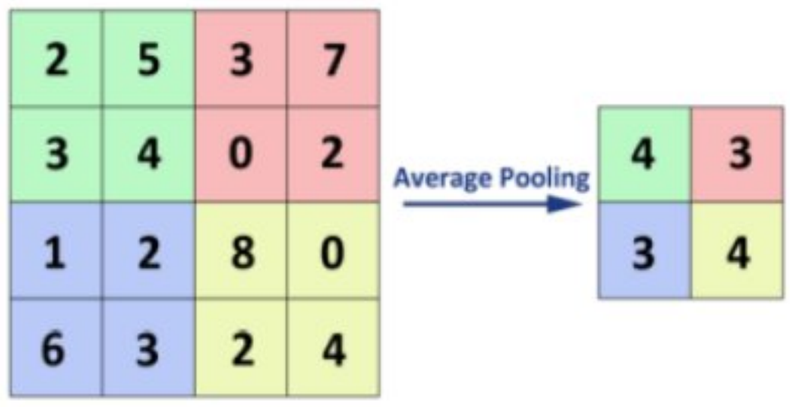

Figura 17. Average Pooling con filtro 2x2 y stride = 2

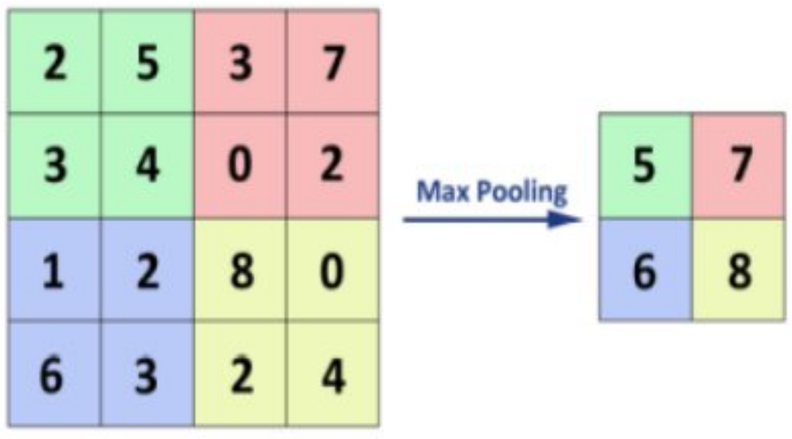

Figura 18. Max Pooling con filtro 2x2 y stride = 2

Fuente: Tomada de [85]

Rectified Linear Unit

La función *Rectified Linear Unit* (*ReLU*) aumenta las propiedades no lineales de las funciones de decisión, mejorando la robustez de las características extraídas. Además, ayuda a detectar neuronas que devuelven un resultado de cero en cada entrada, lo que puede hacer que el proceso de aprendizaje sea más computacionalmente eficiente [85]. *ReLU* lleva a cabo una operación con un umbral de valor, donde todos los valores se conservan o se llevan a cero (Ec. 3).

$$f\,(x) \;=\; max\,(0, x) \tag{3}$$

Fully Conected Layer

Las capas totalmente conectadas son utilizadas al final de una Red Neuronal Convolucional para convertir la salida de las capas anteriores a una forma legible por el usuario. Por ejemplo, en el caso de la clasificación de imágenes, las capas totalmente conectadas pueden ser utilizadas para convertir la salida de las capas anteriores en una probabilidad de pertenencia a cada una de las clases. Estas capas también pueden ser utilizadas para reducir la dimensionalidad de la salida y para adaptarla a la tarea específica que se esté realizando.

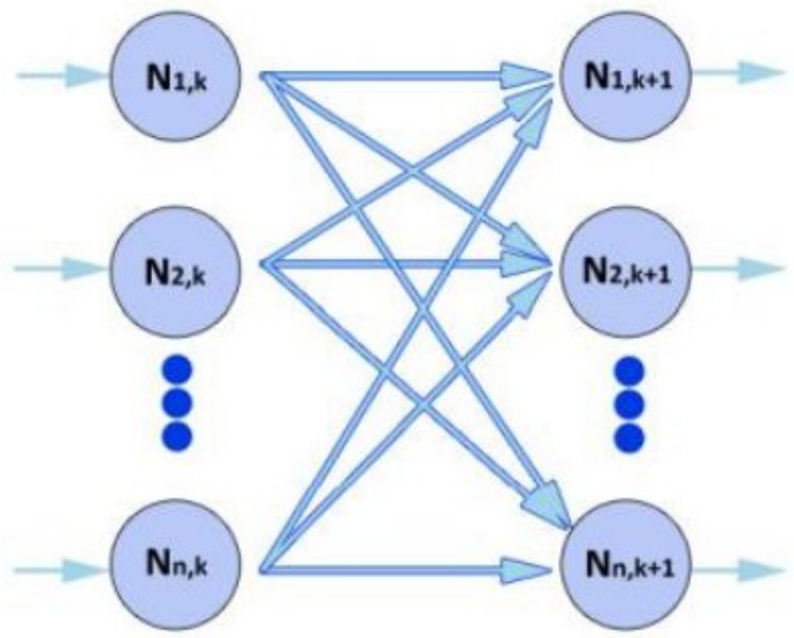

Figura 19. Fully Conected Layer
Fuente: Tomada de [85]

Las CNN están compuestas por capas de neuronas interconectadas que procesan los datos de entrada a través de una serie de operaciones matemáticas. Las CNN utilizan capas de filtros de características que se aplican de forma descentralizada a los datos de entrada para extraer características relevantes. Estas características se utilizan luego para realizar tareas de clasificación o para realizar predicciones. [86]

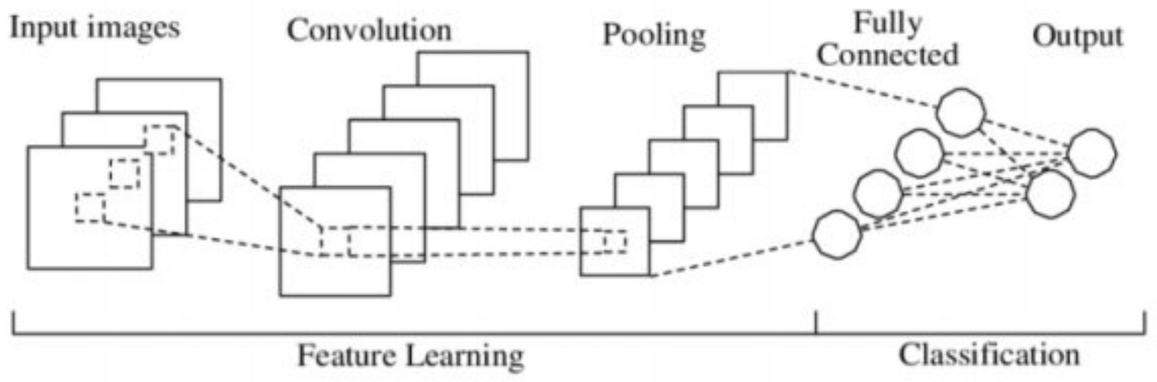

Figura 20. Diagrama de una red neuronal convolucional.
Fuente: Tomada de [86]

Según [87], "Las redes neuronales convolucionales es un algoritmo de *Deep Learning* que está diseñado para trabajar con imágenes, tomando estas como input, asignándole importancias (pesos) a ciertos elementos en la imagen para así poder diferenciar unos de otros".

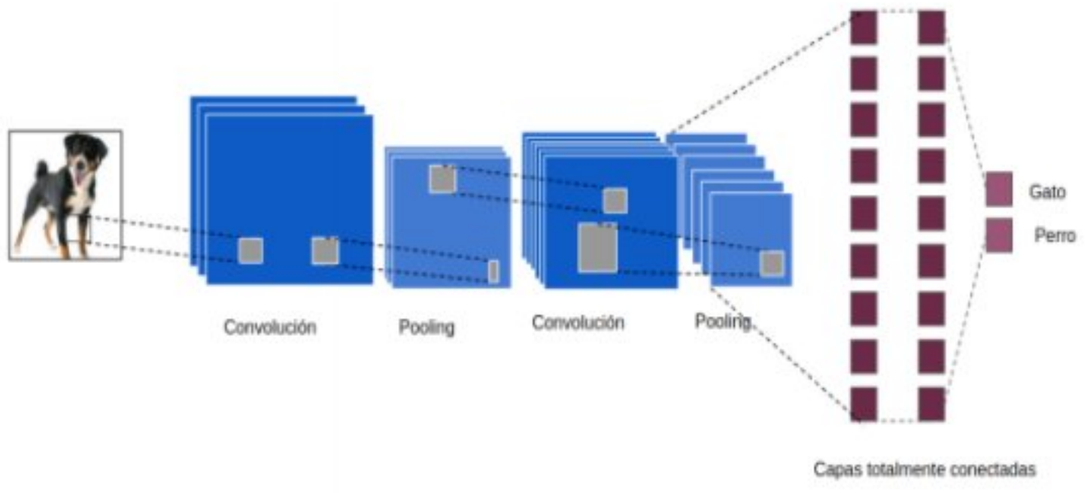

Figura 21. Diagrama de una red neuronal convolucional identificando un animal

Fuente: Tomada de [87]

En la actualidad existen diferentes investigaciones referentes a la detección facial las cuales han implementado redes neuronales [87]. En el 2015, por ejemplo, el trabajo presentado por H. Liet., propone una serie de CNN en forma de cascada para detectar rostros directamente de la propia imagen, la red neuronal aprende automáticamente las características en las que se tenía que basar. Concretamente, en su trabajo describen seis CNN, tres de las cuales son de detección y tres de calibración, estando intercaladas unas con otras. Las redes de detección, especializadas en 12, 24 y 48 píxeles de resolución respectivamente, eran utilizadas para localizar posibles caras en la imagen. Luego, la correspondiente red de calibración se encargaba de procesar las ventanas detectadas en las imágenes para ajustar su tamaño y localización para acercarse a una cara potencial cercana. En definitiva, se encargaba de calibrar los recuadros que rodeaban las caras para ajustarlos a la cara que es posible que puedan contener.

De esta forma, la cascada iba acercándose a las localizaciones de los rostros en la imagen hasta detectarlas por completo. Este proceso se ve representado en la Figura 22.

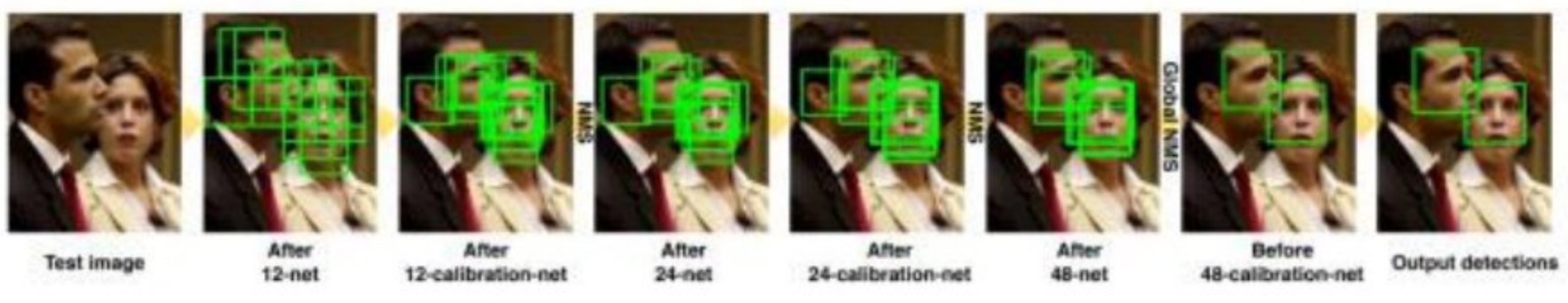

Figura 22. Localización de rostros

Fuente: Tomada de [87]

1.6.8.3 Detección usando el método de los HOG

El método de los histogramas de gradientes orientados (OHG, por sus siglas en inglés) es un algoritmo utilizado en visión por computadora para detectar y seguir objetos en una imagen.

Este método se basa en la idea de construir histogramas de gradientes orientados en diferentes regiones de la imagen para representar la orientación de los bordes en esas regiones. Estos histogramas se utilizan luego para identificar y seguir los objetos en la imagen.

En general, el método de los histogramas de gradientes orientados es un enfoque utilizado en la detección de objetos en imágenes que se basa en la detección y seguimiento de bordes en la imagen. Este método se utiliza en aplicaciones como la detección de caras en imágenes, el seguimiento de objetos en vídeos, y la navegación autónoma de vehículos. Algunas de las ventajas del método de los histogramas de gradientes orientados son que es rápido y eficiente en términos de recursos computacionales, y que puede manejar imágenes de alta resolución de manera efectiva.

Un ejemplo de cómo se puede utilizar el método de los histogramas de gradientes orientados es en la detección de caras en imágenes. Para hacer esto, primero se aplica un filtro de detección de bordes a la imagen para resaltar los bordes de los objetos en la imagen. Luego, se divide la imagen en diferentes regiones y se calculan los histogramas de gradientes orientados en cada una de esas regiones. Estos histogramas se utilizan luego para comparar con un modelo de cara previamente entrenado e identificar si hay una cara en esa región de la imagen. Si se encuentra una cara, se puede utilizar el método de los histogramas de gradientes orientados para seguir el movimiento de la cara en la imagen y rastrear su posición a medida que cambia en el tiempo.

Existen algoritmos que no han sido desarrollados específicamente para la detección de rostros, sino más bien, para la detección de objetos en general. Mas, sin embargo, presentan un buen comportamiento si se ajustan al problema de la detección de rostros. El algoritmo de los histogramas de gradientes orientados (HOG) es uno de ellos, fue presentado en 2015 por [16] y que, a pesar de que lo utilizaron para la detección de peatones en imágenes, es un método que puede ser utilizado para la detección de cualquier objeto, según como se modele.

En comparación con otros métodos de descripción de funciones, HOG tiene algunas ventajas que se pueden destacar, debido a que HOG opera en las celdas de la cuadrícula local de la imagen, lo que permite que pueda mantener una buena invariancia a las deformaciones geométricas y ópticas de la imagen; Estos dos tipos de deformaciones solo aparecerán en un espacio mayor. Otra de las ventajas es que, en las condiciones de muestreo espacial aproximado, muestreo de dirección fina y normalización óptica local fuerte, siempre que el peatón pueda mantener una postura erguida en general, puede permitir que los peatones tengan

algunos movimientos corporales sutiles. Estos movimientos sutiles pueden ser ignorados sin afectar el efecto de detección. Por lo tanto, la función HOG es particularmente adecuada para la detección humana en imágenes. Su funcionamiento se basa en dividir la imagen en varias celdas, dentro de las cuales, y analizando la intensidad de los píxeles, se obtiene su respectivo descriptor HOG (Figura 23)

Figura 23. Aplicación de los descriptores HOG. Izquierda: Imagen bajo test dividida en varias celdas. Derecha: Visualización de los descriptores HOG sobre la imagen.

Fuente: Tomada de [85]

Preprocesado

La fase de preprocesado tiene como objetivo normalizar y alinear los rostros detectados para una correcta extracción de características. Se lleva a cabo a través de transformaciones geométricas sobre la imagen. El preprocesado incluye las siguientes operaciones:

Rotación: El preprocesado incluye la normalización de los rostros detectados mediante la rotación de los mismos para alinearlos y facilitar la tarea del clasificador. Esto se puede hacer utilizando como referencia la línea de los ojos. De esta manera, se pueden aplicar transformaciones geométricas a la imagen para prepararla para la correcta extracción de características.

Escalado. Los algoritmos de escalado permiten cambiar el tamaño de la imagen, ya sea aumentando o disminuyendo su tamaño. Esto es útil para que todos los rostros tengan las mismas proporciones. La distancia entre los ojos se utiliza para calcular la tasa de aumento o disminución del tamaño de la imagen.

Recorte: La operación de recorte sirve para seleccionar la región de la imagen que contiene un rostro. Para hacer esto, se selecciona la submatriz de píxeles que delimitan al rostro en la imagen, que es una matriz de píxeles. Esta operación se realiza para facilitar la tarea del clasificador.

Ecualización del histograma: La fase de preprocesado incluye las siguientes operaciones: normalizar y alinear los rostros mediante la rotación utilizando la línea de los ojos como referencia; escalar los rostros para tener las mismas proporciones utilizando la distancia entre ojos para calcular la ratio de aumento o disminución del tamaño de la imagen; recortar la imagen para seleccionar solo la región que contiene el rostro; y aplicar ecualización a los histogramas para mejorar el contraste y la diferenciación de los rasgos.

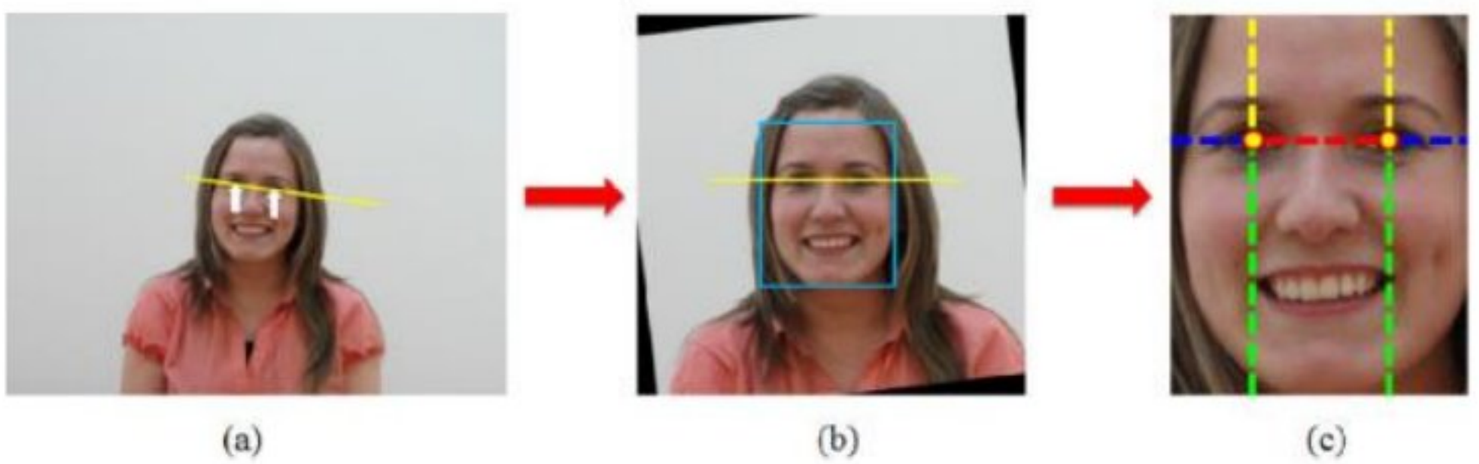

Figura 24. Operación de alineación y recorte de un rostro utilizando como referencia la línea de los ojos

Fuente:Autor

Extracción de características

La extracción de características consiste en la obtención de información relevante de un rostro mediante la reducción de redundancias y características irrelevantes. Existen una amplia variedad de algoritmos para llevar a cabo la extracción de características, pero en su mayoría, pueden ser clasificados a través de dos grandes aproximaciones.

Basados en los rasgos geométricos: En este método, se utilizan conjuntos de características biométricas como la distancia entre los ojos, el tamaño de la boca, la anchura de la nariz, etc. La ventaja de este método es su robustez frente a los cambios de expresión y orientación del rostro, así como que es menos afectado por los cambios en la iluminación. Sin embargo, esto conlleva el requerimiento de conocimiento previo de la imagen y puede ser más lento y complejo en comparación con los métodos basados en apariencia.

Holísticos o basados en la apariencia: Los métodos holísticos o basados en la apariencia se

enfocan en las propiedades globales del rostro. Estos métodos aproximan el problema de reconocimiento como un problema estadístico en lugar de analítico. La principal ventaja de este método es su rapidez y aplicabilidad a imágenes de baja resolución. Sin embargo, un inconveniente es la necesidad de tomar un mayor número de muestras en comparación con los métodos basados en rasgos geométricos. Además, estos métodos son sensibles a las expresiones y a la iluminación del rostro.

Los métodos de extracción de características basados en comparación de plantillas consisten en comparar características específicas de una imagen con plantillas previamente establecidas. Los métodos basados en redes neuronales utilizan redes neuronales para extraer características de las imágenes de entrada. Estos métodos se pueden incluir en el mapa conceptual mostrado en la Figura 21 [48].

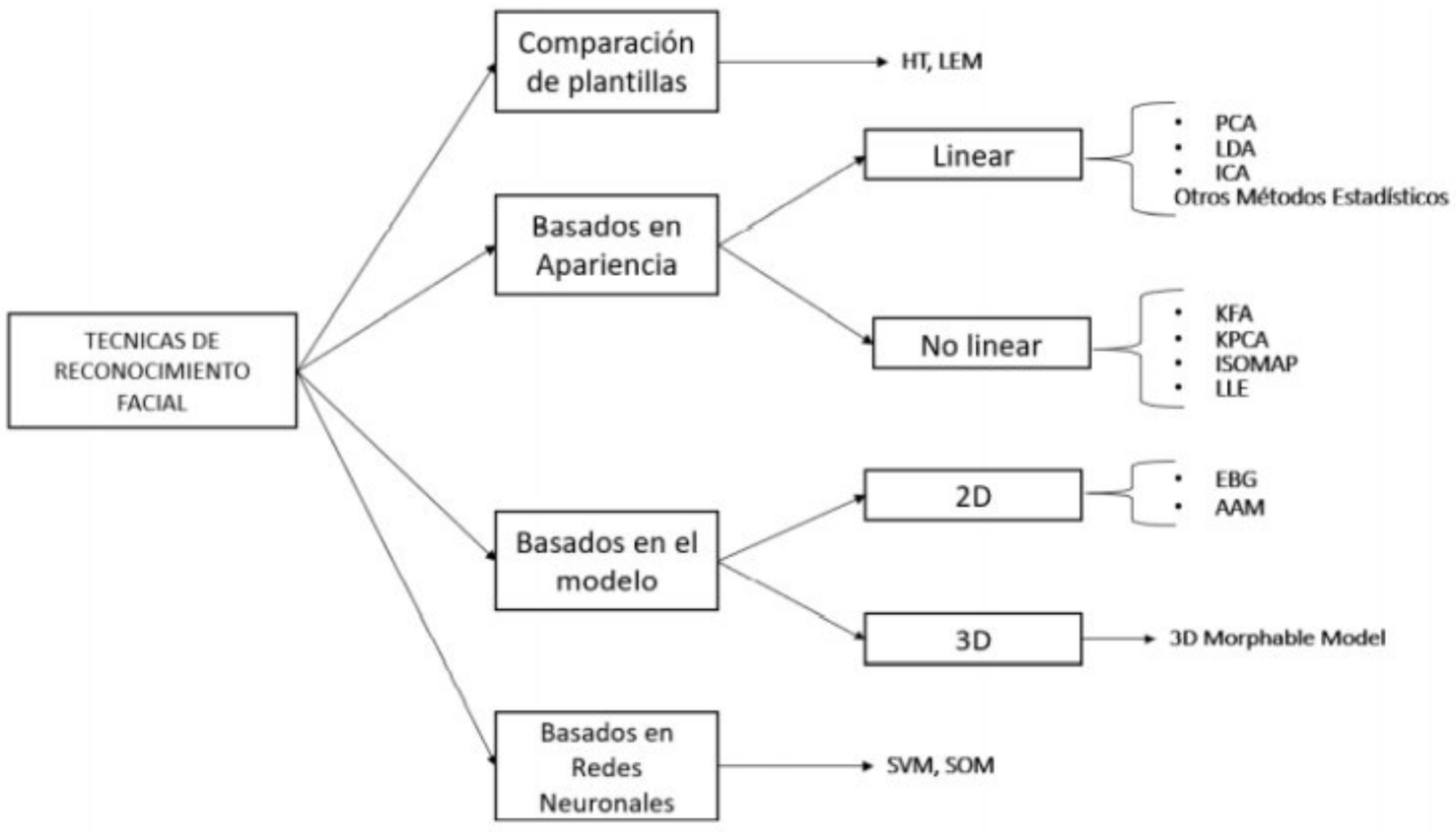

Figura 25. Clasificación de Técnicas de Extracción de Características

Fuente: Tomada de [48]

A modo de introducción, se presentarán dos técnicas de extracción de características: una basada en rasgos geométricos y una basada en apariencia.

Local Binary Patterns

Los Patrones Locales Binarios (Local Binary Patterns - LBP) es un descriptor visual simple pero muy eficiente utilizado para la clasificación de texturas. LBP puede ser visto como un enfoque que combina la estadística tradicional y el análisis de modelos estructurales de textura. Su principal propiedad es que este descriptor es invariante a los cambios en los niveles de color en escala de grises debidos a cambios en la iluminación, lo que lo convierte en un descriptor ideal para aplicaciones en el mundo real. Debido a que es un descriptor simple, puede procesar

imágenes en tiempo real [88].

El descriptor original de LBP fue presentado por [89]. En este trabajo, el descriptor se basa en una vecindad cercana a un pixel central de 3x3, y el valor central se utiliza como un umbral.

LBP suele trabajar en vecindades más grandes de píxeles (por ejemplo, regiones de 8x8 pixeles, 16x16 píxeles, etc.). Dentro de cada una de estas vecindades, se realiza una comparación de los píxeles vecinos al píxel central con diferentes radios (Figura 22). Si el valor del píxel central es mayor, se indica un valor de 0, y si es menor, se indica un valor de 1. Esto genera un número binario que se convierte a un valor decimal por conveniencia. Luego, se calcula el histograma de la frecuencia de estos valores, obteniendo así el vector característico de la región de píxeles. El vector característico de toda la imagen se obtiene a través de la unión de todos los vectores de las regiones.

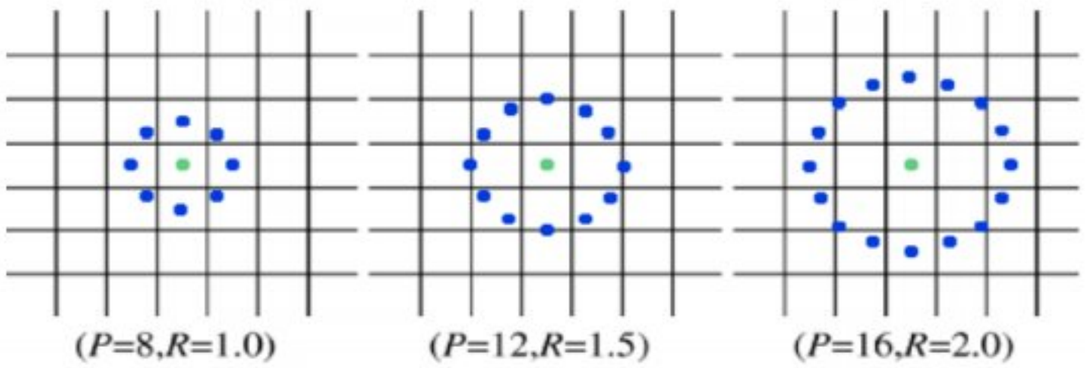

Figura 26. Ejemplos de vecindad de pixeles en LBP, se muestra el número de pixeles (P) y el radio utilizado (R)
Fuente: Tomada de [89].

Bloques LBP Multi-escala

Una variante de LBP fue propuesta por Laio et al. [90] que consiste en crear regiones más grandes de pixeles y promediar sus valores para comparar su valor con el promedio central de manera similar al LBP tradicional. Este pequeño cambio en la forma de generar el vector característico de LBP es útil para usarlo en conjunto con las Cascadas de Haar, ya que las diferencias entre estas regiones se utilizan de manera similar a los kernels de convolución y con la ventaja de ser más rápidos en términos de tiempo de procesamiento. En la Figura 11 se muestra una comparación entre estos dos enfoques. Por ejemplo, al calcular el LBP de una región, se genera un valor en binario (que representa el valor de la región en entero). Al obtener todas las representaciones binarias, se procede de la misma forma en la imagen integral, pero teniendo en cuenta que el tamaño de la imagen se reduce, lo que reduce el tiempo de procesamiento posterior en las cascadas de Haar.

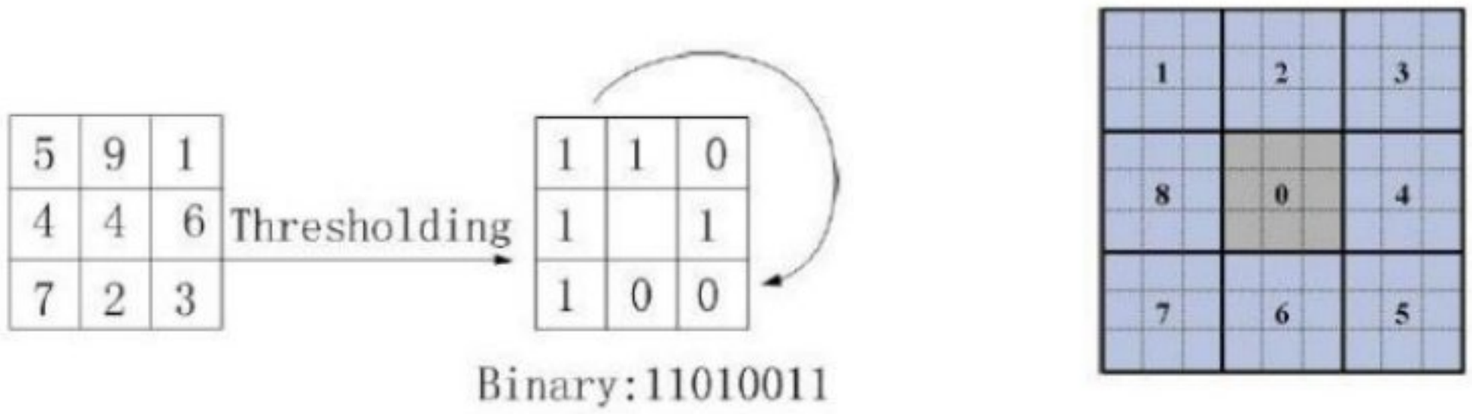

Figura 27. Ejemplo de LBP Multi-escala, en la izquierda se muestra el LBP tradicional, en la derecha la variación para bloques LBP Multi-escala

Fuente: Tomada de [90].

1.6.8.3.1 Principal Component Analysis

El método *Eigenfaces* [91] es uno de los más populares, construido sobre técnicas de Principal Component Analysis (PCA) [92]. Dicho método transforma la imagen a un subespacio (ejemplo en Figura 24) por el cual es posible obtener vectores de características de menor dimensionalidad sin una pérdida de información discriminativa importante.

Figura 28. Eigenfaces de un conjunto de imágenes de la base Extended Yale Face Database B

Fuente: Tomada de [89].

El análisis de componentes principales es una técnica de análisis de datos utilizada para reducir la dimensionalidad de un conjunto de datos. Se basa en identificar patrones en los datos y eliminar redundancias para encontrar un conjunto de vectores que sean lo más informativos posible del conjunto de datos original. Es comúnmente utilizado en minería de datos y aprendizaje automático para manejar conjuntos de datos grandes y complejos, reduciendo su dimensionalidad para hacerlos más fáciles de manipular y analizar. También se utiliza en aplicaciones de análisis de imagen y visión por computadora, donde se pueden tener imágenes de alta resolución con mucha información redundante.[92]

El método denominado *Eigenfaces*, está basado en PCA [93] el cual se fundamenta en la creación de un espacio en donde las imágenes únicamente son representadas en base a las características que son más relevantes.

Figura 29. Eigenfaces de un conjunto de imágenes de la base Extended Yale Face Database B.

Fuente: Tomada de [93]

1.6.8.3.2 Eigenfaces (PCA)

Eigenfaces, es un algoritmo de análisis de componentes principales que se utiliza en el reconocimiento de patrones, especialmente en el reconocimiento de rostros. Se basa en la idea de que una imagen de rostro puede representarse como una combinación lineal de un número limitado de "rostros base", llamados "*eigenfaces*". Esta técnica ha demostrado ser eficiente y precisa en el reconocimiento de rostros en imágenes. [94].

Figura 30. Imágenes de entrenamiento

Fuente: Tomada de [94].

Eigenfaces no trabaja directamente con imágenes, sino que primero las convierte en matrices (vectores). Es decir, una imagen de n x n píxeles (cada píxel tiene un valor entre 0 y 255) se convierte en un vector de n2 x 1. Se calcula un promedio a partir de todos los vectores, llamado

"vector promedio del rostro". Cada vector se resta de este promedio para obtener los vectores normalizados. Se calcula la matriz de covarianza para calcular los eigenvectores, cuya matriz tiene una dimensión de n2 x n2. Este cálculo requiere mucha memoria para almacenar la matriz. La solución es aplicar una SVD (descomposición en valores singulares), lo que reduce la dimensión de la matriz de n2 x n2 a M x M, donde M es el número de imágenes de entrenamiento. La SVD descompone una matriz en un producto de tres matrices, una de las cuales contiene los eigenvectores y otra los eigenvalues. [95].

En este proceso, se seleccionan los n vectores con los valores propios más grandes. Luego se calculan los vectores propios de la matriz de covarianza para encontrar las *eigenfaces*. Una nueva imagen de rostro se forma con las *eigenfaces* [95]. Se resta el vector promedio del rostro de la nueva imagen y se multiplica con cada vector donde se encontraron las *eigenfaces*. El objetivo es determinar cuál de las imágenes del conjunto de entrenamiento se parece más a la imagen de entrada, utilizando una fórmula conocida como distancia Euclidiana. Aquella imagen que tenga la menor distancia se considera como el rostro de la imagen de entrada.

Figura 31. Conjunto de Eigenfaces

Fuente: Tomada de [95].

De aquí se desprende que los valores de las distancias dependen en cierta medida del tamaño de la base de datos, puesto que la matriz de covarianza y los vectores propios son calculados a partir de la matriz formada por la imagen de entrada y las ya almacenadas.

Comparación y clasificación

Los métodos hasta ahora presentados en este trabajo han permitido localizar rostros en imágenes, normalizarlos y codificarlos para representar sus características como un conjunto

de componentes de un espacio vectorial. El objetivo de esta fase final es comparar dicho vector de características con aquellos almacenados en la base de datos del sistema, buscando similitudes y diferencias para establecer si el rostro en análisis pertenece al conjunto de rostros ya previamente analizados, o si por el contrario se trata de un rostro nuevo desconocido. En este apartado se exponen algunas de las diferentes técnicas empleadas en el problema de clasificación de imágenes.

Distancia Euclídea

La distancia euclídea no es un método de clasificación en sí misma, sino que es la herramienta básica que se utiliza para determinar la similitud entre muestras. Haciendo uso de el Teorema de Pitágoras se puede generalizar que la distancia euclídea de dos vectores X = [x1 , x2 ,...,xN] e Y = [y1 , y2 ,...,yN] pertenecientes a un espacio de N dimensiones es:

$$d(X,Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_N - y_N)^2} \quad \text{(X1)}$$

1.6.8.4 Random Forest

Los Bosques Aleatorios o Random Forest (FR), es un algoritmo de aprendizaje automático que construye un conjunto de árboles de decisión y utiliza la media de sus predicciones para mejorar la precisión y controlar el sobreajuste. Este algoritmo es muy utilizado en problemas de clasificación y regresión.[96]. Para construir los árboles de decisión en Random Forest [96], se comienza creando un conjunto de datos de entrenamiento a partir de un proceso de muestreo llamado bootstrapping. Luego, se generan árboles de decisión utilizando pares de variables del vector de características. Finalmente, se utilizan los datos que no se han utilizado en el proceso de entrenamiento para validar el modelo utilizando el conjunto de datos "out-of-bag"

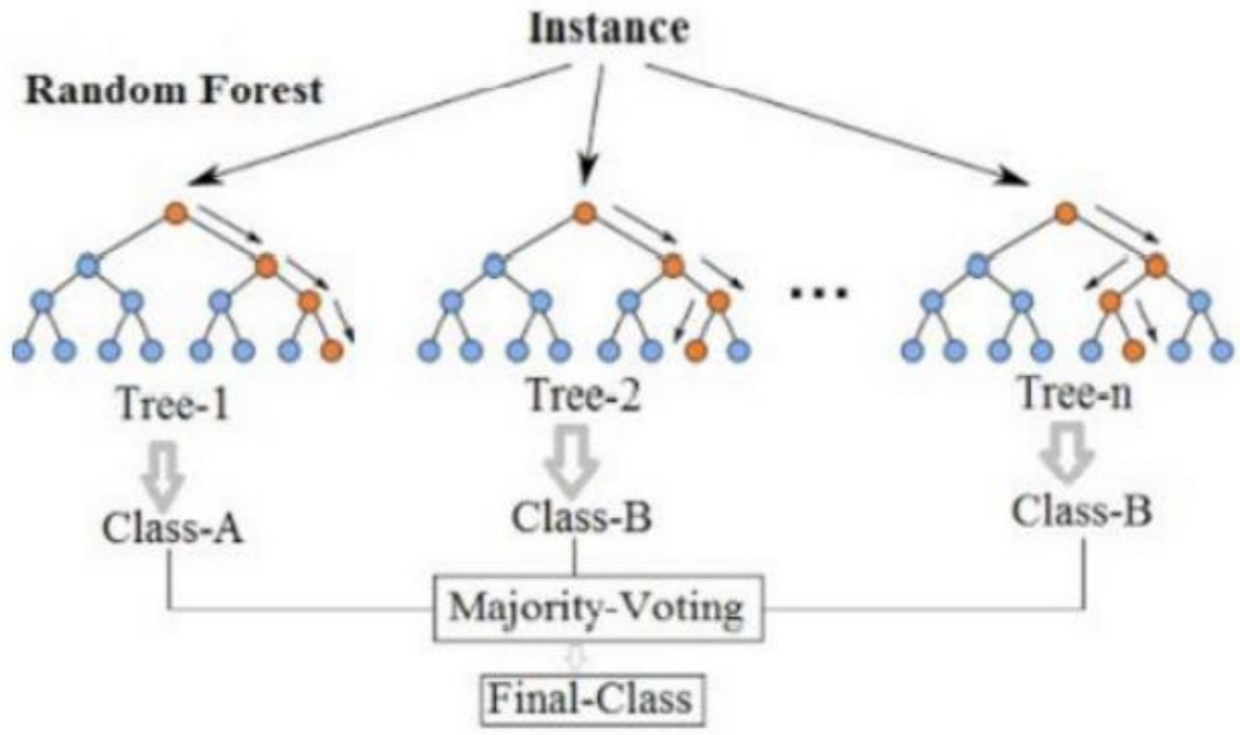

Figura 32. Diagrama Random Forest

Fuente: Tomada de [96].

1.6.8.5 K-Nearest Neighbours

K-Nearest Neighbors (K-NN) es un algoritmo de clasificación y regresión basado en la idea de tomar la decisión en función de la mayoría de los vecinos más cercanos. Es decir, para predecir la clase de una nueva instancia, el algoritmo selecciona a las K instancias más similares (en términos de distancias) y la clase predicha es la que más se repite entre los vecinos más cercanos [97][98].

El funcionamiento del algoritmo es el siguiente:

1. Determinar el número de vecinos más cercanos (K) que se usarán para hacer la clasificación.
2. Calcula la distancia entre la muestra a clasificar y el resto de las muestras del conjunto de datos.
3. Selecciona a los K vecinos más cercanos.
4. Asigna la clase de la muestra a clasificar basándose en la mayoría de las clases de los vecinos seleccionados.

Este algoritmo tiene varias ventajas, como su simplicidad y su capacidad para manejar datos de entrenamiento con ruido. También es efectivo con conjuntos de entrenamiento muy grandes. Sin embargo, también tiene algunos problemas, como la necesidad de determinar el número óptimo de vecinos (K) como se muestra en la figura 29. Estos cálculos tienen un alto coste computacional, ya que cada muestra de entrada debe compararse con cada muestra del conjunto de datos observado.

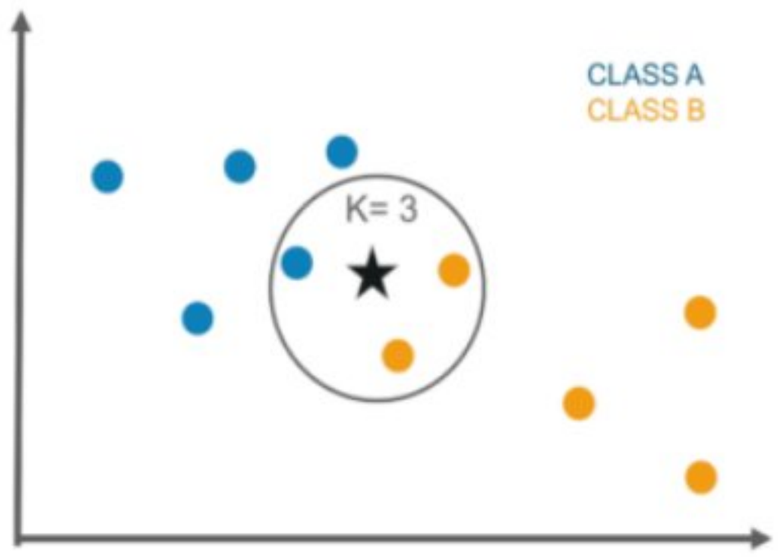

Figura 33. Para K=3, la muestra se clasifica como clase B, pero para K=5 como clase A
Fuente: Tomada de [97][98].

1.6.8.6 Support Vector Machines

Support Vector Machines (SVM) es un algoritmo de aprendizaje automático utilizado para clasificación y regresión. Funciona buscando un hiperplano de separación óptimo en un espacio dimensional muy alto (siendo el número de dimensiones igual al número de características de los datos de entrada). Los puntos más cercanos al hiperplano se conocen como vectores de soporte y son utilizados para definir el hiperplano [99].

1.6.8.7 Redes Neuronales

Las redes neuronales son un tipo de algoritmo de aprendizaje automático inspirado en el funcionamiento del cerebro humano. Se componen de capas de neuronas interconectadas que procesan y transmiten información. Las redes neuronales se utilizan a menudo para clasificación y regresión, así como para la detección de patrones y la toma de decisiones [100]. Se utilizan comúnmente para la clasificación de datos. La clasificación es una tarea de aprendizaje automático que consiste en asignar etiquetas o clases a un conjunto de datos, en función de sus características o atributos. Por ejemplo, una red neuronal puede utilizarse para clasificar imágenes de perros y gatos, basándose en las características de las imágenes (tamaño, forma del rostro, patrón de pelaje, etc.) [100]

Para realizar la clasificación, la red neuronal se entrena con un conjunto de datos etiquetados que incluyen imágenes y las etiquetas correspondientes (perro o gato). A partir de este entrenamiento, la red neuronal aprende a asignar la etiqueta correcta a nuevas imágenes que se le presentan. Esto se logra a través del ajuste de los pesos y sesgos de las conexiones entre las neuronas de la red, de modo que se pueda realizar una predicción precisa para cada nueva

imagen. [101]

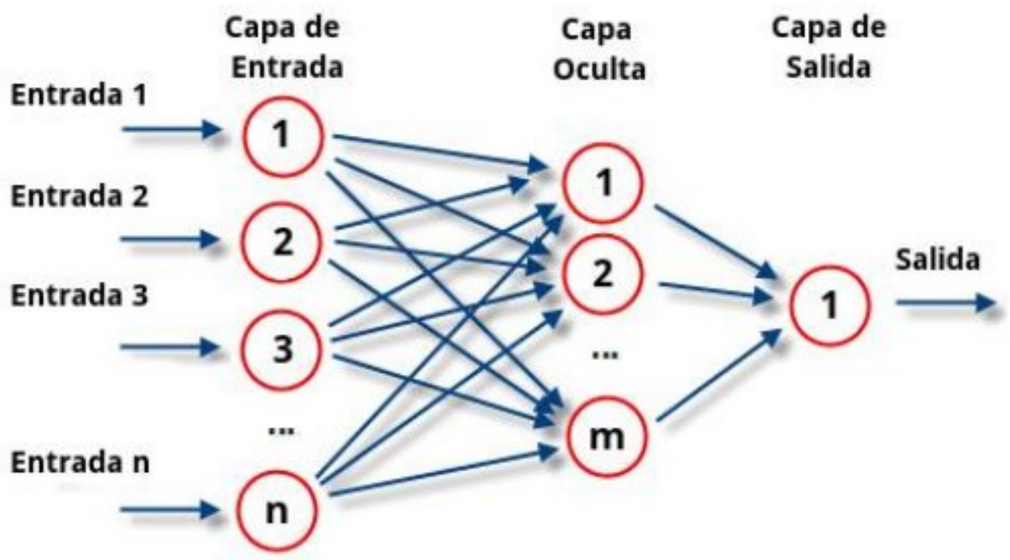

Figura 34. Red neuronal con una capa oculta

Fuente: Tomada de [100].

Una vez entrenada, la red neuronal puede utilizarse para clasificar nuevas imágenes de perros y gatos, simplemente procesando la imagen a través de la red y utilizando la etiqueta asignada por la red como resultado de la clasificación. Las redes neuronales se utilizan comúnmente para la clasificación de datos en diversas aplicaciones, incluyendo el reconocimiento de voz, el procesamiento del lenguaje natural y el reconocimiento facial.

1.6.8.8 Deep Learning

El aprendizaje profundo o *Deep Learning* es un subcampo del aprendizaje automático o *Machine Learning*, que a su vez, es un subcampo de la IA [102]. *Deep learning* es una técnica de aprendizaje automático que se basa en el uso de redes neuronales profundas (*deep neural networks*) para aprender a partir de datos. Las redes neuronales profundas son un tipo de modelo de aprendizaje automático que se inspira en la estructura y el funcionamiento del cerebro humano. Estas redes están compuestas por muchas capas de neuronas interconectadas, lo que les permite procesar y analizar grandes cantidades de datos de manera automática [102]. El *deep learning* se ha utilizado con éxito en una amplia variedad de aplicaciones, incluyendo el reconocimiento de voz, el procesamiento del lenguaje natural, el reconocimiento facial y la traducción automática. También ha sido utilizado para mejorar la precisión y la eficiencia de los sistemas de detección de spam, la identificación de anomalías en señales de sensores y la detección de patrones en grandes conjuntos de datos.

En general, el *deep learning* se considera una técnica de vanguardia en el campo del aprendizaje automático y ha demostrado tener un gran potencial para resolver problemas complejos que involucran el procesamiento y análisis de grandes cantidades de datos [103][104].

El principal objetivo de la IA es ofrecer algoritmos y técnicas para resolver problemas que las personas pueden resolver de forma intuitiva o automatizada.

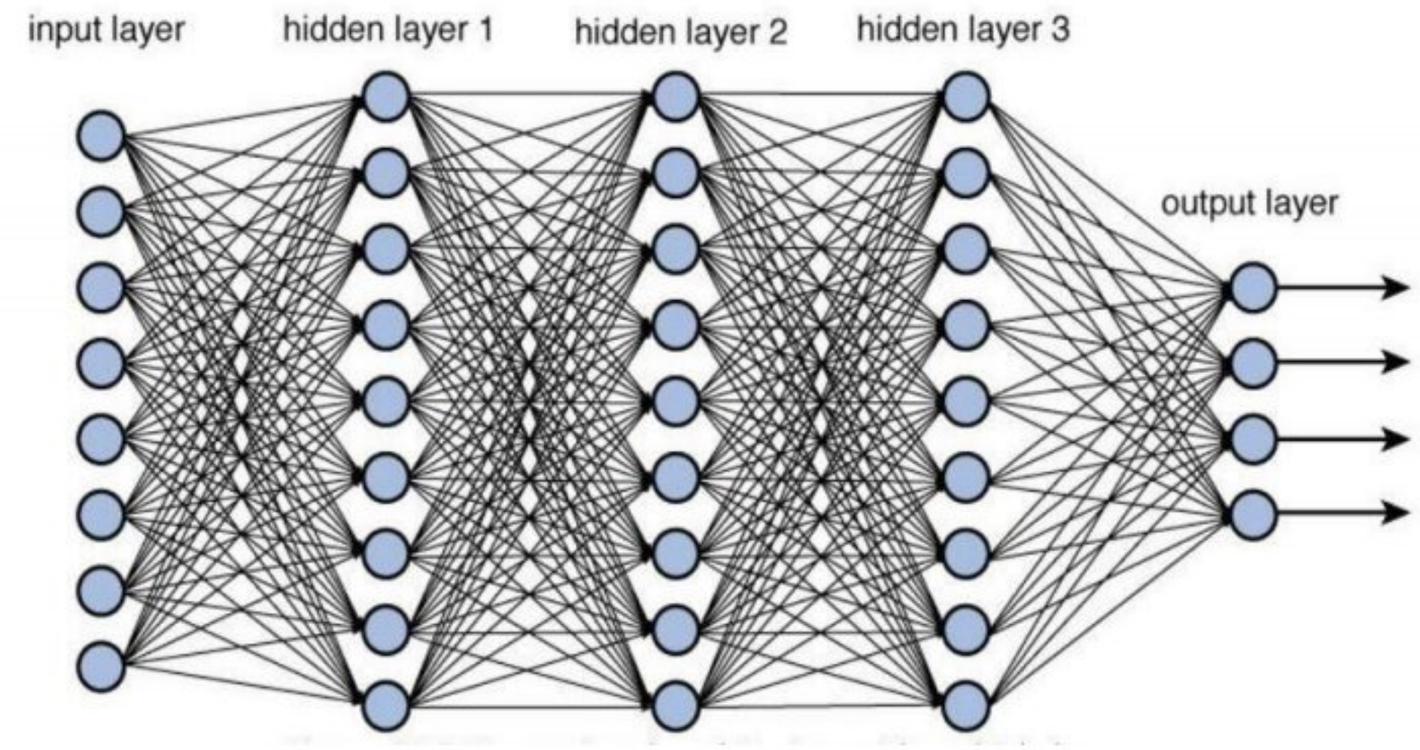

Figura 35. Red neuronal con varias capas ocultas.
Fuente: Tomada de [104].

1.6.8.9 Biblioteca Face-recognition

La biblioteca *Face-recognition* es una herramienta de código abierto desarrollada en el lenguaje de programación Python que permite la detección y reconocimiento de rostros en imágenes y videos. La biblioteca *Face-recognition* utiliza una técnica de reconocimiento facial basada en aprendizaje profundo llamada "*embedding*" [128]. Esta técnica consiste en entrenar una red neuronal convolucional (CNN) para aprender una representación de características altamente discriminativas de cada cara en una imagen. La red neuronal se entrena utilizando un conjunto de datos de imágenes etiquetadas con las caras de interés y, una vez entrenada, se puede utilizar para extraer las características faciales de una nueva imagen.

Esta biblioteca utiliza una técnica de aprendizaje profundo conocida como redes neuronales convolucionales (CNN por sus siglas en inglés), que es capaz de aprender y extraer características de las imágenes para identificar patrones y detectar rostros con alta precisión. Se basa en técnicas de aprendizaje automático [129], como el análisis de componentes principales (PCA) y las redes neuronales convolucionales (CNN), para realizar tareas de reconocimiento facial de alta precisión y eficiencia. Estos métodos se han mejorado significativamente gracias a la aplicación de técnicas de aprendizaje automático, como el aprendizaje profundo, que han permitido la creación de sistemas de reconocimiento facial altamente precisos y eficientes.

Ha sido ampliamente utilizada en diferentes proyectos de investigación, como en la identificación de emociones faciales [130], en la detección de la fatiga de conductores [131], la mejora de la accesibilidad de personas con discapacidades visuales [132], aplicaciones de seguridad, como el control de acceso en edificios, la vigilancia en aeropuertos y la identificación de sospechosos en investigaciones criminales.

En cuanto a la implementación de la biblioteca de reconocimiento facial, existen diversas opciones de software y hardware disponibles en el mercado.

1.6.8.10 Biblioteca *Dlib*

La biblioteca *Dlib* es una popular biblioteca de software libre para C++ que incluye una amplia variedad de algoritmos para el procesamiento de imágenes y el aprendizaje automático, incluyendo funciones para el reconocimiento facial [134]. Esta biblioteca ofrece una interfaz sencilla y fácil de usar, lo que la hace popular entre los desarrolladores de aplicaciones de reconocimiento facial. Entre las funciones de reconocimiento facial que ofrece *Dlib* se encuentra la detección y el seguimiento de rostros, la extracción de características faciales y la comparación de imágenes de rostros. La biblioteca también ofrece herramientas para la creación de modelos de aprendizaje automático personalizados para el reconocimiento facial [135]. Algunas de las aplicaciones que han utilizado la biblioteca *Dlib* incluyen la detección de rostros en fotografías, la identificación de sospechosos en imágenes de cámaras de vigilancia y la detección de emociones en imágenes faciales.

1.6.8.11 Biblioteca FaceNet

La biblioteca FaceNet es una herramienta de reconocimiento facial basada en redes neuronales profundas que permite la extracción de características faciales altamente discriminativas. Esta biblioteca utiliza una técnica llamada "*triplet loss*" que busca minimizar la distancia entre las características de imágenes de la misma persona y maximizar la distancia entre las características de imágenes de diferentes personas [136].

FaceNet ha logrado un alto rendimiento en varias tareas de reconocimiento facial, superando a muchos otros métodos tradicionales. Además, ha sido implementado en diversas aplicaciones de seguridad y de identificación de personas en tiempo real, tales como sistemas de acceso a edificios y vigilancia en tiempo real [137].

1.6.8.12 Framework CaffeModel

CaffeModel es una herramienta de aprendizaje profundo basada en el marco de trabajo *Caffe*, que permite la construcción y entrenamiento de modelos de redes neuronales profundas para una variedad de tareas de visión artificial, incluyendo el reconocimiento facial. *CaffeModel* proporciona una interfaz fácil de usar para cargar, modificar y entrenar modelos de redes neuronales pre entrenados para una tarea específica, o para crear modelos personalizados desde cero [d1]. Una de las ventajas de *CaffeModel* es su velocidad de entrenamiento, lo que lo hace ideal para aplicaciones de tiempo real. También es compatible con varias arquitecturas de redes neuronales profundas, incluyendo *Convolutional Neural Networks (CNNs) y Recurrent Neural Networks (RNNs)*. Además, cuenta con una amplia comunidad de desarrolladores que ofrecen soporte y recursos adicionales para el desarrollo de modelos de aprendizaje profundo [138].

1.7 Metodología CRISP DM

CRISP-DM (*Cross Industry Standard Process for Data Mining*) es un proceso estandarizado para el análisis de datos que se utiliza para abordar problemas de negocio. Este proceso se compone de seis fases [105]:

- **Comprensión del negocio:** En esta fase, se entienden los objetivos del negocio y se define el problema a resolver mediante el análisis de datos.
- **Comprensión de los datos:** En esta fase, se recolectan y analizan los datos necesarios para abordar el problema definido en la fase anterior.
- **Preparación de los datos:** En esta fase, se limpian, integran y transforman los datos para que estén listos para el análisis.
- **Modelado:** En esta fase, se utilizan técnicas estadísticas y de aprendizaje automático para construir modelos que pueden ser utilizados para abordar el problema de negocio.
- **Evaluación:** En esta fase, se evalúa el rendimiento del modelo construido en la fase anterior y se determina si el modelo es adecuado para abordar el problema de negocio.
- **Implementación:** En esta fase, se implementa el modelo seleccionado en un entorno de producción y se mide su rendimiento en condiciones reales.

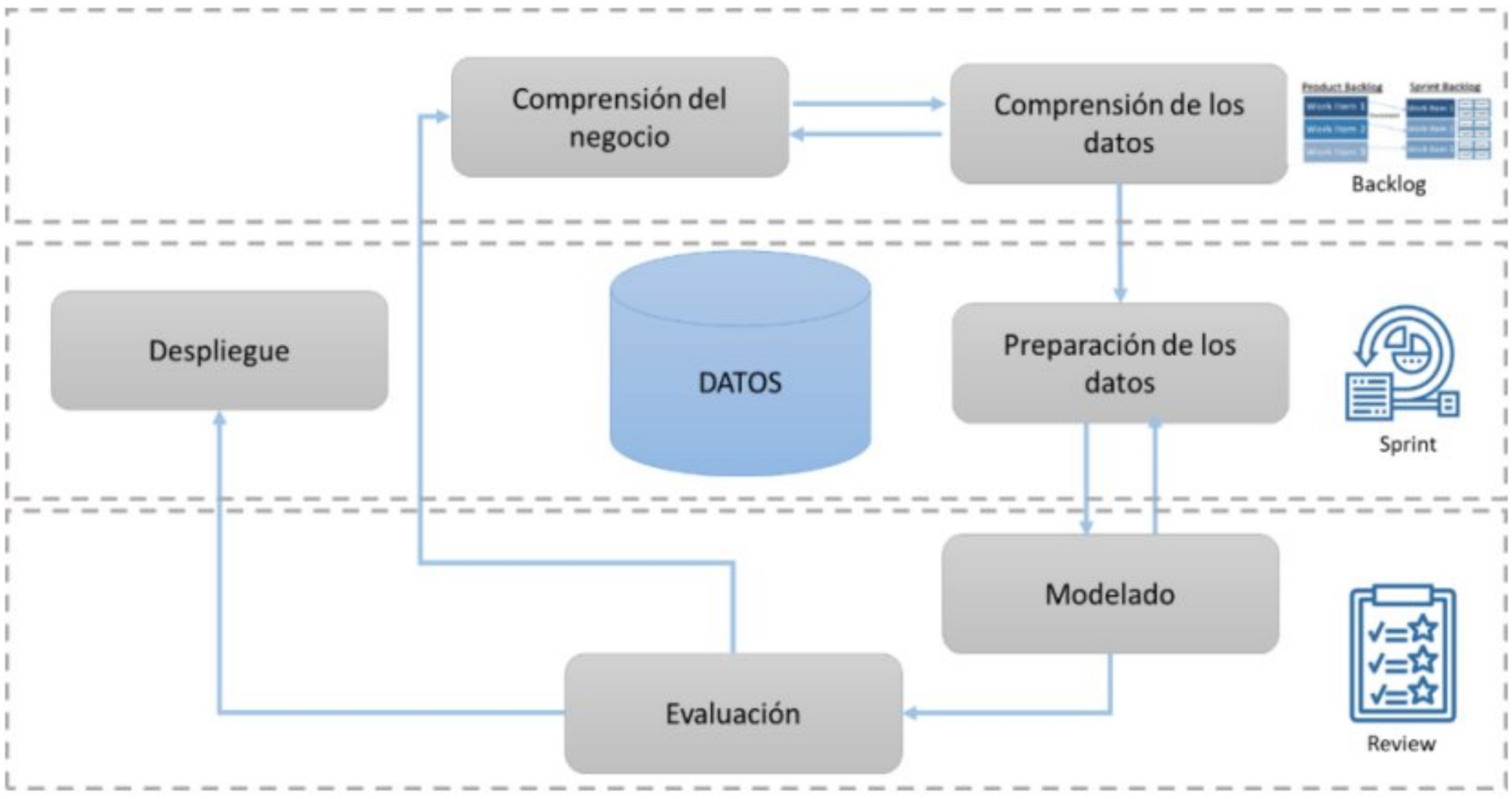

Figura 36. Metodología CRISP-DM.
Fuente: Tomada de [105]

Según [105], La metodología *CRISP-DM* es un proceso ampliamente utilizado en el mundo empresarial debido a su enfoque en el negocio y al proceso estandarizado que facilita la comunicación entre los diferentes departamentos y roles involucrados en un proyecto de análisis de datos. Además, permite una gestión adecuada del proyecto y una mejor comprensión del negocio.

CAPÍTULO II

2. Materiales y métodos

2.1. Diseño de la investigación

Según [110], existen dos tipos de investigación: investigación experimental e investigación no experimental. La investigación no experimental se divide en dos tipos: diseños transeccionales o transversales y diseños longitudinales. Los diseños experimentales son típicos de la investigación cuantitativa, mientras que los diseños no experimentales se pueden utilizar tanto en enfoques cuantitativos como cualitativos [110]

2.1.1 Diseño experimental

El diseño de la presente investigación es de tipo experimental ya que, los algoritmos seleccionados se someten a pruebas en base al caso de estudio, mediante prueba y error a través de la experimentación de cada técnica de detección reconocimiento de rostros, validando en cada prueba aspectos como: falsos positivos y falsos negativos.

Según [111], un diseño experimental es un enfoque de investigación que se utiliza para establecer causalidad entre variables. En el campo de la detección y reconocimiento facial, un diseño experimental se utiliza para determinar si al existir cambios en una variable, como el tamaño de la muestra de imágenes utilizadas para el entrenamiento, afectan a la precisión de un algoritmo de detección y reconocimiento facial.

Otro ejemplo sería un estudio en el que se compara el rendimiento de un algoritmo de detección y reconocimiento facial utilizando diferentes tamaños de muestra de imágenes para el entrenamiento [112].

2.3 Paradigma o Enfoque

El enfoque es el positivismo, porque los resultados en el proceso experimental de prueba y error, se evalúan en forma cuantitativa, empírico-analítico y científico tecnológico. Este enfoque sustentará los resultados de la presente investigación ya que, en cumplimiento a los objetivos de la misma permitirán comprobar la hipótesis inicial de la propuesta.

2.3.1 Enfoque cuantitativo

Se realiza este enfoque, para plasmar los resultados de cada comparación de la experimentación de cada métodos y técnicas en una matriz de confusión, ya que, según [113], "*En el campo de la inteligencia artificial y el aprendizaje automático, el uso de una matriz de confusión resulta una herramienta que permite visualizar el desempeño de un algoritmo de aprendizaje supervisado y se puede contrastar con el resultado esperado*".

Una matriz de confusión se utiliza a menudo en el reconocimiento facial para evaluar el rendimiento de un modelo de reconocimiento facial. La matriz de confusión es una tabla que muestra el número de veces que el modelo clasificó correctamente a cada persona (verdaderos positivos), el número de veces que el modelo clasificó incorrectamente a otra persona como la persona correcta (falsos positivos), el número de veces que el modelo no reconoció correctamente a la persona (falsos negativos) y el número de veces que el modelo no reconoció a nadie (verdaderos negativos) [114].

2.4. Cálculo de población y muestra

Por lo característica de la investigación se ha empleado un muestreo no probabilístico, la unidad de análisis se centra en los rostros de los estudiantes. El objetivo principal es evaluar el desempeño de los algoritmos de reconocimiento facial en la suplantación de identidad durante los exámenes en línea. Por lo tanto, el enfoque de la población y muestra se ajusta a esta perspectiva.

La población objetivo consiste en los rostros de los estudiantes que participan en los exámenes en línea. Sin embargo, dado que no es práctico analizar todos los rostros de los estudiantes de forma exhaustiva, se requerirá una muestra representativa para realizar los experimentos y evaluaciones pertinentes.

La muestra representativa seleccionada para este estudio estará compuesta por 10 estudiantes, los cuales serán elegidos utilizando un enfoque sistemático. Cada algoritmo se entrena con 500 imágenes de cada estudiante, obteniendo un dataSet de 5000 imágenes de entrenamiento.

Es importante destacar que, aunque el tamaño de la muestra será limitado, se busca que los resultados obtenidos sean indicativos del desempeño de los algoritmos de reconocimiento facial en la suplantación de identidad durante los exámenes en línea. Aunque no se pueda

generalizar de forma directa a toda la población de estudiantes, los resultados servirán como base para comprender y tomar decisiones en relación con la implementación de un modelo de reconocimiento facial utilizando inteligencia artificial en el contexto de la supervisión remota de exámenes en línea.

2.5 Técnicas estadísticas

En cuanto las técnicas estadísticas que se aplican en la investigación, se tienen los siguientes:

- **Medición de la precisión:** La precisión es una métrica utilizada para evaluar el rendimiento de un sistema de reconocimiento facial o de cualquier otro sistema de clasificación. Se refiere a la proporción de casos en los que el sistema ha clasificado correctamente a una persona o imagen en relación con el total de casos clasificados.

  La precisión mide la cantidad de veces que el sistema identifica correctamente a una persona. Es calculada como el número de veces que el sistema identifica correctamente a una persona dividido por el número total de veces que se realiza una identificación.

- **Medición del *recall*:** Se refiere a la proporción de casos en los que el sistema ha clasificado correctamente a una persona o imagen en relación con el total de casos que debería haber clasificado correctamente. El *recall* mide la cantidad de veces que el sistema detecta a todas las personas autorizadas. Es calculado como el número de veces que el sistema identifica correctamente a una persona dividido por el número total de veces que una persona está autorizada para ser identificada.

- **Valor de referencia:** El valor de referencia, también conocido como punto de corte o umbral de decisión, es un parámetro utilizado en el reconocimiento facial y en otros sistemas de clasificación para determinar cuándo una persona o imagen debe ser clasificada como "positiva" o "negativa". El valor de referencia se establece en un punto específico en el rango de salida del sistema, y todos los casos con un valor de salida superior a ese punto son clasificados como "positivos", mientras que todos los demás son clasificados como "negativos".

- **Matriz de confusión**: La matriz de confusión es una tabla que muestra las predicciones correctas y las predicciones incorrectas del sistema. Se utiliza para identificar las áreas en las que el sistema tiene dificultades para identificar correctamente a las personas.

- **Índices *Kappa*:** Son medidas estadísticas utilizadas para evaluar la concordancia o acuerdo entre dos o más observadores o evaluadores en una tarea de clasificación o medición. El coeficiente kappa mide la proporción de acuerdo que se logra después de eliminar el acuerdo que se produciría por azar. Un coeficiente kappa igual a 1 indica una concordancia perfecta, mientras que un coeficiente kappa igual a 0 indica una concordancia igual a la que se produciría por azar

Estas métricas son importantes para evaluar la precisión de un algoritmo ya que se pueden calcular varias medidas de rendimiento como la precisión, el *recall* y el valor de referencia.

2.6 Métodos teóricos

2.6.1 Método Inductivo

El método inductivo en investigación es un proceso de razonamiento que comienza con datos específicos y generaliza a partir de ellos para llegar a una conclusión más amplia. En el contexto de reconocimiento facial, el método inductivo podría utilizarse para analizar datos de imágenes faciales y extraer patrones y características que se utilizan para identificar a una persona. La investigación se basa en este método ya que el objetivo es crear un prototipo de reconocimiento facial que se ajuste y mejore con el tiempo a medida que se alimenta con más datos.

2.7 Metodología CRISP-DM

Para garantizar que el proyecto se lleve a cabo de manera estructurada y eficientemente se plantea utilizar CRISP-DM como metodología de desarrollo, ya que es ampliamente utilizada para llevar a cabo proyectos de minería y análisis de datos, además de su enfoque estructurado y su énfasis en la comunicación entre los diferentes roles involucrados en el proyecto. Así mismo en un estudio realizado por Piatetsky [115][116], en donde realiza una comparación de metodologías para proyectos enfocados en análisis de datos y aprendizaje automático, concluye que CRISP-DM es comúnmente más usado en proyectos de minería de datos de gran volumen y considera que es una metodología útil ya que proporciona un marco estructurado para planificar ejecutar y evaluar proyectos de análisis de datos.

Como se mencionó en el capítulo anterior, la metodología CRISP-DM consta de seis fases, en este capítulo se describe la adaptación de cada fase con la investigación.

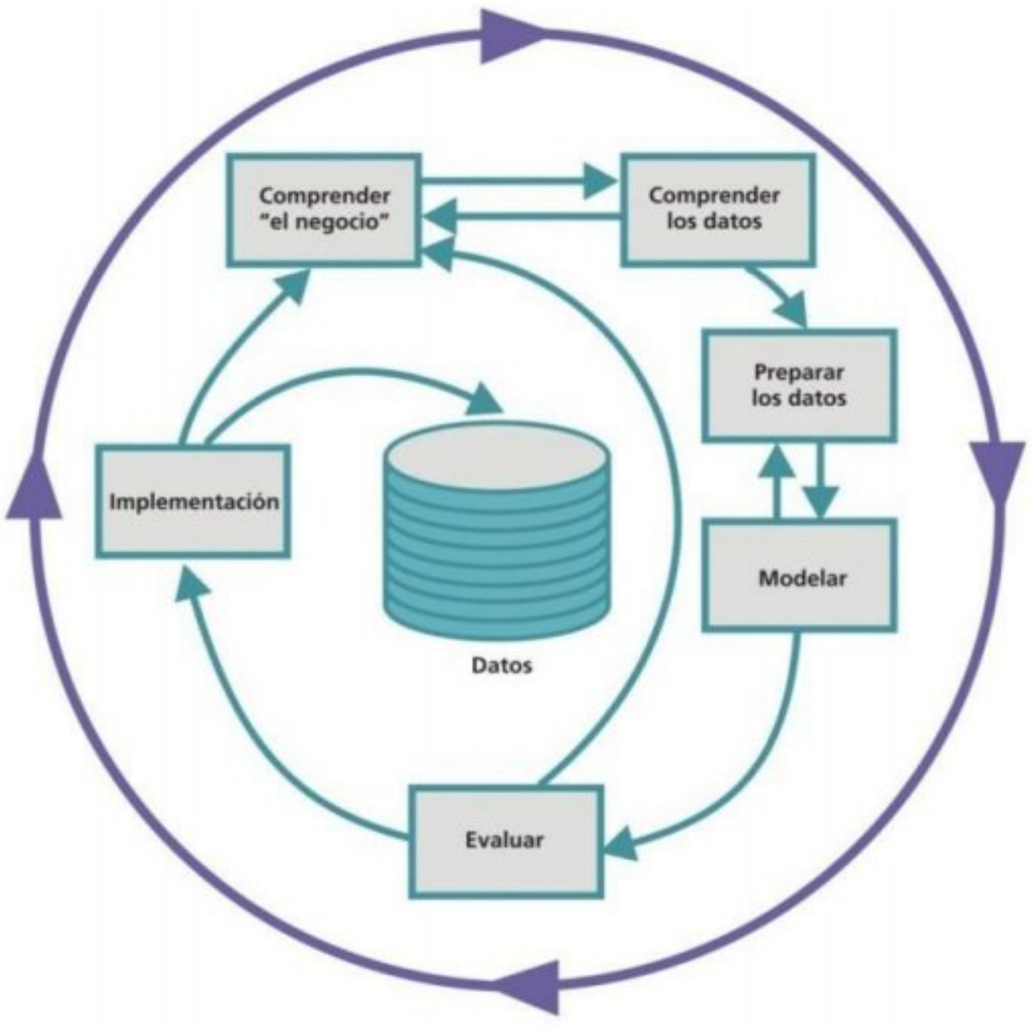

Figura 37. Fases metodología CRISP-DM

Fuente: Tomada de [115][116]

2.7.1 Fase de comprensión del negocio

2.7.1.2 Situación Actual

En el capítulo introductorio, se menciona sobre el problema de la suplantación de identidad en exámenes en línea y como esto es un problema grave que puede afectar la integridad de los exámenes y la confiabilidad de los resultados, así mismo, a través de la hipótesis se describe de como un sistema de reconocimiento facial puede ayudar a prevenir la suplantación de identidad.

2.7.1.3 Objetivos

Desarrollar un prototipo que sea capaz de evitar la suplantación de identidad en un escenario donde el estudiante tenga que rendir una evaluación a través de plataformas virtuales con la presencia remota del docente, a través del uso de técnicas de reconocimiento facial, que sean fiables y viables para una institución educativa.

2.7.1.4 *Stakeholders*

Rol	Responsabilidades
Estudiante	• Contar con una *WebCam* instalada y configurada para un proceso de

	videollamada. • Estar presto a las indicaciones que emita el sistema, tales como: Posicionarse frente a la cámara, mover ligeramente su cabeza o colocar en diversas posiciones. • Proporcionar información personal y de identificación, incluyendo imágenes faciales, para registrarse en el sistema. • Cumplir con los requisitos de verificación de identidad antes de iniciar el examen en línea. • Aceptar los términos y condiciones del uso del sistema de reconocimiento facial. • Participar activamente en la verificación de identidad durante la realización del examen en línea, lo que incluye mantener una postura y una distancia adecuadas frente a la cámara y permitir que se capte una imagen clara de su rostro.
Docente	• Iniciar el sistema de evaluaciones en línea, autenticándose con su usuario y contraseña proporcionadas para el acceso al sistema. • Establecer políticas claras y detalladas para la verificación de identidad a través del reconocimiento facial. • Monitorear y administrar el uso del sistema de reconocimiento facial, incluyendo la identificación y resolución de problemas técnicos. • Proporcionar instrucciones claras y detalladas a los estudiantes sobre cómo participar en la verificación de identidad. • Proteger la privacidad y la seguridad de la información personal y las imágenes faciales de los estudiantes. • Mantener un registro de los resultados de la verificación de identidad y tomar medidas para garantizar la integridad y la autenticidad de los resultados de los exámenes en línea. • Monitorear los registros de eventos

Tabla 2. Matriz de stakeholder

Fuente: Autor.

2.7.1.5 Alcance

Se definieron los alcances juntamente con los *stakeholders* detallados en la siguiente tabla:

- **Verificación de identidad:** El sistema deberá ser capaz de verificar de manera fiable la identidad de los estudiantes antes de que comiencen el examen en línea.

- **Prevención de suplantación de identidad:** Al verificar la identidad de los estudiantes antes y durante el examen, se evita la posibilidad de que otras personas tomen el examen en su nombre.
- **Integridad de los resultados:** Al evitar la suplantación de identidad, se garantiza la integridad de los resultados de los exámenes en línea.
- **Facilidad de uso:** El sistema debería ser fácil de usar para los estudiantes y los docentes, con instrucciones claras y un proceso de verificación de identidad intuitivo.
- **Seguridad de la información personal:** El sistema debería proteger la privacidad y la seguridad de la información personal y las imágenes faciales de los estudiantes.
- **Eficiencia:** El sistema debería ser eficiente y permitir la verificación de la identidad de los estudiantes de manera rápida y sencilla.
- **Compatibilidad:** El sistema debería ser compatible con una amplia variedad de dispositivos y sistemas operativos, incluyendo computadoras, tabletas y teléfonos móviles.
- **Seguridad:** La solución contempla medidas de seguridad para evitar cualquier pericia, tales como: el uso de cámaras virtuales, suplantación de identidad con fotografías o videos, así también, el prototipo prevalece y salvaguarda la privacidad de la información, fotografías y datos personales del individuo se deben de usar exclusivamente para el cometido del proceso. La toma de decisión siempre estará del lado del docente, ya que el mismo cuenta con panel de monitoreo donde puede observar todas las novedades durante el proceso

2.7.1.6 Requerimientos Funcionales

En la siguiente tabla se detallan los requerimientos que fueron levantados durante el desarrollo de esta fase:

- **Registro de usuario:** El sistema debería permitir a los estudiantes registrarse en el sistema proporcionando información personal e imágenes faciales.
- **Verificación de identidad:** El sistema debería ser capaz de verificar la identidad de los estudiantes antes de iniciar el examen en línea.
- **Captura de imágenes:** El sistema debería tener la capacidad de capturar imágenes faciales de alta calidad para su posterior verificación.
- **Análisis de imágenes:** El sistema debería ser capaz de analizar las imágenes faciales para compararlas con las imágenes registradas y determinar si la persona es la misma.

- **Interfaz de usuario intuitiva:** El sistema debería tener una interfaz de usuario intuitiva que permita a los estudiantes y docentes interactuar con el sistema de manera sencilla.
- **Notificaciones:** El sistema debería generar notificaciones en caso de fallos en la verificación de identidad o problemas técnicos.
- **Reportes:** El sistema debería generar reportes sobre la verificación de identidad y los resultados de los exámenes en línea.
- **Seguridad de la información:** El sistema debería implementar medidas de seguridad adecuadas para proteger la privacidad y seguridad de la información personal y las imágenes faciales de los estudiantes.

Requerimientos funcionales	
Código	**Descripción**
REQ001	Registro de usuario
REQ002	Verificación de identidad
REQ003	Captura de imágenes
REQ004	Análisis de imágenes
REQ005	Interfaz de usuario intuitiva
REQ006	Notificaciones
REQ007	Reportes
REQ008	Seguridad de la información

Tabla 3. Requerimientos funcionales

Fuente: Autor.

De cada uno de los requerimientos se especifican en las siguientes tablas:

Código:	REQ001	**Prioridad:**	Alta
Nombre:	Registro de usuario		
Fecha:	02-02-2023	**Responsable:**	Carlos Quezada C.

Observación:	El sistema debería permitir a los estudiantes registrarse en el sistema proporcionando información personal e imágenes faciales.
Actores:	Estudiantes
Condiciones:	Es importante asegurarse de que el proceso de registro sea fácil de seguir y que los estudiantes proporcionen la información necesaria e imágenes faciales adecuadas para una verificación efectiva.

Tabla 4. REQ001 Registro de usuario

Fuente: Autor.

Código:	REQ002	**Categoría**:	Alta
Nombre:	Verificación de identidad		
Fecha:	02-02-2023	**Responsable**:	Carlos Quezada C.
Observación:	El sistema debería ser capaz de verificar la identidad de los estudiantes antes de iniciar el examen en línea.		
Actores:	Estudiantes		
Condiciones:	Es esencial que el sistema sea capaz de verificar la identidad de los estudiantes de manera precisa y confiable para evitar la suplantación de identidad.		

Tabla 5. REQ002 Verificación de identidad

Fuente: Autor.

Código:	REQ003	**Categoría**:	Alta
Nombre:	Captura de imágenes		
Fecha:	02-02-2023	**Responsable**:	Carlos Quezada C.
Observación:	El sistema debería tener la capacidad de capturar imágenes faciales de alta calidad para su posterior verificación.		
Actores:	Estudiantes		
Condiciones:	Es importante que el sistema capture imágenes faciales de alta calidad para que la verificación sea efectiva.		

Tabla 6. REQ003 Captura de imágenes.

Fuente: Autor.

Código:	REQ004	**Categoría**:	Alta
Nombre:	Análisis de imágenes		
Fecha:	02-02-2023	**Responsable**:	Carlos Quezada C.
Observación:	El sistema debería ser capaz de analizar las imágenes faciales para compararlas con las imágenes registradas y determinar si la persona es la misma.		
Actores:	Estudiantes		
Condiciones:	Es crucial que el sistema sea capaz de analizar las imágenes faciales y realizar una comparación precisa con las imágenes registradas.		

Tabla 7. REQ004 Análisis de imágenes

Fuente: Autor.

Código:	REQ005	**Categoría**:	Media
Nombre:	Interfaz de usuario intuitiva		
Fecha:	02-02-2023	**Responsable**:	Carlos Quezada C.
Observación:	El sistema debería tener una interfaz de usuario intuitiva que permita a los estudiantes y docentes interactuar con el sistema de manera sencilla.		
Actores:	Estudiantes, Docentes		
Condiciones:	Una interfaz de usuario intuitiva y fácil de usar es importante para garantizar que los estudiantes y docentes puedan interactuar con el sistema sin problemas.		

Tabla 8. REQ005 Interfaz de usuario

Fuente: Autor.

Código:	REQ006	**Categoría**:	Alta
Nombre:	Notificaciones		
Fecha:	02-02-2023	**Responsable**:	Carlos Quezada C.
Observación:	El sistema debería generar notificaciones en caso de fallos en la verificación de identidad o problemas técnicos.		

Actores:	Estudiantes, Docentes
Condición:	Las notificaciones permiten mantener a los usuarios informados sobre problemas técnicos y fallos en la verificación de identidad.

Tabla 9. REQ006 Notificaciones

Fuente: Autor.

Código:	REQ007	**Categoría**:	Alta
Nombre:	Reportes		
Fecha:	02-02-2023	**Responsable**:	Carlos Quezada C.
Observación:	El sistema deberá generar reportes sobre la verificación de identidad y los resultados de los exámenes en línea.		
Actores:	Docentes		
Condiciones:	Los reportes son importantes para evaluar el desempeño del sistema y para comprender los resultados de los exámenes en línea.		

Tabla 10. REQ007 Reportes

Fuente: Autor.

Código:	REQ008	**Categoría**:	Alta
Nombre:	Seguridad de la información		
Fecha:	02-02-2023	**Responsable**:	Carlos Quezada C.
Observación:	El sistema debería implementar medidas de seguridad adecuadas para proteger la privacidad y seguridad de la información personal y las imágenes faciales de los estudiantes.		
Actores:	Estudiantes, Docentes		
Condiciones:	Es esencial que el sistema implemente medidas de seguridad adecuadas para proteger la privacidad y seguridad de la información personal y las imágenes faciales de los estudiantes.		

Tabla 11. REQ008 Seguridad de la información.

Fuente: Autor.

Se han elegido 3 algoritmos como candidatos para el modelo de reconocimiento de rostros, ya que son algoritmos que han sido reconocidos y ampliamente utilizados en el campo del reconocimiento facial. Cada uno de ellos tiene características únicas y diferentes fortalezas, lo que permite compararlos y elegir el que mejor se adapte a los requisitos y objetivos de la investigación. Al utilizar tres algoritmos diferentes (tabla 12), se espera tener una comprensión más completa y profunda de las fortalezas y debilidades de cada uno, y utilizar esta información para desarrollar un modelo de reconocimiento de rostros que sea preciso, eficiente y adaptable a diferentes condiciones y entornos. Así mismo, permitirá evaluar y comparar diferentes enfoques y técnicas para el reconocimiento facial, y elegir el que mejor cumpla con los requisitos y objetivos de la investigación. Los mismos serán expuestos a pruebas en el caso de estudio de la investigación.

Algoritmo	**Biblioteca**	**Técnica**
HaarCascade	EigenFace	Clasificadores de cascada
Dlib	Face-recognition - CaffeModel	Redes neuronales Convolucionales
Facenet	Tensorflow	Deep-Learning

Tabla 12. Algoritmos seleccionados.

Fuente: Autor.

Como ya se profundizó en el capítulo 1, *HaarCascade* es un algoritmo de detección de rostros que se utiliza comúnmente para la detección de objetos en imágenes y videos, *FaceRecognition* es una biblioteca de Python que utiliza técnicas de aprendizaje profundo para realizar tareas de reconocimiento facial y *FaceNet* es un modelo de aprendizaje profundo desarrollado por *Google* que ofrece una alta precisión en tareas de reconocimiento facial.

2.7.1.7 Criterios para los experimentos

Las pruebas para el reconocimiento de personas se realizaron bajo los siguientes criterios:

- Número de estudiantes sujetos a las pruebas: 10.
- Cada estudiante simula efectuar un examen y se ubica frente al computador durante 5 minutos.
- Todos los algoritmos se realizan bajo las mismas condiciones de iluminación y entorno.
- Para equilibrar las pruebas y que los resultados de cada experimento estén sometidos bajo las mismas condiciones, se realizó un video de la iteración de cada estudiante. La

secuencia de video es evaluada por todos los algoritmos en cada experimento, con esto se asegura que todos los experimentos efectuaron el mismo reconocimiento sobre una misma toma.

- Todos los modelos fueron entrenados con 600 rostros de cada estudiante.
- Se utiliza una matriz de confusión la cual es llenada con base a los resultados emitidos por cada algoritmo. Cada algoritmo tiene un contador en donde indica cuantas veces fue reconocido el rostro y cuantas veces no fue capaz de reconocer la identidad del sujeto, así mismo cuantas veces el algoritmo interpretó con otra identidad al sujeto y las veces que indicó una identidad sobre una persona desconocida.

2.7.2 Fase de comprensión de los datos

Para la construcción del modelo, se extraen rostros de individuos, los cuales sirven para el entrenamiento del mismo. En primer lugar, se recopila una base de datos de imágenes de rostros de los estudiantes autorizados para tomar el examen. Estas imágenes deben ser tomadas con una webcam y deben ser de alta calidad para garantizar un rendimiento óptimo del sistema. Dentro de esta fase se prueban 3 algoritmos los cuales permiten detectar en una secuencia de video o en tiempo real a través de una webcam el rostro de una persona. Se prueban los algoritmos: *HaarCascade*, Redes Neuronales Convolucionales (CNN) y utilizando técnicas de aprendizaje profundo - DNN con modelos pre entrenados de la biblioteca de *Google Tensorflow*.

2.7.2.1 Desarrollo de los experimentos

2.7.2.1.1 Detección usando el Algoritmo: *HaarCascade*

Algoritmo: *EigenFace*

Técnica: Clasificadores de Cascada

El algoritmo *HaarCascade,* contiene un conjunto de cascadas de *Haar* pre entrenadas que pueden utilizarse para detectar objetos específicos en imágenes. Estas cascadas se almacenan en archivos XML y se utilizan para describir las características de los objetos que se quieren detectar. Algunos de los archivos XML incluidos en la librería son:

- ***haarcascade_frontalface_default.xml*** que detecta rostros humanos en imágenes.
- ***haarcascade_eye.xml*** que detecta ojos humanos en imágenes.
- ***haarcascade_fullbody.xml*** que detecta cuerpos humanos completos en imágenes.

- ***haarcascade_upperbody.xml*** que detecta cuerpos humanos desde la cintura hacia arriba en imágenes.

Para el proceso de detección de rostros se usará el archivo *haarcascade_frontalface_default.xml*, el cual contiene una cascada de *Haar* pre entrenada para reconocer características específicas de los rostros, como la forma de la mandíbula, la forma de los ojos y la forma de la nariz y así detectar rostros humanos en imágenes y vídeos.

detectorcaras.py > ...

```python
import cv2
import numpy as np

cap = cv2.VideoCapture(0)

faceClassif = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

while True:
    ret,frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    faces = faceClassif.detectMultiScale(gray, 1.3, 5)

    for (x,y,w,h) in faces:
        cv2.rectangle(frame, (x,y),(x+w,y+h),(0,255,0),2)

    cv2.imshow('frame',frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

Figura 38. Código fuente detección rostro haarCascade

Fuente: Autor.

El algoritmo detecta el rostro, el resultado es el siguiente:

Figura 39. Resultado de la detección de rostro usando EigenFace

Fuente: Autor.

Se aplicaron diferentes pruebas al algoritmo, las cuales incluyeron secuencias de vídeo (Fig.

43) e imágenes (Fig. 44) los cuales contenían diferentes rostros, para que el algoritmo sea capaz de detectar, extraer los rostros y verificar cuantos se podían almacenar en una carpeta local. A continuación, se muestra la detección de rostros en una imagen, para poder analizar cuántos rostros es capaz de detectar el algoritmo.

har.py > ...

```python
import cv2
# Cargar el clasificador de rostro Haar
face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
# Leer la imagen y convertirla a escala de grises
img = cv2.imread('rostros5_1.jpg')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
# Detectar rostros en la imagen
faces = face_cascade.detectMultiScale(gray, 1.3, 5)
# Dibujar un rectángulo alrededor de cada rostro detectado
for (x,y,w,h) in faces:
    cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
# Mostrar la imagen con los rectángulos dibujados
cv2.imshow('Detección de rostros - UtMachala',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Figura 40. Código en fuente para detectar rostros usando HaarCascade

Fuente: Autor.

Figura 41. Resultado del algoritmo HaarCascade

Fuente: Autor

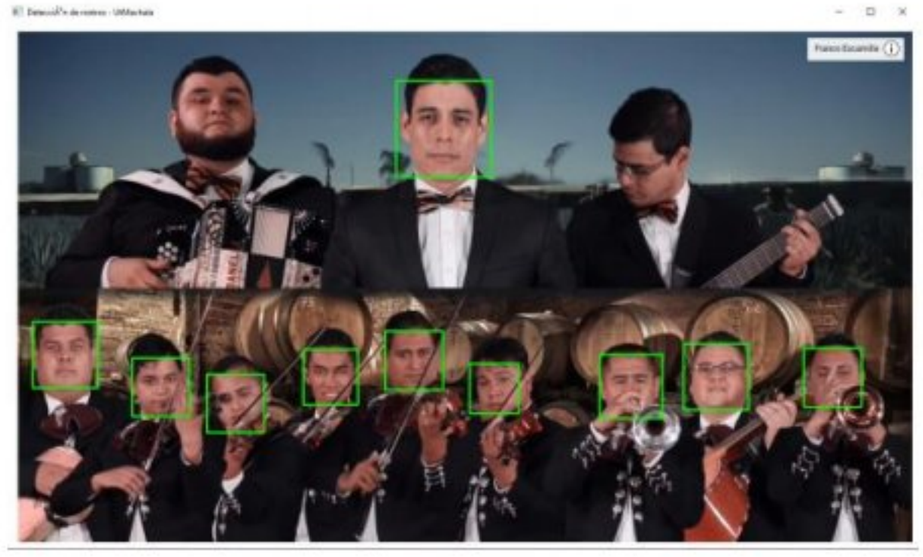

Figura 42. Resultado del algoritmo HaarCascade

Fuente: Autor

Se puede apreciar que el algoritmo no es capaz de detectar dos rostros en la secuencia de video (Figura 44), sin embargo, en una imagen si es capaz de detectar todos los rostros dentro de ella (Figura 43).

2.7.2.1.2 Detección de rostros usando el algoritmo: Dlib

Biblioteca: Face-recognition - **Framework:** Caffemodel

Técnica: Red Neuronal Convolucional (CNN)

Así mismo se puede realizar la detección y reconocimiento de rostros a través de modelos ya

entrenados y el uso de framework, tal es el caso de *caffemodel*, que es un modelo que contiene los pesos y configuraciones de una red neuronal ya entrenada. Caffe utiliza una variedad de algoritmos de aprendizaje profundo, incluyendo convolutional neural networks (ConvNets o CNNs), recurrent neural networks (RNNs), y long short-term memory networks (LSTMs). También admite la integración de otros algoritmos de aprendizaje supervisado y no supervisado. En general, Caffe permite la implementación y entrenamiento de modelos personalizados, lo que lo hace una herramienta poderosa y versátil para muchas tareas de visión por computadora y procesamiento de señales. El modelo *Caffe* se utiliza para inferencia, es decir, para aplicar las inferencias de una red neuronal ya entrenada a nuevos datos de entrada. Una vez que se tiene un modelo *Caffe*, se puede utilizar para clasificar imágenes, detectar objetos, generar imágenes y muchas otras tareas relacionadas con el aprendizaje automático.

```python
import cv2
# ------------ Leer el modelo DNN ------------
# Arquitectura del modelo
prototxt = "model/deploy.prototxt"
# Pesos
model = "model/res10_300x300_ssd_iter_140000.caffemodel"
# Leer el modelo
net = cv2.dnn.readNetFromCaffe(prototxt, model)
# ------- Leer la imagen y procesarla -------
cap = cv2.VideoCapture(0)
while True:
    ret, frame = cap.read()
    if ret == False:
        break
    height, width, _ = frame.shape
    frame_resized = cv2.resize(frame, (300, 300))
    # Create a blob
    blob = cv2.dnn.blobFromImage(frame_resized, 1.0, (300, 300), (104, 117, 123))
    # ------- DETECTIONS AND PREDICTIONS ----------
    net.setInput(blob)
    detections = net.forward()
    print("detections.shape:", detections.shape)
    for detection in detections[0][0]:
        #print("detection:", detection)
        if detection[2] > 0.5:
            box = detection[3:7] * [width, height, width, height]
            x_start, y_start, x_end, y_end = int(box[0]), int(box[1]), int(box[2]), int(box[3])
            cv2.rectangle(frame, (x_start, y_start), (x_end, y_end), (0, 255, 0), 2)
            cv2.putText(frame, "Rostro: {:.2f}".format(detection[2] * 100), (x_start, y_start - 5), 1, 1.2, (0, 255, 255), 2)
    cv2.imshow("Detección de Rostros - Usando un modelo Deep Learning", frame)
    if cv2.waitKey(1) & 0xFF == 27:
        break
cap.release()
cv2.destroyAllWindows()
```

Figura 43. Código en python para detectar rostros usando Framework: Caffemodel

Fuente: Autor

El modelo es capaz de identificar un rostro humano.

Figura 44. Resultado de la detección de rostro usando Código Framework: Caffemodel

Fuente: Autor

2.7.2.1.3 Detección usando el algoritmo Facenet

Técnica: Redes neuronales convolucionales (CNN)

Biblioteca: Tensorflow

Se realiza una captura de rostros usando una imagen o una secuencia de video en donde exista una persona o un grupo de personas desde la carpeta *train_img*. Esto se puede hacer utilizando una cámara web o cargando una imagen o video previamente almacenado, en este proceso se extraen los posibles rostros y se almacenan en la carpeta *aligned_img* para en el siguiente paso entrenar el modelo en base a esta última carpeta.

```python
import os
import cv2
import tensorflow as tf
# Carga el modelo entrenado
model = tf.keras.models.load_model('face_detection_model.h5')
# Inicializa la cámara
cap = cv2.VideoCapture(0)
# Crea una carpeta para almacenar las imágenes
if not os.path.exists("train_img"):
    os.makedirs("train_img")
count = 0
while True:
    # Lee un frame de la cámara
    ret, frame = cap.read()
    # Convierte el frame a una imagen de TensorFlow
    image = tf.keras.preprocessing.image.img_to_array(frame)
    image = tf.expand_dims(image, axis=0)
    # Realiza la detección de rostros
    faces = model.predict(image)
    # Dibuja un rectángulo alrededor de cada rostro y guarda la imagen
    for face in faces:
        x, y, w, h = face
        cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 0), 2)
        crop_img = frame[y:y+h, x:x+w]
        cv2.imwrite("train_img/face_{}.jpg".format(count), crop_img)
        count += 1
    # Muestra el frame con los rectángulos
    cv2.imshow('Face Detection', frame)
    # Si se presiona la tecla 'q', detiene el bucle
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
# Libera la cámara y cierra la ventana de OpenCV
cap.release()
cv2.destroyAllWindows()
```

Figura 45. Código en python para detectar y guardar rostros usando TensorFlow

Fuente: Autor

```python
from preprocess import preprocesses

input_datadir = './train_img'
output_datadir = './aligned_img'

obj=preprocesses(input_datadir,output_datadir)
nrof_images_total,nrof_successfully_aligned=obj.collect_data()

print('Total number of images: %d' % nrof_images_total)
print('Number of successfully aligned images: %d' % nrof_successfully_aligned)
```

Figura 46. Entrenamiento de imágenes capturadas

Fuente: Autor

2.7.3 Fase de preparación de datos

En esta fase, se preparan los datos, que en este caso son los rostros detectados en la fase anterior. Los rostros que cada algoritmo detectó, son extraídos y almacenados.

2.7.3.1 Extracción de rostros algoritmo haarCascade

Figura 47. Imágenes extraídas de una secuencia de video

Fuente: Autor

En esta prueba, se puede observar en la figura 49, las imágenes capturadas que corresponden a los rostros de las pruebas de la secuencia de video e imágenes de grupos de personas. Así mismo, en la misma figura, se puede observar que una de las imágenes extraídas corresponde a una parte de un rostro, pero el algoritmo lo considera como rostro, en este caso se considera un falso positivo.

Se realiza otra prueba con el algoritmo a un sujeto de prueba en tiempo real. El sujeto de prueba se colocó en diferentes posiciones para captar todas las perspectivas, gestos y contrastes de su rostro. Así como lo muestra la figura 50.

Figura 48. Detección de rostros en diversas posiciones.

Fuente: Autor

A continuación, se procede con la extracción de rostros para entrenar el modelo de reconocimiento facial usando el algoritmo *HaarCascade* desde una webcam como dispositivo

de entrada con su respectiva etiqueta de la persona. Ver código fuente del proceso en Anexo CV1. A continuación en la figura 51, se muestran los rostros extraídos mediante la detección que hizo el algoritmo

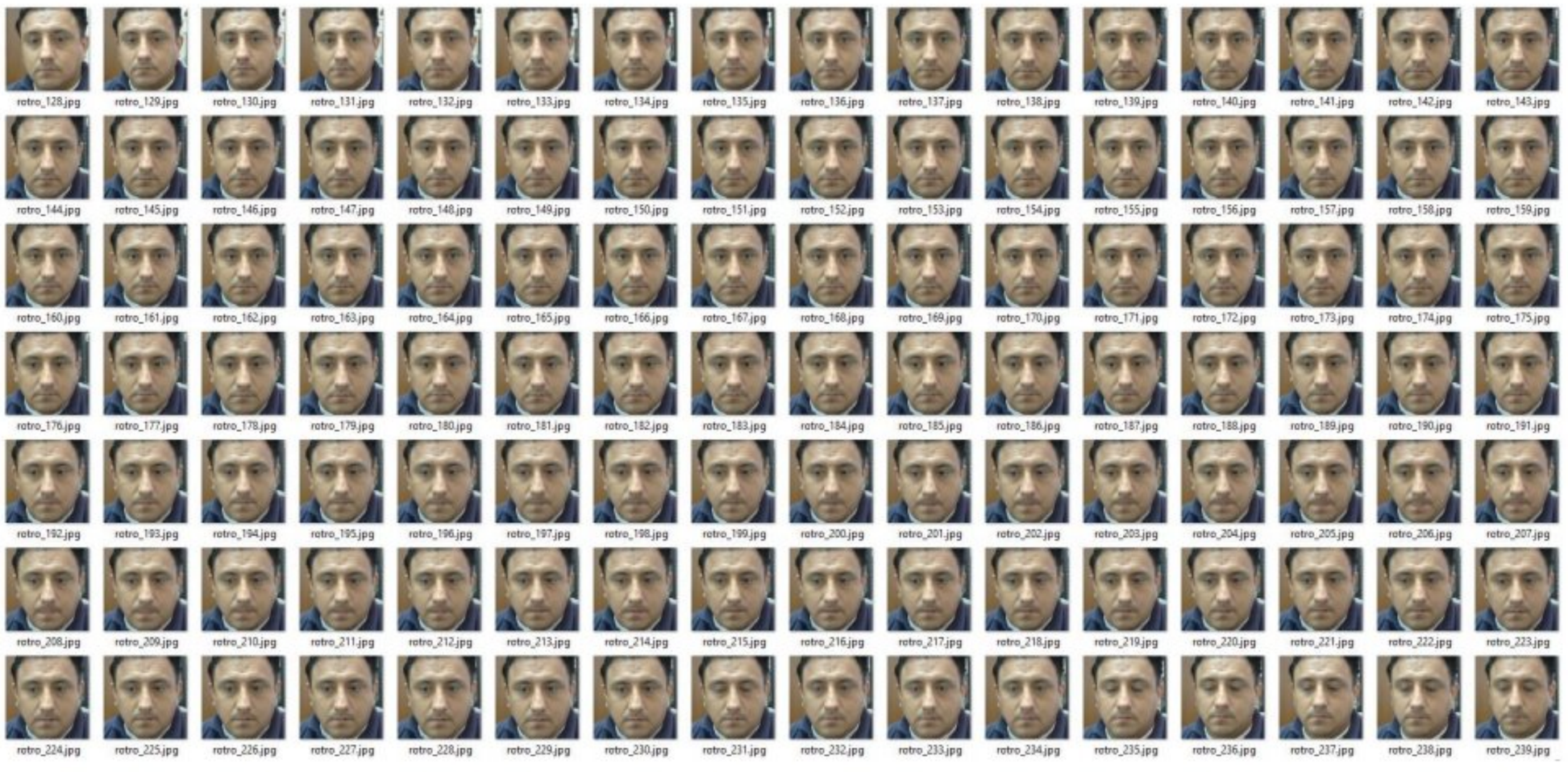

Figura 49. Rostros extraídos.

Fuente: Autor

El algoritmo es capaz de capturar rostros según las necesidades y exigencias para el entrenamiento del algoritmo, para la etapa de entrenamiento se capturan 300 rostros (fig. 51) y son almacenadas en la carpeta establecida para el sujeto de prueba, con el nombre que corresponde al rostro, en este caso “carlos”.

2.7.3.2 Extracción de rostros framework CaffeModel

De igual forma, en esta prueba, el individuo se posiciona frente a una *webcam*, se ejecuta el algoritmo de detección y extracción de rostros y se procede a obtener el rostro. Se capturan 300 imágenes de 150x150 píxeles (Fig. 52).

Figura 50. Rostros extraídos del framework CaffeModel.

Fuente: Autor

2.7.3.3 Resultados en la etapa de extracción de rostros

Al probar los dos algoritmos (*HaarCascade y Dlib Framework CaffeModel*) en la fase de extracción de rostros, se puede constatar que los dos algoritmos cumplen al 100% con el objetivo de la fase, ya que cuentan con modelos pre entrenados robustos los cuales permiten detectar el rostro humano de manera correcta.

2.7.4 Fase de modelado

2.7.4.1 Entrenamiento del modelo mediante la técnica de Eigenface

Para entrenar el modelo se usa *Eigenface*, el cual necesita un conjunto de imágenes de rostros etiquetadas. A partir de estas imágenes, se extraen las características de cada rostro y se utilizan para calcular un conjunto de caras propias. Estas caras propias se utilizaron como una especie de "base" para representar cada rostro de manera única.

En el Anexo CB3, se muestra el código usado para entrenar el modelo, el cual genera un modelo llamado *modelEigenFace.xml*

2.7.4.2 Entrenando el modelo usando Face_recognition

En el anexo CB4, se muestra el código usado para entrenar el modelo, el cual genera un modelo llamado *trained_knn_model.clf,* el cual es producto del entrenamiento que el algoritmo le da a través de los rostros suministrados de cada persona, este se almacena en la carpeta ***classifier,*** tal como se puede observar en la figura 53.

```
H:\tesis\DlibFaceRecognition-main>python Train_main.py
processing : train_img\alejandro\2023-01-07 15_03_27-Window.png
processing : train_img\alejandro\2023-01-07 15_03_39-Window.png
processing : train_img\alejandro\2023-01-07 15_05_15-Window.png
processing : train_img\carlos\fotocarlosq.png
processing : train_img\Maribel\foto1.png
Training complete
```

Figura 51. Resultado del entrenamiento

Fuente: Autor

2.7.3.3 Entrenando el modelo usando FaceNet

De igual forma, el tercer algoritmo efectúa el entrenamiento en base a los rostros extraídos en la fase anterior. El modelo pre entrenado llamado *20230113-Modelo.pb* se genera el cual se ha colocado de nombre: *classifier.pkl*.

```python
from __future__ import absolute_import
from __future__ import division
from __future__ import print_function
import sys
from classifier import training

datadir = './aligned_img'
modeldir = './model/20180402-114759.pb'
#modeldir = './model/20170511-185253.pb'
classifier_filename = './class/classifier.pkl'
print ("Training Start")
obj=training(datadir,modeldir,classifier_filename)
get_file=obj.main_train()
print('Saved classifier model to file "%s"' % get_file)
sys.exit("All Done")
```

Figura 52. Código fuente entrenamiento del modelo

Fuente: Autor

```
C:\pruebasTesis\metodo3\Facenet_Tensorflow-main>py train_main.py
Training Start
2023-02-01 13:27:27.874901: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimize
Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations:  AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
Classes: 14
Images: 163
Model filename: ./model/20230113-Modelo.pb
WARNING:tensorflow:From C:\pruebasTesis\metodo3\Facenet_Tensorflow-main\facenet.py:378: FastGFile.__init__ (from te
tform.gfile) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.gfile.GFile.
Extracting features of images for model
2023-02-01 13:27:32.105082: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:357] MLIR V1 optimization pa
2023-02-01 13:27:40.112111: W tensorflow/tsl/framework/cpu_allocator_impl.cc:82] Allocation of 247405312 exceeds 10
mory.
2023-02-01 13:27:40.334988: W tensorflow/tsl/framework/cpu_allocator_impl.cc:82] Allocation of 247405312 exceeds 10
mory.
Training Started
Saved classifier model to file "./class/classifier.pkl"
All Done
```

Figura 52. Código fuente entrenamiento del modelo

Fuente: Autor

Con la prueba de los algoritmos se han obtenido 3 modelos pre entrenados con los rostros del sujeto 1. Tal y como lo muestra la figura 55.

 modeloEigenFace.xml	 trained_knn_model.clf	 classifier.pkl
HaarCascade	Redes Convolucionales - K-Nearest Neighbors (KNN)	DeepLearning - FaceNet

Figura 53. Modelos obtenidos

Fuente: Autor

2.7.5 Evaluación

La fase 5 de CRISP-DM, se lleva a cabo en el capítulo III del estudio. Durante esta etapa, se obtienen los resultados mediante la aplicación de los criterios de validación establecidos. Se generan diferentes métricas de evaluación, como la matriz de confusión, precisión, recall y f1-score, para cada uno de los tres algoritmos. El análisis de estas métricas es fundamental para evaluar y comparar el desempeño de los algoritmos y determinar cuál de ellos se ajusta mejor a los objetivos y requisitos del proyecto.

2.8 Métodos empíricos

Para evaluar el rendimiento del prototipo se utilizan los siguientes métodos empíricos:

- **Pruebas de laboratorio:** Consiste en realizar pruebas en un entorno controlado para evaluar el rendimiento del sistema en un conjunto de datos conocido.
- **Pruebas de campo:** Consiste en evaluar el rendimiento del sistema en un entorno real con un conjunto de datos desconocido.
- **Evaluación comparativa:** Consiste en comparar el rendimiento de un sistema de reconocimiento facial con otros sistemas existentes en el mercado.
- **Pruebas de seguridad:** Consiste en evaluar la capacidad del sistema para proteger la

privacidad y seguridad de los datos.

- **Evaluaciones a largo plazo:** Consiste en evaluar el rendimiento del sistema a lo largo del tiempo para detectar cualquier cambio en el rendimiento.

CAPÍTULO 3

3.1 RESULTADOS

3.1.1 Fase de evaluación

Para determinar el algoritmo adecuado para el desarrollo del prototipo, se utilizó una matriz de confusión [117] [118], ya que esta herramienta es comúnmente utilizada para evaluar el rendimiento de un algoritmo de clasificación, tal y como se profundiza en el capítulo 1. La matriz de confusión se compone de indicadores o métricas, los cuales son: Verdaderos Positivos (VP), Falsos Positivos (FP), Verdaderos Negativos (VN) y Falsos Negativos (FN)m distribuidos de la siguiente forma en la matriz:

		Predicción	
		Positivos	**Negativos**
Observación	**Positivos**	Verdaderos Positivos (VP)	Falsos Negativos (FN)
	Negativos	Falsos Positivos (FP)	Verdaderos Negativos (VN)

Tabla 13. Matriz de confusión

Fuente: Autor

3.1.1.1 Métricas

Con estos cuatro valores se pueden calcular métricas como la precisión, *recall* o el valor de referencia.

$$Precisión = (Verdaderos\ Positivos) / (Verdaderos\ Positivos + Falsos\ Positivos)$$

$$Sensibilidad\ (Recall) = (Verdaderos\ Positivos) / (Verdaderos\ Positivos + Falsos\ Negativos)$$

$$Valor\ de\ referencia = 2\ X\ (Precision\ X\ Sensibilidad)/(Precisión + Sensibilidad)$$

3.1.4.2 Índice o coeficiente de Kappa

Los índices Kappa, o coeficientes kappa, son medidas estadísticas utilizadas para evaluar la concordancia o acuerdo entre dos o más observadores o evaluadores en una tarea de clasificación o medición [119]. El coeficiente kappa mide la proporción de acuerdo que se logra después de eliminar el acuerdo que se produciría por azar [120][121]. En esencia, el coeficiente kappa compara la cantidad de acuerdos observados entre los evaluadores con la cantidad de acuerdos que se esperarían por azar. Un coeficiente kappa igual a 1 indica una concordancia

perfecta, mientras que un coeficiente kappa igual a 0 indica una concordancia igual a la que se produciría por azar [122]. Los coeficientes kappa se utilizan en una amplia variedad de disciplinas para evaluar la fiabilidad de la medición y la concordancia entre los observadores, incluyendo la psicología, la medicina, la epidemiología, la ingeniería y la ciencia de la computación. El valor kappa está dentro del rango [-1, 1], sea k el índice de *kappa* entonces según [122], se define:

$k < 0$: Negativo, significa no existe concordancia, puede alcanzar hasta -1

$k = 0$: Los valores observados son independientes.

$k > 0$: Positiva, existe concordancia; siendo +1 concordancia perfecta.

Coeficiente Kappa	**Fuerza de concordancia**
0,00	Pobre (*Poor*)
0,01 - 0,20	Leve (*Slight*)
0,21 - 0,40	Aceptable (*Fair*)
0,41 - 0,60	Moderada (*Moderate*)
0,61 - 0,80	Considerable (*Substantial*)
0,81 - 1,00	Casi perfecta (*Almost perfect*)

Tabla 14. Valoración de concordancia del coeficiente Kappa.

Fuente: Tomada de [122][123]

3.1.4.3 Resultados Obtenidos Experimento 1 - *HaarCascade - Eigenface*

EXPERIMENTO 1	MATRIZ DE CONFUSIÓN										
	PREDICCIÓN										TOTAL
INDIVIDUO	Estudiante 1	Estudiante 2	Estudiante 3	Estudiante 4	Estudiante 5	Estudiante 6	Estudiante 7	Estudiante 8	Estudiante 9	Estudiante 10	
Estudiante 1	**1204**	0	15	0	5	1	0	0	15	0	1240
Estudiante 2	0	**1178**	5	16	0	0	0	0	0	11	1210
Estudiante 3	0	0	**1358**	0	25	0	0	1	10	19	1413
Estudiante 4	20	12	0	**1102**	0	20	25	2	10	9	1200
Estudiante 5	0	0	0	0	**1068**	15	0	12	16	0	1111
Estudiante 6	0	0	14	16	2	**1378**	0	12	0	8	1430
Estudiante 7	23	0	13	17	26	0	**1698**	2	6	12	1797
Estudiante 8	12	0	18	2	14	0	0	**1287**	0	13	1346
Estudiante 9	19	0	2	5	2	0	0	0	**1076**	0	1104
Estudiante 10	24	0	14	6	0	2	16	14	0	**1179**	1255
SUMA TOTAL	**1302**	**1190**	**1439**	**1164**	**1142**	**1416**	**1739**	**1330**	**1133**	**1251**	**13106**
PRECISIÓN	92,47%	98,99%	94,37%	94,67%	93,52%	97,32%	97,64%	96,77%	94,97%	94,24%	95,50%
ERROR DE OMISIÓN	7,53%	1,01%	5,63%	5,33%	6,48%	2,68%	2,36%	3,23%	5,03%	5,76%	
ERROR DE COMISION	2,90%	5,00%	3,89%	8,17%	3,87%	3,64%	5,51%	4,38%	2,54%	6,06%	
PRECISIÓN GENERAL	95,50%										

Tabla 15. Experimento 1 - HaarCascade – Eigenface

Fuente: Autor

Resultados obtenidos en las métricas de evaluación experimento 1

EXPERIEMENTO 1	MATRIZ DE OBSERVACIÓN								
	TP	TN	FP	FN	Precisión	Recall	Po	Pe	K
Estudiante 1	1204	168	98	24	0,92473	0,9805	0,9183	0,7392	0,6869
Estudiante 2	1178	189	12	37	0,98992	0,9695	0,9654	0,7438	0,865
Estudiante 3	1358	201	81	26	0,94371	0,9812	0,9358	0,7406	0,7524
Estudiante 4	1102	36	62	16	0,94674	0,9857	0,9359	0,8835	0,4492
Estudiante 5	1068	147	74	50	0,9352	0,9553	0,9074	0,7364	0,6487
Estudiante 6	1378	100	38	15	0,97316	0,9892	0,9654	0,8483	0,7718
Estudiante 7	1698	182	41	10	0,97642	0,9941	0,9736	0,8081	0,8624
Estudiante 8	1287	178	43	14	0,96767	0,9892	0,9625	0,7653	0,8404
Estudiante 9	1076	201	57	5	0,94969	0,9954	0,9537	0,7128	0,8388
Estudiante 10	1179	145	72	141	0,94245	0,8932	0,8614	0,7253	0,4956
								PROMEDIO KAPPA	0,7211

Tabla 16. Resultados obtenidos en las métricas de evaluación experimento 1

Fuente: Autor

Análisis

En los experimentos realizados con el primer algoritmo propuesto, se obtuvieron los siguientes resultados:

- La precisión general del experimento alcanzó el 95.50%, lo que indica un buen desempeño en el reconocimiento facial.
- Se observó que el algoritmo logró reconocer al estudiante 2 con una alta precisión de

98.99%, destacando su eficacia en la identificación de este individuo en particular. Sin embargo, se identificó que el estudiante 1 obtuvo la precisión más baja, con un 92.47%. Es importante considerar la necesidad de mejorar el reconocimiento facial para este estudiante en futuros experimentos.

- Los índices promedio de Kappa mostraron un nivel de reconocimiento facial considerable, con un valor de 0.7211. Esto sugiere una concordancia razonable entre las predicciones del algoritmo y los resultados reales.

Estos resultados destacan la efectividad general del primer algoritmo en el reconocimiento facial, aunque se requiere un análisis más detallado y comparativo con los otros algoritmos propuestos para obtener una evaluación completa de su desempeño.

3.1.4.4 Resultados Obtenidos Experimento 2 - *Dlib, FaceRecognition* y Knn

EXPERIMENTO 2	MATRIZ DE CONFUSIÓN										
	PREDICCIÓN										
INDIVIDUO	Estudiante 1	Estudiante 2	Estudiante 3	Estudiante 4	Estudiante 5	Estudiante 6	Estudiante 7	Estudiante 8	Estudiante 9	Estudiante 10	TOTAL
Estudiante 1	957	2	1	0	8	14	2	0	1	3	988
Estudiante 2	0	628	3	4	8	11	2	1	3	7	667
Estudiante 3	2	1	1189	0	2	8	7	0	0	0	1209
Estudiante 4	0	3	5	967	0	0	7	0	0	2	984
Estudiante 5	5	0	0	5	1258	1	9	0	0	1	1279
Estudiante 6	1	1	1	2	0	1189	2	1	0	3	1200
Estudiante 7	0	0	2	1	7	0	1124	0	0	8	1142
Estudiante 8	0	0	3	1	9	1	0	971	1	1	987
Estudiante 9	1	2	0	0	0	3	1	0	936	2	945
Estudiante 10	0	3	7	0	2	1	8	6	2	1138	1167
SUMA TOTAL	966	640	1211	980	1294	1228	1162	979	943	1165	10568
PRECISIÓN	99,07%	98,13%	98,18%	98,67%	97,22%	96,82%	96,73%	99,18%	99,26%	97,68%	98,09%
ERROR DE OMISIÓN	0,93%	1,88%	1,82%	1,33%	2,78%	3,18%	3,27%	0,82%	0,74%	2,32%	
ERROR DE COMISION	3,14%	36,44%	1,65%	1,73%	1,64%	0,92%	1,58%	1,62%	0,95%	2,49%	
PRECISIÓN GENERAL	98,09%										

Tabla 17. Experimento 2 - Dlib, FaceRecognition y Knn

Fuente: Autor

Resultados obtenidos en las métricas de evaluación experimento 2

EXPERIEMENTO 2	MATRIZ DE OBSERVACIÓN								
	TP	TN	FP	FN	Precisión	Recall	Po	Pe	K
Estudiante 1	957	168	9	24	0,99068	0,9755	0,9715	0,732	0,8937
Estudiante 2	628	189	12	37	0,98125	0,9444	0,9434	0,6281	0,8479
Estudiante 3	1189	201	22	26	0,98183	0,9786	0,9666	0,736	0,8735
Estudiante 4	967	36	13	16	0,98673	0,9837	0,9719	0,9069	0,6981
Estudiante 5	1258	147	36	50	0,97218	0,9618	0,9423	0,7776	0,7407
Estudiante 6	1189	100	39	15	0,96824	0,9875	0,9598	0,8286	0,7654
Estudiante 7	1124	182	38	10	0,9673	0,9912	0,9645	0,7418	0,8627
Estudiante 8	971	178	8	14	0,99183	0,9858	0,9812	0,7293	0,9306
Estudiante 9	936	201	7	5	0,99258	0,9947	0,9896	0,7046	0,9646
Estudiante 10	1138	145	27	141	0,97682	0,8898	0,8842	0,7311	0,5694
								PROMEDIO KAPPA	0,8147

Tabla 18. Resultados obtenidos en las métricas de evaluación experimento 2

Fuente: Autor

Análisis

En el segundo experimento, se obtuvieron los siguientes resultados:

- El experimento logró una impresionante precisión general del 98.09%, lo cual demuestra un alto nivel de acierto en el reconocimiento facial.
- Se observó que el algoritmo alcanzó la máxima precisión en el estudiante 8, con un porcentaje de acierto del 99.26%. Por otro lado, el estudiante 7 obtuvo la precisión más baja, con un 96.73%. A pesar de esto, ambos valores siguen siendo muy altos y muestran una buena capacidad de reconocimiento facial por parte del algoritmo.
- Los índices promedio de Kappa revelaron un nivel casi perfecto de reconocimiento facial, con un valor de 0.8147. Esto indica una excelente concordancia entre las predicciones del algoritmo y los resultados reales.

Estos resultados destacan la alta precisión general alcanzada por el segundo algoritmo, junto con la capacidad de reconocer a los estudiantes con un elevado nivel de acierto. Además, los índices promedio de Kappa respaldan la calidad del reconocimiento facial realizado por el algoritmo. Estos hallazgos respaldan la eficacia y prometedor desempeño del segundo algoritmo en el contexto de la investigación, lo que sugiere su potencial para mitigar la suplantación de identidad en exámenes en línea.

3.1.4.5 Resultados Obtenidos Experimento 3 - FaceNet - Tensorflow

EXPERIMENTO 3	MATRIZ DE CONFUSIÓN										
	PREDICCIÓN										
INDIVIDUO	Estudiante 1	Estudiante 2	Estudiante 3	Estudiante 4	Estudiante 5	Estudiante 6	Estudiante 7	Estudiante 8	Estudiante 9	Estudiante 10	TOTAL
Estudiante 1	687	0	0	0	0	0	0	1	0	1	689
Estudiante 2	0	598	1	0	0	0	0	0	0	0	599
Estudiante 3	0	0	897	0	0	0	0	0	0	0	897
Estudiante 4	0	0	0	982	0	0	0	0	0	0	982
Estudiante 5	0	0	0	0	924	0	0	0	0	0	924
Estudiante 6	0	0	0	0	0	936	0	0	0	0	936
Estudiante 7	0	0	0	0	0	0	798	0	0	0	798
Estudiante 8	0	0	0	0	0	0	0	782	0	0	782
Estudiante 9	0	0	0	0	0	0	1	0	934	0	935
Estudiante 10	0	0	0	0	1	0	0	1	1	789	792
SUMA TOTAL	687	598	898	982	925	936	799	784	935	790	8334
PRECISIÓN	100,00%	100,00%	99,89%	100,00%	99,89%	100,00%	99,87%	99,74%	99,89%	99,87%	99,92%
ERROR DE OMISIÓN	0,00%	0,00%	0,11%	0,00%	0,11%	0,00%	0,13%	0,26%	0,11%	0,13%	
ERROR DE COMISION	0,29%	13,21%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,11%	0,38%	
PRECISIÓN GENERAL	99,92%										

Tabla 19. Experimento 3 - FaceNet – Tensorflow

Fuente: Autor

Resultados obtenidos en las métricas de evaluación experimento 3

EXPERIEMENTO 3	MATRIZ DE OBSERVACIÓN								
	TP	TN	FP	FN	Precisión	Recall	Po	Pe	K
Estudiante 1	687	168	0	24	1	0,9662	0,9727	0,6739	0,9163
Estudiante 2	598	189	0	37	1	0,9417	0,9551	0,6222	0,8812
Estudiante 3	897	201	1	26	0,99889	0,9718	0,976	0,6911	0,9223
Estudiante 4	982	36	0	16	1	0,984	0,9845	0,9184	0,8104
Estudiante 5	924	147	1	50	0,99892	0,9487	0,9545	0,7388	0,826
Estudiante 6	936	100	0	15	1	0,9842	0,9857	0,8163	0,9223
Estudiante 7	798	182	1	10	0,99875	0,9876	0,9889	0,6931	0,9638
Estudiante 8	782	178	2	14	0,99745	0,9824	0,9836	0,6914	0,9469
Estudiante 9	934	201	1	5	0,99893	0,9947	0,9947	0,7063	0,9821
Estudiante 10	789	145	1	141	0,99873	0,8484	0,868	0,6706	0,5993
								PROMEDIO KAPPA	0,877

Tabla 20. Resultados obtenidos en las métricas de evaluación experimento 3

Fuente: Autor

Análisis

En el tercer experimento realizado, se obtuvieron los siguientes resultados:

- El experimento logró una precisión general del 95.50%, indicando un buen desempeño en el reconocimiento facial.
- El algoritmo demostró una alta precisión al reconocer a cuatro estudiantes, alcanzando un 100% de precisión en estos casos.

- Sin embargo, se identificó que el estudiante 8 obtuvo la precisión más baja, con un 99.74%. Aunque sigue siendo una precisión muy alta, se podría investigar cómo mejorar el reconocimiento facial para este estudiante específico.
- Los índices promedio de Kappa revelaron un nivel casi perfecto de reconocimiento facial, con un valor de 0.877. Esto indica una concordancia muy alta entre las predicciones del algoritmo y los resultados reales.

Estos resultados resaltan la efectividad general del algoritmo en el tercer experimento, con una alta precisión y un nivel de reconocimiento facial casi perfecto. Sin embargo, es importante seguir explorando y mejorando el modelo para garantizar una precisión óptima en todos los casos, incluyendo el estudiante con menor precisión. Continuar evaluando y perfeccionando el algoritmo puede conducir a mejoras significativas en el reconocimiento facial en futuras implementaciones.

3.1.4.6 Resultados generales de los experimentos

EXPERIMENTO	Precisión
Experimento 1	95.20%
Experimento 2	98.09%
Experimento 3	99.91%

Tabla 21. Resultados generales de los experimentos

Fuente: Autor

Al efectuar la fase de experimentación, en la cual fueron expuestos cada algoritmo, bajo las mismas condiciones como se mencionó anteriormente, se puede observar en la tabla 21, que en el experimento 3, el grado de precisión es considerablemente más alto en relación al resto de pruebas efectuadas. En la fase experimentación se puede observar que *FaceNet* es más preciso que el resto de algoritmos. Estos hallazgos respaldan la eficacia y el rendimiento superior de FaceNet en el contexto específico de este estudio. La consistencia de los resultados y la destacada precisión obtenida en el Experimento 3 sugieren que FaceNet puede ser una opción prometedora para aplicaciones que requieran una alta precisión en el análisis

3.1.4.7 Resultados en base a la Matriz de confusión

ESTUDIANTE	PRECISIÓN		
	EXPERIMENTO 1	EXPERIMENTO 2	EXPERIMENTO 3
Estudiante 1	92,47%	99,07%	100,00%
Estudiante 2	98,99%	98,13%	100,00%
Estudiante 3	94,37%	98,18%	99,89%
Estudiante 4	94,67%	98,67%	100,00%
Estudiante 5	93,52%	97,22%	99,89%
Estudiante 6	97,32%	96,82%	100,00%
Estudiante 7	97,64%	96,73%	99,87%
Estudiante 8	96,77%	99,18%	99,74%
Estudiante 9	94,97%	99,26%	99,89%
Estudiante 10	94,24%	97,68%	99,87%

Tabla 22. Resultados en base a la Matriz de confusión precisión

Fuente: Autor

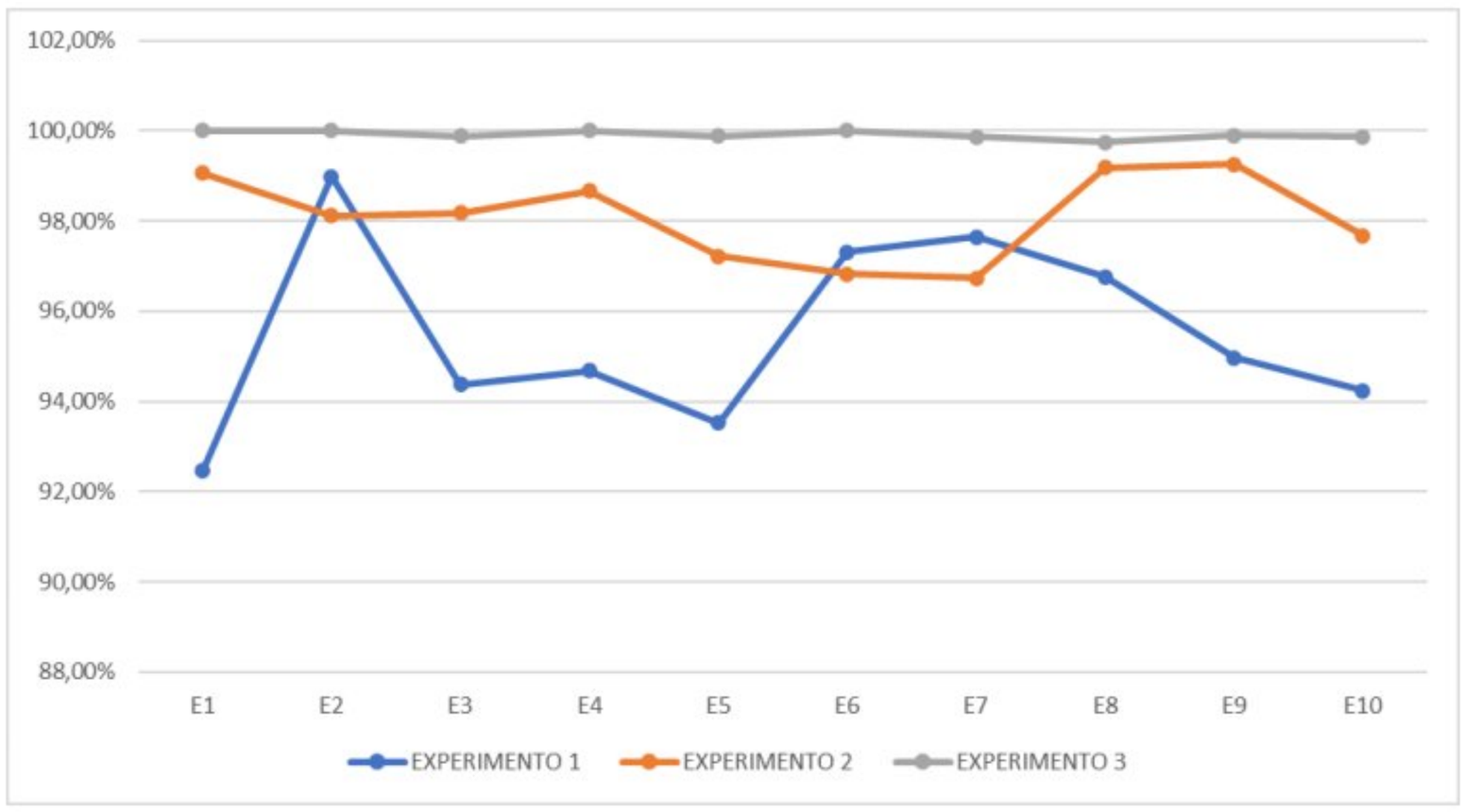

Figura 54. Resultados en base a la Matriz de confusión precisión

Fuente: Autor

ESTUDIANTE	Sensibilidad (Recall)		
	EXPERIMENTO 1	EXPERIMENTO 2	EXPERIMENTO 3
Estudiante 1	98,37%	98,76%	96,62%
Estudiante 2	98,25%	97,82%	94,17%
Estudiante 3	99,12%	99,33%	97,18%
Estudiante 4	99,46%	99,08%	98,40%
Estudiante 5	94,60%	97,98%	94,87%
Estudiante 6	99,14%	99,92%	98,42%
Estudiante 7	99,36%	99,56%	98,76%
Estudiante 8	96,62%	95,29%	98,24%
Estudiante 9	95,81%	97,30%	99,47%
Estudiante 10	94,85%	93,51%	84,84%

Tabla 23. Resultados en base a la Matriz de confusión sensibilidad (Recall)

Fuente: Autor

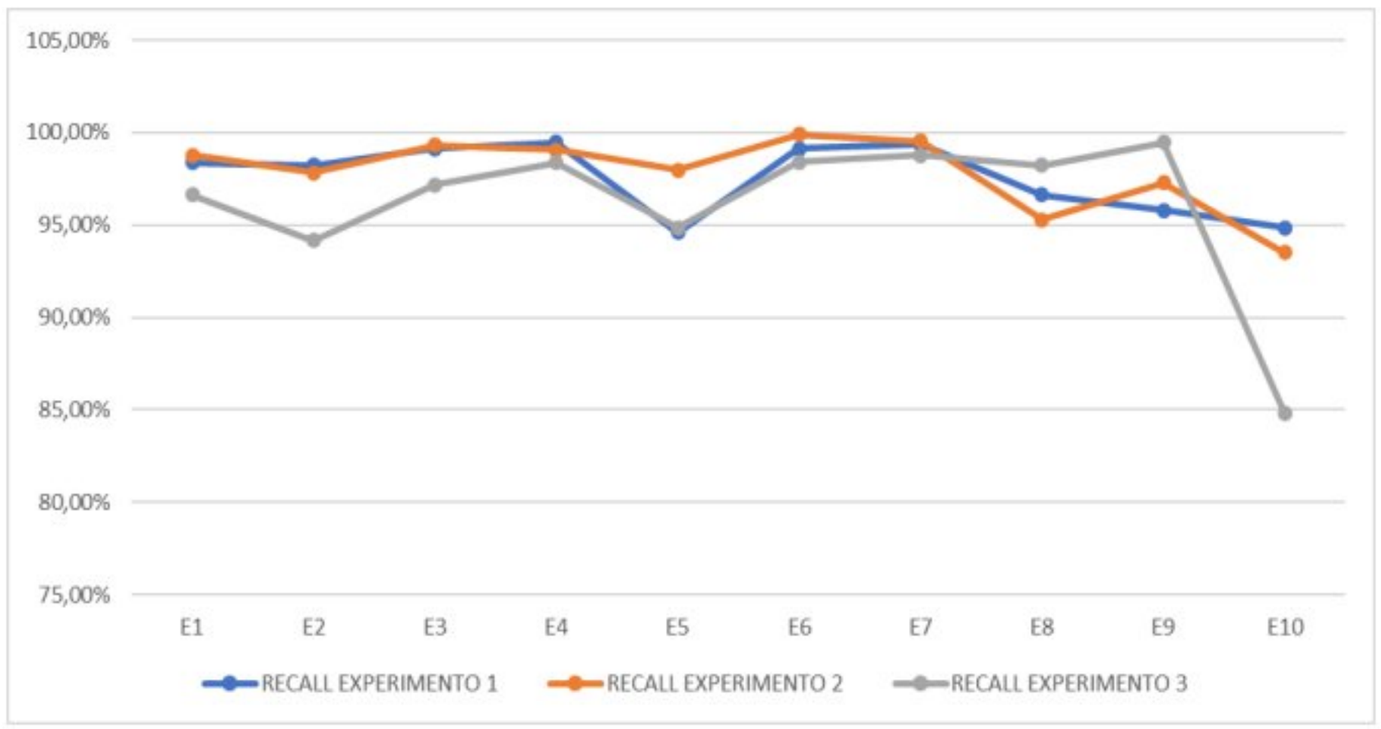

Figura 55. Resultados en base a la Matriz de confusión sensibilidad (Recall)

Fuente: Autor

De acuerdo a los resultados obtenidos, se puede observar que en el segundo experimento se evidencia un mayor grado de sensibilidad en el algoritmo de aprendizaje automático K-Nearest Neighbors (Knn) en comparación con los otros algoritmos evaluados en la fase de experimentación.

Es importante destacar que el término "sensibilidad" se refiere a la capacidad del algoritmo para responder y adaptarse a los cambios en los datos de entrada. Esta observación sugiere que

el algoritmo Knn muestra una mayor capacidad para capturar y responder a las variaciones en los datos durante el segundo experimento, en comparación con los otros algoritmos evaluados.

Contrastación de Hipótesis

A partir de los resultados obtenidos, se realizan la contrastación de las hipótesis planteadas por la investigación:

H1. Un modelo de reconocimiento facial basado en IA aplicado a la supervisión remota de exámenes en línea, mejorará la detección de suplantación de identidad identificando a una persona con una precisión mayor o igual al 98%.

Basándonos en los resultados de los experimentos, podemos concluir que se cumple la hipótesis planteada en la investigación. El modelo de reconocimiento facial basado en IA aplicado a la supervisión remota de exámenes en línea, logró mejorar la detección de suplantación de identidad, identificando a las personas con una precisión mayor o igual al 98%. Durante los experimentos, dos algoritmos demostraron consistentemente una precisión superior al umbral establecido, lo que respalda la validez de la hipótesis planteada. Sin embargo, es el experimento 3, con el algoritmo de *FaceNet* quién alcanza el 99.91% de precisión siendo el mejor y elegido para el desarrollo del prototipo.

Estos resultados destacan el potencial de los modelos de reconocimiento facial basados en IA como herramientas efectivas para abordar el desafío de la suplantación de identidad en los exámenes en línea, brindando una mayor confiabilidad y seguridad en el proceso de evaluación.

H2. Existen diferencias significativas en el desempeño de los diferentes algoritmos de reconocimiento facial en términos de precisión y eficiencia.

Prueba de Kruskal Wallis

	Precisión	Precisión	Precisión
ESTUDIANTE	**EXPERIMENTO 1**	**EXPERIMENTO 2**	**EXPERIMENTO 3**
Estudiante 1	0,92473	0,99098	1
Estudiante 2	0,98992	0,98125	1
Estudiante 3	0,94371	0,98183	0,99889
Estudiante 4	0,94674	0,98673	1
Estudiante 5	0,9352	0,97218	0,99892
Estudiante 6	0,97316	0,96824	1
Estudiante 7	0,97642	0,9673	0,99875
Estudiante 8	0,96767	0,99183	0,99745

Estudiante 9	0,94969	0,99258	0,99893
Estudiante 10	0,94245	0,97682	0,99873

Hipótesis

Ho= Experimento 1 igual a: Experimento 2 igual a: Experimento 3

H1= Experimento 1 diferente a: Experimento 2 diferente a: Experimento 3

Cálculo

GRUPO	VALOR	RANGO
Estudiante 1	0,92473	1
Estudiante 5	0,9352	2
Estudiante 10	0,94245	3
Estudiante 3	0,94371	4
Estudiante 4	0,94674	5
Estudiante 9	0,94969	6
Estudiante 7	0,9673	7
Estudiante 8	0,96767	8
Estudiante 6	0,96824	9
Estudiante 5	0,97218	10
Estudiante 6	0,97316	11
Estudiante 7	0,97642	12
Estudiante 10	0,97682	13
Estudiante 2	0,98125	14
Estudiante 3	0,98183	15
Estudiante 4	0,98673	16
Estudiante 2	0,98992	17
Estudiante 1	0,99098	18
Estudiante 8	0,99183	19
Estudiante 9	0,99258	20
Estudiante 8	0,99745	21
Estudiante 10	0,99873	22
Estudiante 7	0,99875	23
Estudiante 3	0,99889	24
Estudiante 5	0,99892	25
Estudiante 9	0,99893	26
Estudiante 1	1	28,5
Estudiante 2	1	28,5
Estudiante 4	1	28,5
Estudiante 6	1	28,5

N	27
K	3

R1	47,5
R2	59,5
R3	43
R4	49,5
R5	37
R6	48,5
R7	42
R8	48
R9	52
R10	38
H CACULADO	32,57671958
Critico	5,991464547

Resultado de la prueba

No es aceptable la Ho, porque el valor sobrepasa el valor crítico, por ende, se acepta H1, la cual supone que los experimentos son diferentes en sus resultados.

Con base en los resultados obtenidos mediante la prueba de Kruskal-Wallis, se confirma la existencia de diferencias significativas en el desempeño de los distintos algoritmos de reconocimiento facial, tanto en términos de precisión como de eficiencia. Los resultados de cada experimento revelan diferencias sustanciales entre los algoritmos evaluados, respaldando así la hipótesis planteada. Estos hallazgos demuestran que no todos los algoritmos de reconocimiento facial son igualmente efectivos ni eficientes, lo que resalta la importancia de seleccionar cuidadosamente el algoritmo adecuado para aplicaciones específicas.

3.1.5 Fase de Despliegue

3.1.5.1 Desarrollo del prototipo

Ver anexo 1

CAPÍTULO 4

4. Discusión de resultados

4.1 Hallazgos significativos

Al comparar *EigenFace*, *Face-recognition* y *FaceNet*, se pueden encontrar varios hallazgos significativos:

- *FaceNet* es uno de los algoritmos de reconocimiento de rostro más precisos, superando en precisión a algoritmos antiguos como *EigenFace y Face-recognition.*
- *FaceNet* utiliza técnicas de aprendizaje profundo, en comparación con *EigenFace y Face-recognition*, que se basan en técnicas estadísticas tradicionales.

4.2 Ventajas y desventajas de cada algoritmo

A continuación, se enumeran en base a la experimentación las ventajas y desventajas de dichos algoritmos:

ALGORITMO	VENTAJAS	DESVENTAJAS
Eigenface	Es un algoritmo simple y fácil de implementar. Requiere menos datos de entrenamiento.	Tiene una precisión limitada y puede ser fácilmente engañado por imitaciones de rostro. No es capaz de manejar cambios en iluminación y ángulos de la cabeza.
Face-recognition	Es más preciso que *EigenFace*. Puede manejar cambios en iluminación y ángulos de la cabeza.	Requiere una mayor cantidad de datos de entrenamiento. Puede ser sensible a cambios en el aspecto del rostro, como el uso de maquillaje o barba.
FaceNet	Es considerado uno de los algoritmos de reconocimiento de rostro más precisos disponibles actualmente. Tiene una mayor capacidad de generalizar a nuevas imágenes y	Requiere una gran cantidad de datos de entrenamiento y es computacionalmente costoso de implementar.

	personas. Puede manejar una gran variedad de cambios en iluminación y ángulos de la cabeza.	

Tabla 24. Ventajas y desventajas de cada algoritmo

Fuente: Autor

4.3 Resultados de la experimentación

En el capítulo anterior se puede observar en la Tabla 20, los resultados que surgen de la experimentación, donde se contrastan los tres algoritmos elegidos para esta investigación (*EigenFace, Face-recognition y FaceNet*). Se miden tres métricas: Precisión, sensibilidad y valor de referencia, se han encontrado hallazgos significativos para la elección del algoritmo adecuado para el prototipo y que solvente los requerimientos planteados en los objetivos por la investigación.

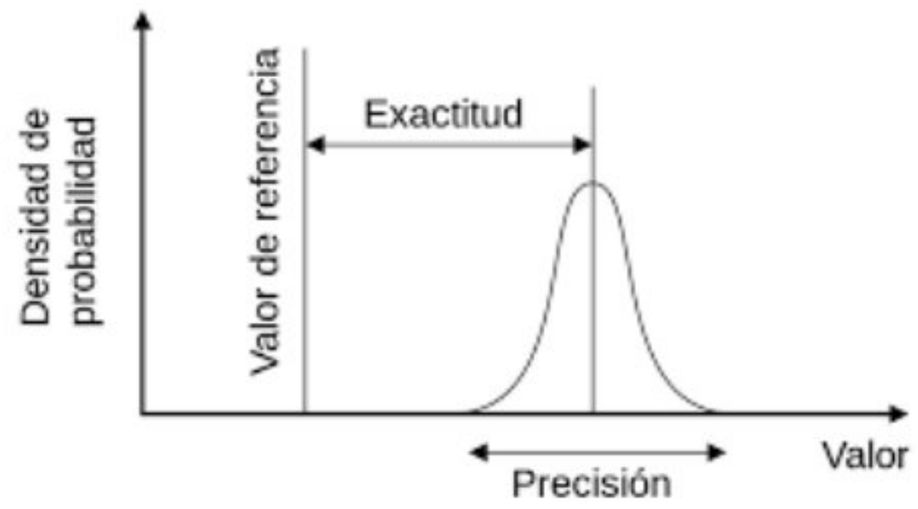

Figura 56. Métricas

Fuente: Autor

4.3.1 Precisión

La precisión se refiere a qué tan acertado es el algoritmo en identificar correctamente los rostros en las imágenes. En este caso, los resultados indican que los tres algoritmos tienen un alto nivel de precisión en la tarea de detección y reconocimiento de rostros, ya que los valores de precisión son muy cercanos a 1 (el valor máximo posible). Así lo muestra la tabla 21, donde se compara la precisión al reconocer a cada estudiante por cada experimento:

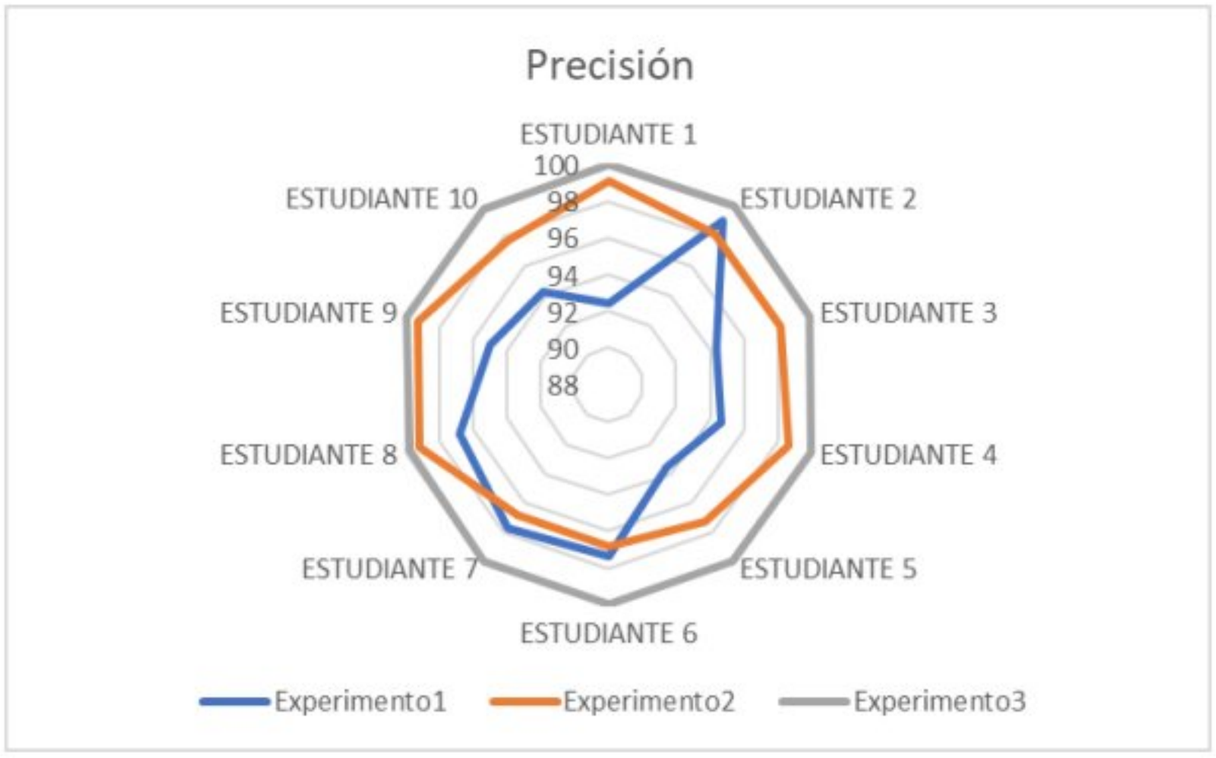

Figura 57. Precisión

Fuente: Autor

Se puede apreciar que el experimento 3 resulta muy preciso al momento de identificar el rostro de cada estudiante con un valor máximo del 100%, el experimento 2 no se queda atrás y su rendimiento en el aspecto de precisión es muy similar al experimento 3, con un máximo de 99.26%. Sin embargo, se puede observar que los resultados del experimento 1 son bajos, pero en cierta medida aceptables porque promedian el 95.5% de precisión general, lo que significa que las predicciones son correctas. El segundo algoritmo tiene una precisión de 0.981, lo que significa que aproximadamente el 98.1% de las predicciones son correctas. Y el tercer algoritmo tiene una precisión de 0.999, lo que significa que aproximadamente el 99.9% de las predicciones son correctas.

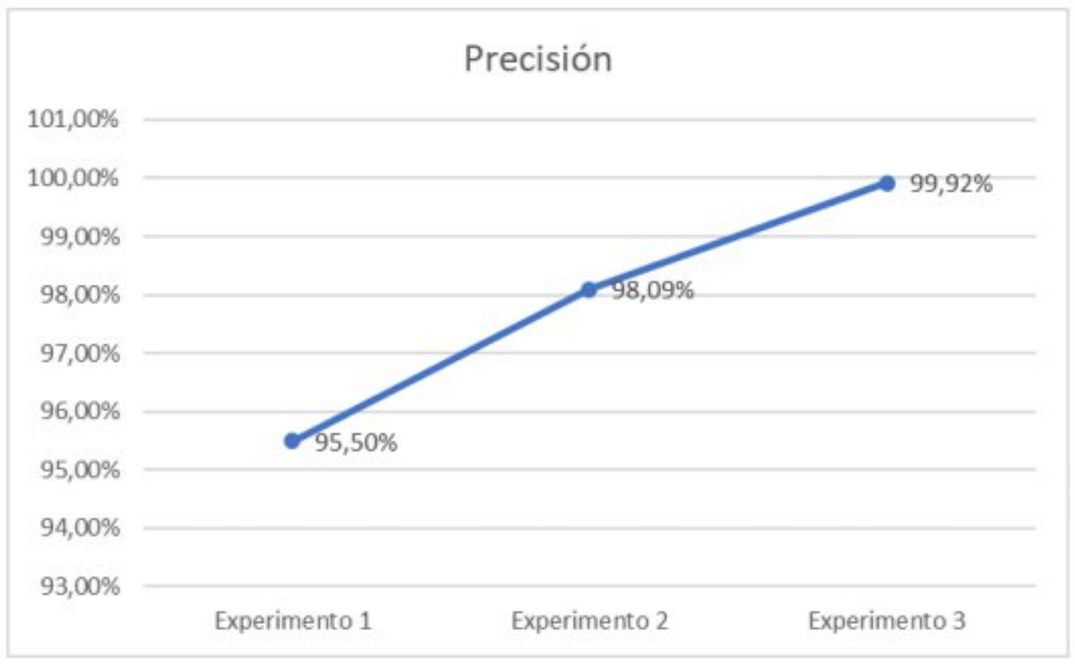

Figura 58. Resultados generales de precisión en base a la Matriz de confusión precisión

Fuente: Autor

En general, estos resultados indican que los tres algoritmos son efectivos en la tarea de detección y reconocimiento de rostros, pero es importante tener en cuenta que otros factores como la velocidad de procesamiento, la complejidad computacional y la capacidad de

generalización también pueden ser relevantes al seleccionar un algoritmo para una aplicación específica.

4.3.2 Sensibilidad (Recall)

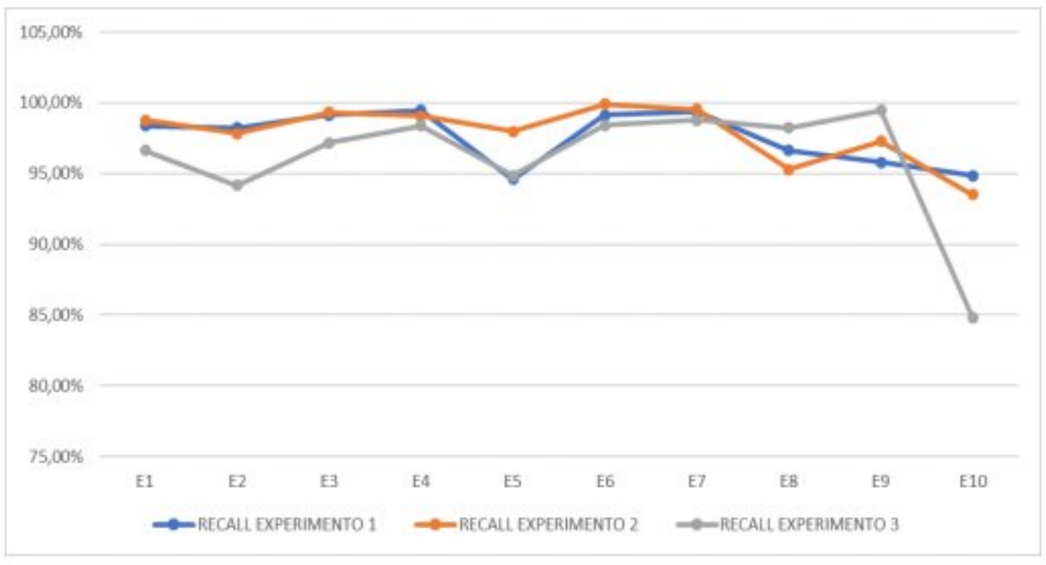

Figura 59. Resultados de la sensibilidad (Recall)

Fuente: Autor

El *recall* mide la capacidad del modelo para identificar todos los casos positivos. En la gráfica 69, se puede observar variaciones considerables entre los tres algoritmos, siendo el estudiante 10 al cual no se lo pudo reconocer de manera efectiva, por ejemplo, en el experimento 3, sin embargo, se considera aceptable ya que alcanza 8.5 que representa un 85.5% de sensibilidad.

4.3.3 Índices *Kappa*

Al comparar los resultados de los índices kappa entre los diferentes experimentos, se puede determinar cuál de los algoritmos es el más preciso y confiable para el reconocimiento facial. Esto es esencial porque los algoritmos de reconocimiento facial se utilizan en una variedad de contextos, como la seguridad e identificación de personas en entornos públicos y privados, y es importante tener un algoritmo que sea preciso y confiable en diferentes condiciones. Según lo muestra la matriz de observación del capítulo anterior, los resultados generales de los índices kappa en cada experimento indican que los tres algoritmos de reconocimiento facial son precisos y tienen un alto grado de concordancia con los evaluadores humanos. Sin embargo, hay una diferencia significativa en los índices *kappa* entre los tres experimentos.

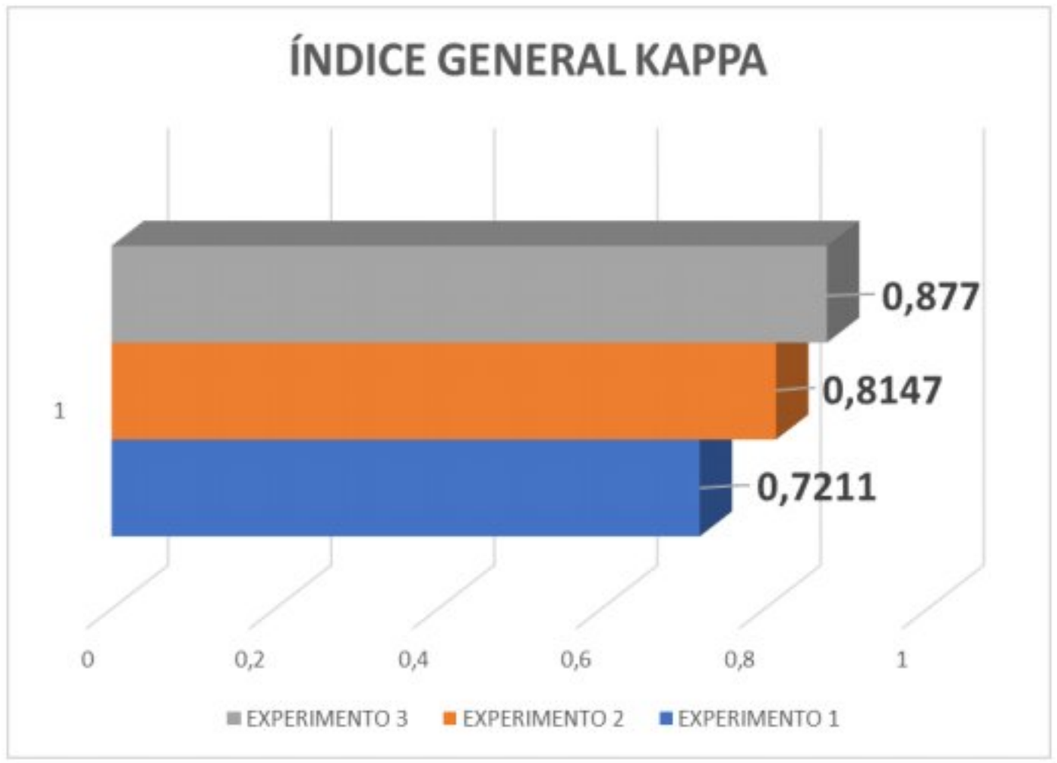

Figura 60. Índices general Kappa

Fuente: Autor

El experimento 3 tuvo la índice kappa más alto (0.877), lo que sugiere que el algoritmo utilizado en este experimento es el más preciso y confiable de los tres algoritmos. En comparación, el experimento 2 también arrojó una índice kappa alto (0.8147), aunque ligeramente más bajo que el experimento 3. El experimento 1, tuvo la índice *kappa* más bajo de los tres (0.7211), lo que indica que el algoritmo utilizado en este experimento puede no ser tan preciso como los otros dos algoritmos. Es importante tener en cuenta que estos resultados generales de los índices kappa pueden no ser representativos de la precisión de los algoritmos en diferentes condiciones, como iluminación, ángulos de la cara u otros factores que puedan afectar el rendimiento del algoritmo en situaciones prácticas.

4.4 Limitaciones

Algunas de las limitaciones más relevantes en la presente investigación son:

- Las condiciones ambientales, como la iluminación, la posición de la cabeza, la distancia de la cámara y la calidad de la imagen, afectan la capacidad del sistema para identificar correctamente a una persona. Esto puede dar lugar a falsos negativos o falsos positivos, lo que puede permitir la suplantación de identidad.
- Las personas pueden cambiar su apariencia facial a través de diferentes métodos, como el uso de maquillaje, pelucas, gafas, barbas, etc. Estos cambios pueden hacer que el sistema de reconocimiento facial no sea capaz de identificar correctamente a la persona, lo que puede permitir la suplantación de identidad.
- La precisión de un sistema de reconocimiento facial depende en gran medida de la calidad y la cantidad de la base de datos de rostros utilizada para entrenar el sistema. Si

la base de datos es limitada o no es representativa de la población objetivo, el sistema puede ser menos preciso y susceptible a errores de identificación.

- Los ataques de *spoofing* son una técnica utilizada para engañar a los sistemas de reconocimiento facial mediante la presentación de imágenes o vídeos falsos o manipulados. Estos ataques pueden permitir la suplantación de identidad y reducir la efectividad del sistema.
- Los sistemas de reconocimiento facial pueden plantear preocupaciones sobre la privacidad y la seguridad de los datos personales. La recopilación y el almacenamiento de imágenes faciales pueden dar lugar a la vulneración de datos personales y la identificación de personas sin su consentimiento.

4.5 Trabajos futuros

En investigaciones futuras, se puede profundizar en el estudio de técnicas para mejorar la precisión y robustez de los sistemas de reconocimiento facial y prevenir la suplantación de identidad. Una posible línea de investigación consiste en el desarrollo de algoritmos de aprendizaje automático más avanzados y complejos, que permitan una mayor capacidad de generalización y detección de ataques de *spoofing*. Además, se puede investigar la integración de múltiples sensores y técnicas biométricas para mejorar la precisión y seguridad de la autenticación de identidad. También es importante abordar las preocupaciones éticas y legales relacionadas con el uso de la tecnología de reconocimiento facial y la protección de la privacidad de los usuarios.

Otra posible área de investigación futura es la evaluación de la efectividad de los sistemas de reconocimiento facial en diferentes escenarios y contextos, como la identificación de individuos en situaciones de poca luz o con cambios en la apariencia facial debido al envejecimiento o la utilización de maquillaje. Además, se puede investigar la implementación de técnicas de interpretabilidad en los algoritmos de reconocimiento facial, para comprender mejor cómo toman decisiones y detectar posibles sesgos o discriminación. Finalmente, se puede explorar la integración de tecnologías emergentes como *blockchain* para mejorar la seguridad y transparencia de los sistemas de reconocimiento facial y garantizar la integridad de los datos biométricos de los usuarios.

Otro aspecto importante a considerar en futuras investigaciones es la seguridad contra ataques que involucran el uso de cámaras virtuales o imágenes/fotografías de la persona que intenta

suplantar la identidad. Esto es particularmente importante porque los sistemas de reconocimiento facial a menudo se basan en la comparación de imágenes en una base de datos con la imagen capturada por la cámara en tiempo real. Por lo tanto, los atacantes pueden intentar engañar al sistema presentando una imagen o video en lugar de la persona real.

Se puede explorar el uso de técnicas de detección de ataques de presentación (PAD) para mejorar la seguridad contra estos tipos de ataques. Estas técnicas pueden incluir la detección de patrones de movimiento o características faciales que solo se pueden observar en una persona real y no en una imagen o video. También es posible investigar la utilización de tecnologías de detección de profundidad, que pueden medir la distancia entre la cámara y la persona para detectar posibles engaños en la representación tridimensional de la cara. En resumen, la investigación futura debe enfocarse en garantizar la seguridad y robustez de los modelos de reconocimiento facial contra todo tipo de ataques, incluyendo aquellos que utilizan cámaras virtuales o imágenes/fotografías.

4.6 Conclusiones

Se ha llevado a cabo una investigación exhaustiva de los fundamentos teóricos del reconocimiento facial, permitiendo la comprensión profunda de los principios y técnicas clave en este campo. Además, se han evaluado y comparado rigurosamente tres algoritmos de reconocimiento facial, midiendo múltiples métricas de precisión y robustez en la detección y verificación de rostros. Como resultado, se ha identificado el algoritmo más adecuado para ser utilizado en el proyecto, mejorando la eficiencia y eficacia de la aplicación de reconocimiento facial propuesta.

Se ha logrado determinar de manera precisa y eficiente si la persona presentada ante el sistema es o no quien afirma ser, identificando posibles suplantaciones de identidad mediante el análisis de características biométricas únicas del rostro y la comparación con registros previos, a través del modelo de reconocimiento facial el cual permite detectar y prevenir la suplantación de identidad con el uso de algoritmos de reconocimiento facial.

Así también, se destaca que *Face-recognition* tiene una menor complejidad computacional y puede ser más eficiente en la identificación de rostros con variaciones en la iluminación y la expresión facial, esto lo convierte en una opción adecuada para utilizar en situaciones que requieren una respuesta inmediata. Por otro lado, los algoritmos basados en redes neuronales,

como *FaceNet*, pueden tener una mayor precisión en el reconocimiento facial, pero su entrenamiento y ajuste pueden ser más complejos.

Para la implementación de un sistema de reconocimiento facial que se adapte a las condiciones de operación, es necesario analizar las ventajas y limitaciones de cada algoritmo y seleccionar el que mejor se adapte a las necesidades del proyecto. Por lo tanto, en la presente tesis se justifica el uso de *Face-recognition* debido a su simplicidad y facilidad de implementación en comparación con otros algoritmos de reconocimiento facial más avanzados, como *FaceNet*.

Finalmente, se ha evaluado la eficacia y eficiencia del modelo en diferentes situaciones y condiciones de captura de imagen, lo que garantiza su robustez y aplicabilidad en contextos reales. Se concluye que el modelo de reconocimiento facial desarrollado tiene un alto potencial para mejorar la seguridad en entornos remotos de toma de evaluaciones y prevenir la suplantación de identidad.

4.7 Recomendaciones

- Se recomienda realizar pruebas adicionales para evaluar la precisión y confiabilidad de los algoritmos en diferentes condiciones. De esta manera, se podrán tomar decisiones más informadas al elegir un algoritmo de reconocimiento facial para su implementación en un entorno práctico.
- Es necesario llevar a cabo pruebas del prototipo en entornos reales para evaluar su efectividad. Se sugiere explorar la posibilidad de implementar el modelo de reconocimiento facial en instituciones educativas o empresas que requieran la verificación de la identidad de sus usuarios en línea.
- Aunque se logró un alto nivel de precisión en el reconocimiento facial, se recomienda investigar cómo mejorar aún más la eficacia del algoritmo. Se sugiere evaluar diferentes técnicas de preprocesamiento de imágenes, la selección de características más adecuadas y la optimización de los hiper parámetros de los algoritmos de IA.
- En el contexto de la supervisión remota de exámenes en línea, es importante proteger la privacidad de los usuarios. Se sugiere investigar cómo utilizar técnicas de enmascaramiento facial para proteger la identidad de los usuarios mientras se verifica su identidad.

- Se sugiere realizar una evaluación de la aceptabilidad para determinar si los usuarios estarían dispuestos a utilizar la tecnología de reconocimiento facial en la toma de exámenes en línea y qué factores podrían influir en su percepción.
- Además de la supervisión remota de exámenes en línea, existen muchas otras aplicaciones potenciales de la tecnología de reconocimiento facial en entornos virtuales. Se sugiere investigar cómo la tecnología podría ser utilizada para mejorar la seguridad en la banca en línea, el comercio electrónico y otras aplicaciones en línea donde se necesita verificar la identidad de los usuarios.

Bibliografía

[1] Ministerio de Trabajo de Ecuador, "Acuerdo Ministerial Nro. MDT-2020-076," 12 de marzo, 2020, [En línea]. Disponible: https://www.trabajo.gob.ec/wp-content/uploads/2020/03/ACUERDO-MDT-2020-076-TELETRABAJO.pdf?x42051

[2] Secretaría Nacional de Gestión de Riesgos, "Del aislamiento al distanciamiento documento explicativo para el funcionamiento del semáforo mes de mayo de 2020 de Ecuador," 2020, Available: https://www.gestionderiesgos.gob.ec/wp-content/uploads/2020/05/Del-aislamiento-al-distanciamiento_Documento-Explicativo-para-el-Funcionamiento-del-Sem%C3%A1foro_Mayo-2020.pdf.

[3] N. CEPAL and UNESCO, "La educación en tiempos de la pandemia de COVID-19," REPOSITORIO DIGITAL - Comisión Económica para América Latina y el Caribe, 2020.

[4] I. A. Guevara Bazán, J. Martínez Cortés, and A. A. Landa Alemán, "La adaptación tecnológica en la educación: una situación emergente," Revista RedCA, vol. 3, no. 8, pp. 49-61, Oct. 2020. DOI: 10.36677/redca.v3i8.15462.

[5] K. Ala-Mutka, Y. Punie, and C. Redecker, "Digital competence for lifelong learning," Madrid, España: JRC European Commission, 2008.

[6] UNESCO. (2020, abril 1). Propuestas de la UNESCO para garantizar la educación online durante la pandemia. Educaweb. Recuperado de https://www.educaweb.com/noticia/2020/04/01/propuestas-unesco-garantizar-educacion-online-pandemia-19132/

[7] González Halcones, M. A. and Pérez González, N. (s.f.). La evaluación del Proceso de Enseñanza-Aprendizaje. Fundamentos Básicos. Recuperado el 13 de abril de 2021, de https://ruidera.uclm.es/xmlui/bitstream/handle/10578/7951/La_evaluaci_n_del_proceso_de_ense_anza-aprendizaje.pdf?sequence=1&isAllowed=y

[8] J. Carstairs and B. Myors, "Internet testing: A natural experiment reveals test score inflation on a high-stakes, unproctored cognitive test," Computers in Human Behavior, vol. 25, no. 3, pp. 738-742, 2009.

[9] A. Friedman, I. Blau, and Y. Eshet-Alkalai, "Cheating and feeling honest: Committing and punishing analog versus digital academic dishonesty behaviors in higher education," Interdisciplinary Journal of e-Skills and Life Long Learning, vol. 12, pp. 193-205, 2016.

[10] S. Poornima, N. Sripriya, B. Vijayalakshmi, and P. Vishnupriya, "Sistema de monitoreo de asistencia utilizando reconocimiento facial con salida de audio y clasificación de género," in Conferencia Internacional sobre Computación, Comunicación y Procesamiento de Señales: Enfoque Especial en IoT, ICCCSP 2017, 2017, pp. 0-4. https://doi.org/10.1109/ICCCSP.2017.7944103

[11] M. Sajid, R. Hussain, and M. Usman, "Un modelo conceptual para el sistema automatizado de marcado de asistencia utilizando el reconocimiento facial," 2014 9a Conferencia Internacional sobre Gestión de la Información Digital, ICDIM 2014, pp. 7-10, 2014, doi: 10.1109/ICDIM.2014.6991407.

[12] Noguera, C. G. (2012). Autenticación por reconocimiento facial para aplicaciones web, utilizando software libre. (Tesis doctoral). Universidad Pontificia Bolivariana.

[13] E. Hjelmas and B.K. Low, "Face detection: A survey," Computer Vision and Image Understanding, vol. 83, no. 3, pp. 236-274, Sep. 2001.

[14] Yang, M. H., Kriegman, D. J., & Ahuja, N. (2002). "Detecting faces in images: A survey." IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(1), 34-58.

[15] A. Rama and F. Tarrés, "Un nuevo método para la detección de caras basado en integrales difusas," Universidad Politécnica de Catalunya, Barcelona, España.

[16] Hernandez, O. J., & Kleiman, M. S. (2005). "Face recognition using multispectral random field texture models, color content, and biometric features." En Applied Image Pattern Recognition Workshop, pp.204-209.

[17] S. B. Lee and S. Tsutsui, "Intelligent biometric techniques in fingerprint and face recognition," Boca Raton, FL, USA: CRC Press, Inc., 1999.

[18] E. Hjelmas, "Biometric Systems: A Face Recognition Approach," 2000.

[19] S. M. V. Palacios, "Sistema de reconocimiento de reconocimiento de rostros," Universidad Peruana de Ciencias Aplicadas (UPC).

[20] R. C. González and R. E. Woods, "Digital Image Processing," 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006.

[21] J. C. Russ, "The image processing handbook", 3rd ed. Boca Raton, FL, USA: CRC Press, Inc., 1999.

[22] Z. Huang, S. Shan, R. Wang, H. Zhang, and S. Lao, "A Benchmark and Comparative Study of Video-Based Face Recognition on COX Face Database," IEEE Transactions on Image Processing, vol. 24, no. 12, pp. 5967-5981, 2015.

[23] Guerrero, D. (2012). Reconocimiento Facial: Pasado, Presente y Futuro. Recuperado el 17 de marzo de 2023, de http://www.diegoguerrero.info/reconocimiento-facial-pasado-presente-y-futuro/

[24] J. S. Bruner and R. Tagiuri, "The perception of people", Technical report, DTIC Document, 1954.

[25] S. Tikoo and N. Malik, "Detection, Segmentation and Recognition of Face and its Features using Neural Network," International Journal of Advanced Research in Computer and Communication Engineering, vol. 5, no. 6, pp. 130-136, 2016.

[26] Nixon, M. (1985). "Eye spacing measurement for facial recognition." En 29th Annual Technical Symposium, páginas 279–285. International Society for Optics and Photonics.

[27] T. Sakai, M. Nagao, and T. Kanade, "Computer analysis and classification of photographs of human faces," in Proc. First USA—Japan Computer Conference, 1972, p. 2.7.

[28] Kanade, T. (2013). Computer recognition of human faces. Available at: https://Computer_recognition_of_human_faces.html

[29] Bruce, K. T. Lucas. An Iterative Image Registration Technique with an Application to Stereo Vision. Proceedings of Imaging Understanding Workshop, Pennsylvania, 1981, pp. 121-130.

[30] Biosys, "Biometria de reconocimiento facial," Obtenido de Biosys web site: http://www.biosys.es/sistemas-biometricos/facial, 2018.

[31] Gonzalez Ferreiro, M. (s.f.). "Reconocimiento facial combinando técnicas 2D y 3D." Recuperado de http://es.scribd.com/doc/28778540/11/Historia-del-reconocimiento-facial.

[32] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 1991.

[33] P. Viola and M. Jones, "Robust real time object detection," in IEEE ICCV Workshop on Statistical and Computational Theories of Vision, Vancouver, 2001.

[34] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in Scandinavian Conference on Image Analysis, 2003, pp. 363-370, Springer.

[35] P. Armstrong and M. Thompson, "Examining exam integrity: A review of research on academic dishonesty in online courses," The Internet and Higher Education, vol. 21, pp. 1-10, 2014.

[36] L. Lu and Y. Li, "A survey of cheating detection techniques in online testing," Educational Technology & Society, vol. 20, no. 4, pp. 52-65, 2017.

[37] R. Paul and L. Elder, "Examining the role of student characteristics, learning strategies, and motivation in online cheating," The Journal of Higher Education, vol. 75, no. 2, pp. 161-186, 2004.

[38] Wang, S. and Newby, T. J. (2013). "A review of cheating in online courses and strategies to prevent it." The Internet and Higher Education, 19, 1-10.

[39] J. Izaguirre Olmedo and F. León Gavilánez, "Análisis de los ciberataques realizados en América Latina," INNOVA Research Journal, vol. 3, no. 9, pp. 172-181, 2018.

[40] A. Luma, "Real Time Access Control Based on Face Recognition," 10.15242/IAE.IAE0615004, 2015. Disponible en: https://www.researchgate.net/publication/284344915_Real_Time_Access_Control_Based_on_Face_Recognition.

[41] Lopez Perez, N., & Toro Agudelo, J. J. (2012). Técnicas de biometría basadas en patrones faciales del ser humano. Tesis. Ing. Sistemas y Computación. recursosbibliotecautp.edu.co.

[42] CES 2008: "Face Detection New Trend for Camcorders", [Online]. Available: http://www.camcorderinfo.com/content/CES-2008-Face-Detection-a-New-Trend-for-Camcorders-34250.htm (in English), Jan. 2008.

[43]L. F. Arboleda-Monsalve, A. M. Gómez-Pulgarín, and L. E. Fierro-González, "Facial recognition system for secure mobile banking in Latin America," in Proceedings of the 2018 IEEE Latin American Conference on Computational Intelligence (LA-CCI), 2018, pp. 1-6. doi: 10.1109/LA-CCI.2018.8641718.

[44] Gonzalez, M. (2006). "Reconocimiento de iris." Trabajo de grado - Ing. Inform., pp. 1–12.

[45] R. G. Elías, "2.300 inocentes marcados como criminales en una noche, el problema del reconocimiento facial -Omicrono," 2018. [Online]. Available: https://omicrono.elespanol.com/2018/05/reconocimiento-facial-policia-falsos-positivos/. [Accessed: 20-Feb-2019].

[46] T. Evelyn, "Banco Guayaquil lanza sistema de reconocimiento facial para transacciones," Revista Líderes, 2015. [Online]. Available: https://www.revistalideres.ec/lideres/banco-guayaquil-lanza-sistema-reconocimiento.html. [Accessed: 20-Feb-2019].

[47] A. Hussain, S. Raza, and F. Rehman, "Mobile banking using facial recognition: A survey of current trends and future directions," in Proceedings of the 2019 3rd

International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), 2019, pp. 1-6. doi: 10.1109/iCoMET.2019.8869126.

[48] M. Jurado and A. Garces, "Sistema de reconocimiento facial con visión artificial para apoyar al ECU 911 con la identificación de personas en la lista de los más buscados," Universidad Técnica de Ambato, 2017. [Online]. Available: https://repositorio.uta.edu.ec/jspui/handle/123456789/24490. [Accessed: Month Day, Year].

[49] S. Chintalapati and M. V. Raghunadh, "Sistema automatizado de gestión de asistencia basado en algoritmos de reconocimiento facial," in 2013 Conferencia Internacional IEEE sobre Inteligencia Computacional e Investigación Informática, 2013, pp. 1-5. https://doi.org/10.1109/ICCIC.2013.6724266

[50] V. Hidalgo and R. Nassib, "Control de Asistencia a Clases Mediante Procesamiento Digital de Imágenes. Propuesta de un Prototipo que Reconozca los Alumnos de un Paralelo," 2017.

[51] J. D. P. Morcote, "Reconocimiento facial en tiempo real orientado a videollamadas o live stream para autenticar identidades durante una audiencia legal," IEEE, 2020.

[52] F. Serratosa, "La biometría para la identificación de las personas," 2013.

[53] J. Quevedo Gonzalez, "Investigacion y prueba de ciberdelito," U. Barcelona, Barcelona-España, 2017.

[54] T. Anderson, “The Theory and Practice of Online Learning”. Athabasca University, 2nd ed. Ed. AU Press. 2009. Biometrics Glossary, fecha de acceso: 20/01/2012, http://www.biometrics.gov/Documents/Glossary.pdf.

[55] L. R. Giraldo de la Caridad and V. B. Silvia Margarita, "La inteligencia artificial en la educación superior. Oportunidades y amenazas," Guayaquil, UIDE-INNOVA, 2017.

[56] Statista. "Ingresos del mercado global de software de IA por región", Disponible en: https://es.statista.com/estadisticas/1128042/ingresos-del-mercado-global-de-software-de-ia-por-region/, [Fecha de acceso: 23 Mayo 2023].

[57] S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach," 3rd ed. Upper Saddle River, NJ, USA: Pearson Education, 2010.

[58] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). New York: Springer.

[59] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.

[60] D. Forsyth and J. Ponce, "Computer Vision - A Modern Approach," Pearson, 2012.

[61] I. García S. and V. Caranqui S., "La Visión Artificial y los Campos de Aplicación," Revista Digital "Tierra Infinita", pp. 1-9, 2015. [Online]. Available: https://revistasdigitales.upec.edu.ec/index.php/tierrainfinita/article/view/76. [Accessed: Feb. 2021].

[62] R. Szeliski, "Computer vision: Algorithms and applications," New York, NY, USA: Springer, 2010.

[63] D. Forsyth and J. Ponce, "Computer vision: A modern approach," 2nd ed. Upper Saddle River, NJ, USA: Pearson Education, 2011.

[64] L. J. Sandoval Serrano, "Algoritmos de aprendizaje automático para análisis y predicción de datos," Editorial ITCA Editores, 2018, ISSN: 2072-568X, http://hdl.handle.net/10972/3626.

[65] Zhou, Z.-H. (2017). "A brief introduction to weakly supervised learning." National Science Review, vol. 5, no. 1, pp. 44-53.

[66] V. Autores, "Introduction to unsupervised learning", Abril 9, 1028. [Online]. Disponible en: https://algorithmia.com/blog/introduction-to-unsupervised-learning.

[67] Zhu, X. and Goldberg, A. B. (2009). "Introduction to semi-supervised learning." Synthesis lectures on artificial intelligence and machine learning, vol. 3, no. 1, pp. 1-300.

[68] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," Robotica, vol. 29, no. 3, pp. 438-441, 2011.

[69] E. Blanzieri and A. Bryl, "A survey of learning-based techniques of email spam filtering," Artificial Intelligence Review, vol. 29, no. 1, pp. 63-92, 2008.

[70] E. Borovikov, "A survey of modern optical character recognition techniques," arXiv preprint arXiv:1412.4183, 2014.

[71] Huang, D., & He, H. (2014). Recent advances in face recognition. In Advances in biometrics (pp. 317-358). Springer..

[72] Viola, P., & Jones, M. J. (2004). Robust real-time face detection. International Journal of Computer Vision, 57(2), 137-154.

[73] H. Li and Y. Hu, "A survey of face detection methods," International Journal of Image and Graphics, vol. 12, no. 2, pp. 185-212, 2012.

[74] Yang, M.-H., Kriegman, D. J., & Ahuja, N. (2002). "Detecting faces in images: a survey." IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(1), 34-58.

[75] Zafeiriou, S., Zhang, C., and Zhang, Z. (2015). A survey on face detection in the wild. Comput. Vis. Image Underst., 138(C):1–24.

[76] Viola, P. and Jones, M. J. (2001). "Rapid object detection using a boosted cascade of simple features." In Proc. IEEE Conf. Computer Vision and Pattern Recognition (pp. 511-518).

[77] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2005, pp. 886-893.

[78] Liu, C., Yuen, P. W., & Torralba, A. (2016). Deep learning face attributes in the wild. En Proc. IEEE Conf. Computer Vision and Pattern Recognition (pp. 5306-5314).

[79] Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 23(10), 1499-1503.

[80] K. Sung, T. Kim, and K. M. Lee, "Facial feature mask: A simple yet effective method for facial feature detection," Image and Vision Computing, vol. 33, pp. 91-99, 2015.

[81] M. Castrillon, O. Deniz, C. Guerra, and M. Hernandez, "Encara2: Real-time detection of multiple faces at different resolutions in video streams," Journal of Visual Communication and Image Representation, vol. 18, no. 2, pp. 130-140, 2007.

[82] Kruppa, H., Schwarz, M. C., & Schiele, B. (2003). Fast and robust face finding via local context. En Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (pp. 1-8). IEEE.

[83] M. Castrillon Santana, J. Lorenzo Navarro, O. Deniz Suarez, J. Isern Gonzalez, and A. Falcon Martel, "Multiple Face Detection at Different Resolutions for Perceptual User Interfaces," in Proceedings of the International Conference on Computer Vision and Graphics, 2005, pp. 445-452.

[85] Habibi Aghdam, H., & Jahromi, E. J. (2017). Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification. Springer International Publishing.

[86] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

[87] S. Silva and E. Freire, "Redes Neuronales Convolucionales," Medium, 23-Nov-2019. [Online]. Available: https://medium.com/@bootcampai/redes-neuronales-convolucionales-5e0ce960caf8. [Accessed: 10-Oct-2020].

[88] M. Pietikainen, A. Hadid, G. Zhao, and T. Ahonen, "Computer Vision Using Local Binary Patterns," Springer-Verlag London, 2011.

[89] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, Jul. 2002.

[90] Liao, S., Jain, A. K., & Li, S. Z. (2014). A fast and accurate unconstrained face detector. CoRR, abs/1408.1656.

[91] Turk, M. A., & Pentland, A. P. (1991). "Face recognition using eigenfaces." En Computer Vision and Pattern Recognition, Proceedings CVPR '91., IEEE Computer Society Conference.

[92] I. T. Jolliffe, Principal Component Analysis, Springer Verlag, 1986.

[93] S. Srinivasan, A. Gopalakrishnan, and R. Balasubramanian, "Facial recognition using eigenfaces and support vector machines," in Proceedings of the 2018 International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, pp. 764-768. doi: 10.1109/ICCONS.2018.8663273.

[94] R. Saha and B. Bhattacharjee, "Faces Recognition Using EigenFaces," International Journal of Emerging Technology and Advanced Engineering, vol. 3, no. 3, pp. 90-93, 2013.

[95] Lorente, L. (1998). Representación de Caras mediante EigenFaces. Escola Técnica Superior d'Enginyers de Telecomunicación de Barcelona, 11(14), 13-20.

[96] L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, pp. 5-32, 2001.

[97] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). New York, NY: Springer.

[98] G. James, D. Witten, T. Hastie, and R. Tibshirani, An Introduction to Statistical Learning, vol. 112. New York: Springer, 2013.

[99] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," ACM Transactions on Intelligent Systems and Technology (TIST), vol. 2, no. 3, pp. 1-27, 2011.

[100] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015..

[101] C. M. Bishop, "Neural Networks for Pattern Recognition," Oxford, UK: Oxford University Press, 1995.

[102] F. Chollet, Deep Learning with Python, 1st ed. Nov. 2017.

[103] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85-117, 2015.

[104] I. Goodfellow, Y. Bengio, and A. Courville, "Deep learning," Cambridge, MA, MIT Press, 2016.

[105] S. Huber, H. Wiemer, D. Schneider, and S. Ihlenfeldt, "DMME: Data mining methodology for engineering applications – a holistic extension to the CRISP-DM model," Procedia CIRP, vol. 79, pp. 403-408, Jan. 2019.

[106] Y. Sun, X. Wang, and X. Tang, "Exploratory Face Recognition with Deep Learning," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[107] Z. Liu, P. Luo, X. Wang, and X. Tang, "Exploratory Study of Deep Learning for Face Recognition," in IEEE Transactions on Image Processing, vol. 26, no. 11, pp. 5012-5024, Nov. 2017.

[108] S. Zhang, L. Chen, and X. Liu, "Correlational Study of Face Recognition Algorithms," in IEEE Transactions on Image Processing, vol. 27, no. 6, pp. 2993-3004, Jun. 2018.

[109] X. Wang, Y. Sun, and X. Tang, "A Correlational Study of Face Recognition using Deep Learning," in IEEE Transactions on Image Processing, vol. 26, no. 11, pp. 5277-5291, Nov. 2017.

[110] Hernández R.; Fernández, C. & Baptista, P. (2003) Metodología de la investigación, 3. ed. México D.F.: McGraw-Hill. 705 pp.

[111] S. Zhang, L. Chen, and X. Liu, "Experimental Study of Face Recognition Algorithms," IEEE Transactions on Image Processing, vol. 27, no. 6, 2018.

[112] Liu, Z., Luo, P., Wang, X., & Tang, X. (2017). An experimental study of deep learning for face recognition. IEEE Transactions on Image Processing, 26(11), 5126-5138.

[113] W. J. Scheirer, T. E. Boult, and R. J. Boyle, "Discrimination-error tradeoff in face recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 6, pp. 603-620, 1997.

[114] A. K. Jain, P. Flynn, and A. Ross, "Handbook of biometric identification", New York: Wiley, 1997.

[115] G. Piatetsky, "Data Mining Methodology: Poll Results," KDnuggets, 2004. [Online]. Available: https://www.kdnuggets.com/polls/2004/data_mining_methodology.htm

[116] G. Piatetsky, "Data Mining Methodology: Poll Results," KDnuggets, 2007. [Online]. Available: https://www.kdnuggets.com/polls/2007/data_mining_methodology.htm

[117] A. Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems," O'Reilly Media, Inc., 2019.

[118] J. M. S. Muñoz, "Análisis de Calidad Cartográfica mediante el estudio de la Matriz de Confusión," Pensamiento matemático, vol. 6, no. 2, pp. 9-26, 2016.

[119] J. Cohen, "A Coefficient of Agreement for Nominal Scales," Educ. Psychol. Meas., vol. 20, no. 1, pp. 37-46, Apr. 1960.

[120] W. Al-Fares, "Historical Land Use/Land Cover Classification Using Remote Sensing," Springer Cham Heidelberg New York Dordrecht London, 2013.

[121] J. Pérez, J. Díaz, J. Garcia-Martin, and B. Tabuenca, "Systematic literature reviews in software engineering—enhancement of the study selection process using Cohen's Kappa statistic," J. Syst. Softw., vol. 168, p. 110657, Oct. 2020.

[122] J. Cerda L, L. Villarroel del P, and Others, "Evaluación de la concordancia interobservador en investigación pediátrica: Coeficiente de Kappa," Rev. Chil. Pediatr., pp. 54-58, 2008.

[123] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," Biometrics, vol. 33, no. 1, pp. 159-174, Mar. 1977.

[124] X. Liu and X. Wang, "A Comparative Study of Deep Learning and Traditional Algorithms for Face Recognition," International Journal of Pattern Recognition and Artificial Intelligence, vol. 32, no. 08, p. 1840001, 2018.

[125] A. Khanna and A. Ross, "Face recognition using TensorFlow," in 2018 International Conference on Computer Vision (ICCV), 2018, pp. 2267-2275. doi: 10.1109/ICCV.2017.243.

[126] V. Ojansivu and H. He, "FaceNet: A unified embedding for face recognition and clustering," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019, pp. 0-0. doi: 10.1109/CVPRW.2019.00087.

[127] Shao, X., & Li, Z. (2017). A comparative study of face recognition algorithms using PCA, LDA, ICA, and kernel methods. International Journal of Advanced Intelligence Paradigms, 9(3), 195-209.

[128] R. Chellappa, A. K. Jain, "Face Recognition: A Literature Survey", ACM Computing Surveys, vol. 35, no. 4, pp. 399-458, 2003.

[129] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification", IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701-1708, 2014.

[130] Gao, S., Chen, J., Chen, J., He, M., & Cao, J. (2020). A study on facial emotion recognition based on FaceNet and SVM. Journal of Physics: Conference Series, 1578(2), 022022.

[131] García, L. A., López, J. C., Rosales, C. G., & Rincón, D. L. (2021). Drowsiness Detection System for Drivers Based on Face Recognition. IEEE Latin America Transactions, 19(6), 924-930.

[132] Wang, Q., Cheng, Y., & Zhang, Y. (2020). An Assistive System for Visually Impaired People Based on Face Recognition. In 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC) (pp. 395-400). IEEE.

[133] Davis, E. (2015). Face Recognition with Dlib. O'Reilly Media, Inc.

[134] King, D. E. (2009). Dlib-ml: A machine learning toolkit. Journal of Machine Learning Research, 10, 1755-1758.

[135] F. Schroff, D. Kalenichenko, J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," IEEE Conference on Computer Vision and Pattern Recognition, pp. 815-823, 2015.

[136] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A Discriminative Feature Learning Approach for Deep Face Recognition," European Conference on Computer Vision, pp. 499-515, 2016.

[137] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., ... & Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. Proceedings of the 22nd ACM international conference on Multimedia, pp. 675-678.

[138] Shi, X., Chen, Z., Wang, H., Yeung, D., Wong, W. K., & Woo, W. C. (2016). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. Advances in neural information processing systems, pp. 802-810.

Anexos

Anexo 1

Panel del examen en línea

PERFIL ESTUDIANTE

El estudiante podrá visualizar el módulo correspondiente al o los exámenes que tenga asignado.

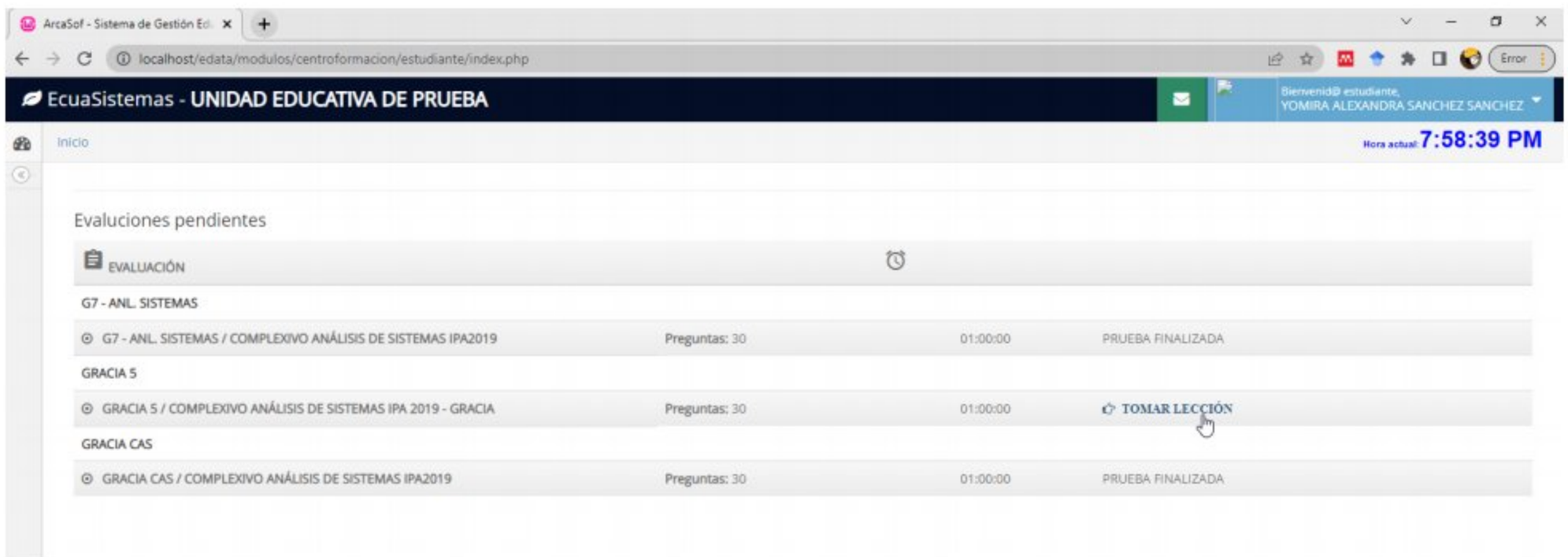

Figura 61. Perfil del estudiante.

Una vez elegido el examen correspondiente el sistema le solicita que se mantenga frente a una webcam para poder realizar el entrenamiento con su rostro. El sistema realizará 600 capturas de rostro en este paso. Por lo que se le solicita al estudiante que mantenga fija su mirada en la cámara.

Figura 62. Captura del rostro del estudiante.

En el siguiente paso, se le solicita al estudiante que coloque su rostro en diferentes posiciones, con el propósito de suministrar diferentes tomas del rostro. El proceso se cumple cuando se han alcanzado 600 tomas de su rostro. Para continuar al siguiente paso, el sistema habilita el botón *Continuar*.

Figura 63. Captura y entrenamiento del modelo.

Por último, se le solicita la contraseña del examen, la cual debe ser suministrada por el docente a cargo de la evaluación.

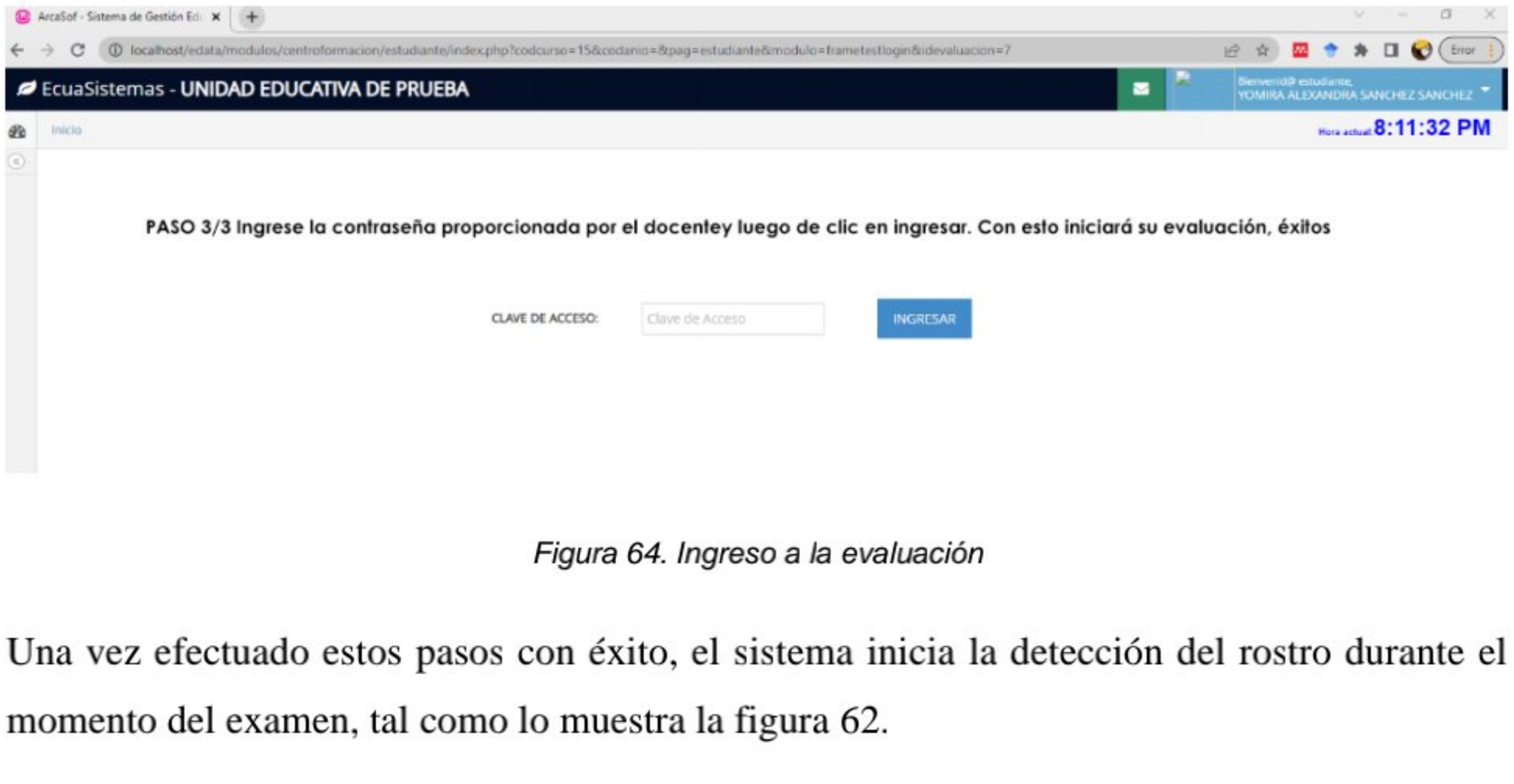

Figura 64. Ingreso a la evaluación

Una vez efectuado estos pasos con éxito, el sistema inicia la detección del rostro durante el momento del examen, tal como lo muestra la figura 62.

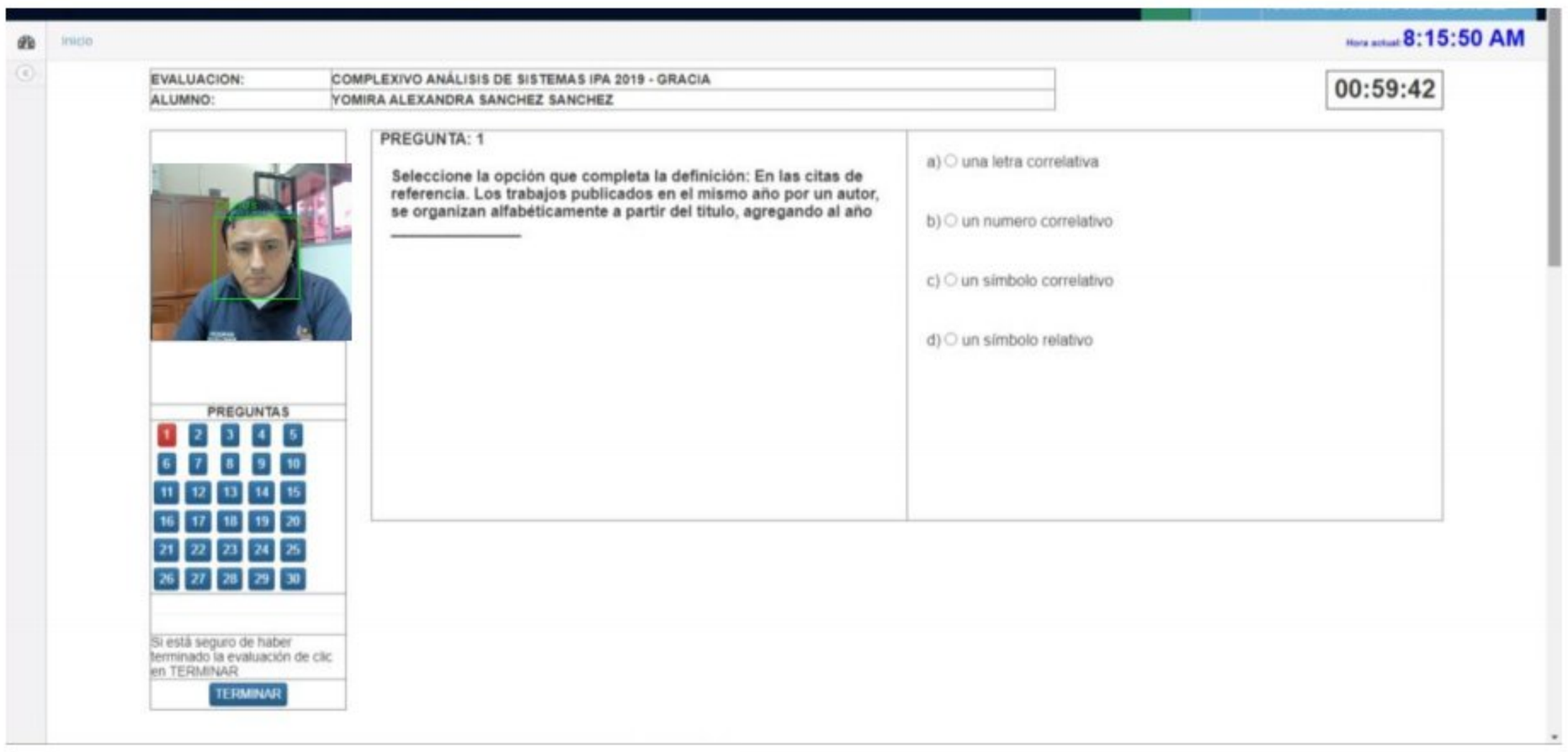

Figura 65. Entorno del examen en línea

Anotaciones del proceso

- Si el estudiante no es detectado durante 10 segundos el sistema registra en la base de datos, un evento de no presencia, el cual es informado al docente en tiempo real y puede ser revisado en el monitor de eventos en el perfil Docente.
- Si el sistema detecta uno o más rostros, el sistema registra en la base de datos un evento de suceso inusual durante el proceso.
- Para evitar que el estudiante implemente artimañas que puedan ocasionar deshonestidad académica y que permitan suplantar la identidad del mismo, a través de cámaras virtuales (OBS, o grabaciones), el sistema mostrará solicitudes aleatorias cada determinado tiempo. Por ejemplo: mostrar su mano mostrando un número aleatorio,

este proceso es capturado y comparado a través del uso de un algoritmo que detecte la acción, si no es correcto o si no hay interacción con lo solicitado, el sistema registra una novedad, la cual puede ser revisada por el docente en su monitor de eventos en el perfil Docente. En el anexo CC1 se puede revisar el código en python usado para este cometido.

Perfil Docente

El docente cuenta con un panel donde administra varios aspectos académicos, entre ellos un monitor de eventos por estudiante, como lo muestra la figura 48, en donde puede observar los tiempos de finalización de la prueba y el puntaje obtenido en la misma, así también, las novedades y los "supuestos" que ha registrado el sistema de reconocimiento facial durante el proceso evaluativo, figura 63.

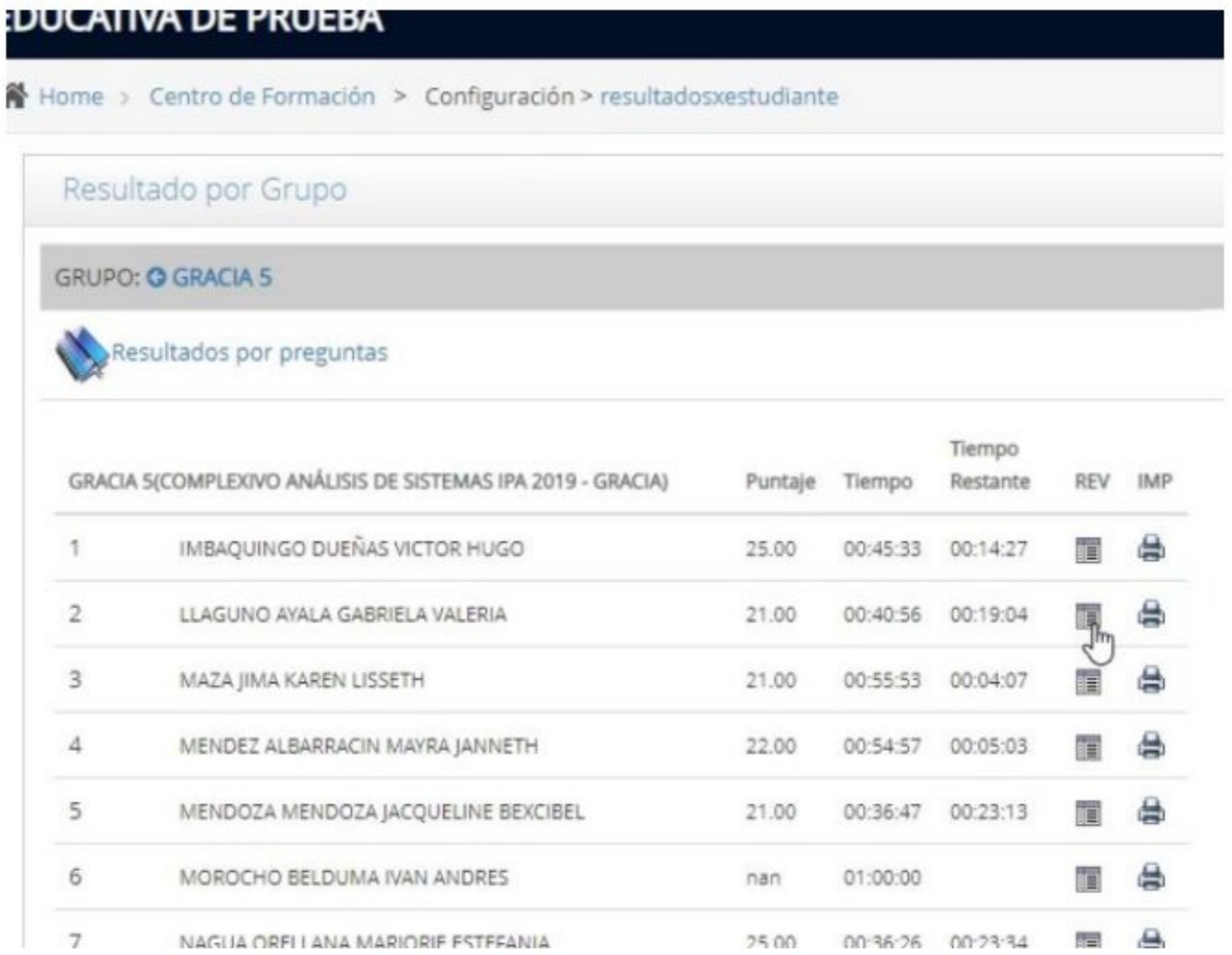

Figura 66. Panel administrativo - Calificación, tiempos, revisión

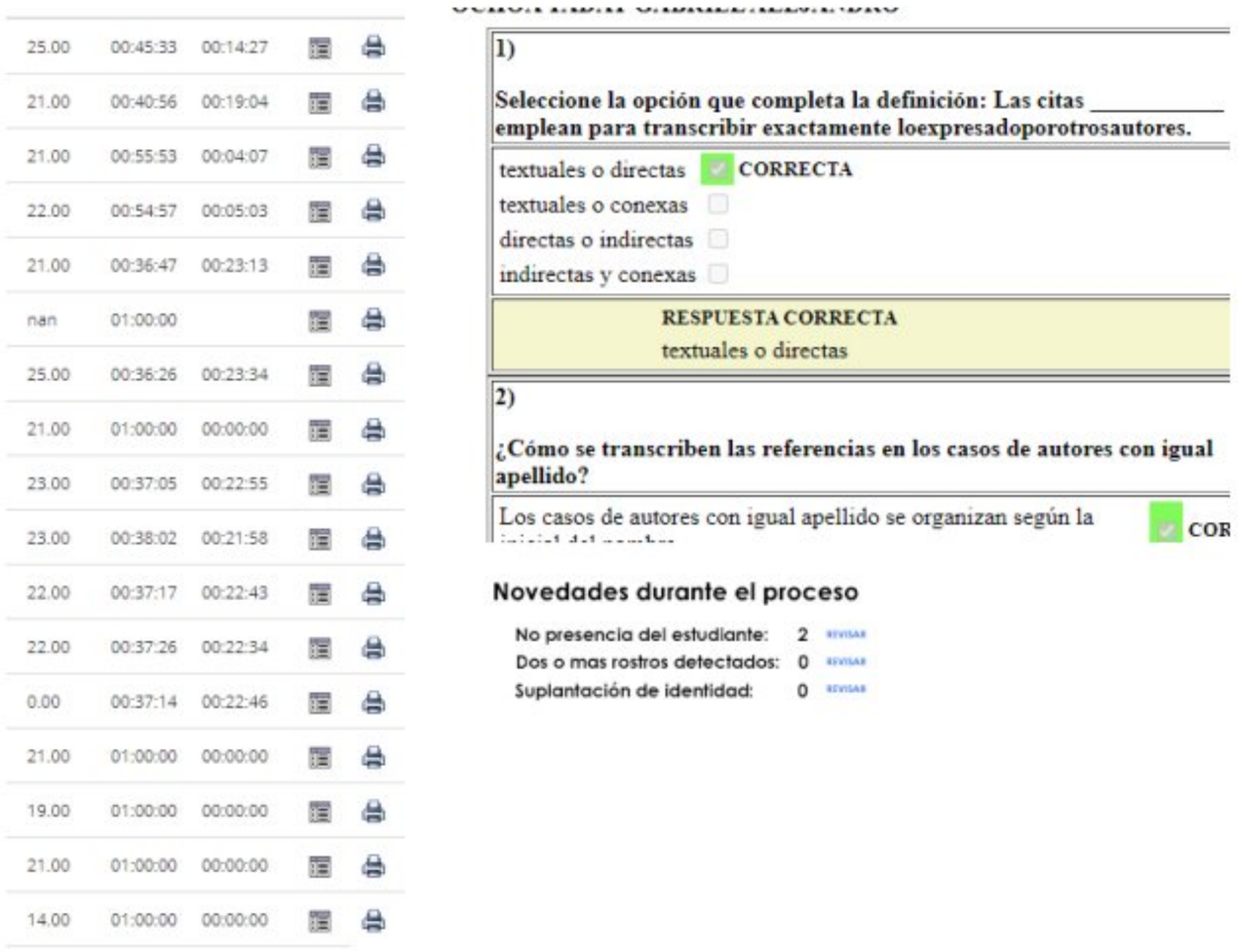

Figura 67. Panel Administrativo eventos y resultados

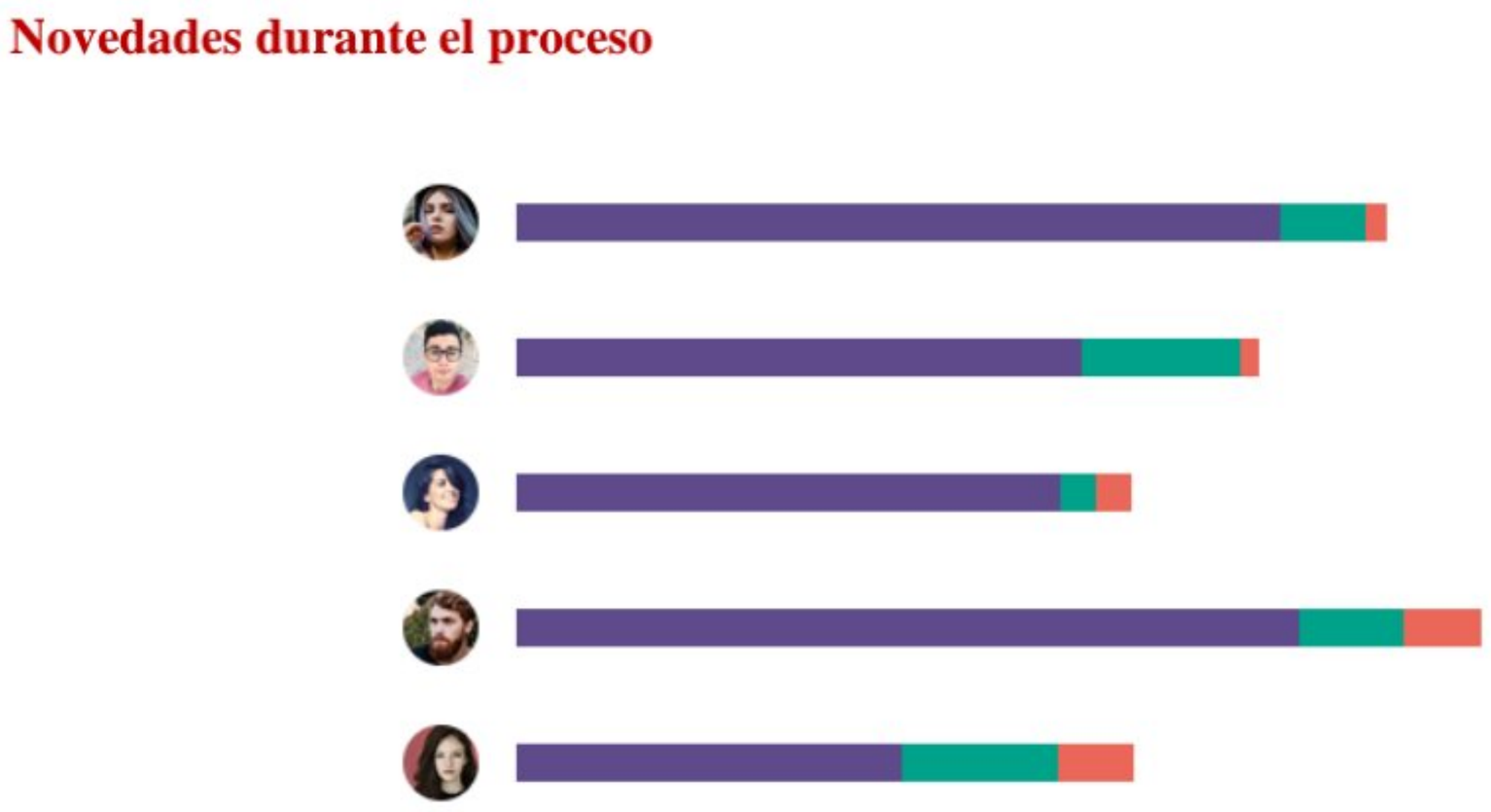

Figura 68. Panel Administrativo eventos y resultados

No presencia del estudiante. - Se presentan los momentos que se registran cuando el sistema de reconocimiento facial no encuentra el rostro del estudiante.

Dos o más rostros detectados. - Son los momentos que el sistema detecta que dos o más rostros se encuentran frente a la cámara.

Suplantación de identidad. - Son los supuestos, que indican que el estudiante no validó o no realizó lo que el sistema le solicitó de forma aleatoria.