

UNIVERSIDAD TECNOLÓGICA (UTEC)
UNIVERSIDADE FEDERAL DO RIO GRANDE (FURG)
UNIVERSIDAD NACIONAL DE RAFAELA (UNRAF)
POSGRADO EN ROBÓTICA E INTELIGENCIA ARTIFICIAL (PRIA)

Proyecto Final

Uso de características profundas en reconocimiento facial inteligente

Matías Camargo

Proyecto final presentado al Posgrado en Robótica e Inteligencia Artificial (PRIA) como requisito parcial para la obtención del grado de Especialista en Robótica e Inteligencia Artificial

Tutor: Prof. Mag. Bruna Guterres
Co-tutor: Prof. Esp. Cuña Pablo

Rivera, 2024

Dedico este trabajo de investigación a mis padres.

AGRADECIMIENTOS

A UTEC, FURG y UNRaf, por la oportunidad de estudio que brindan.

A mi compañera de vida. Quien ha sabido sostenerme y darme fuerzas en esos momentos de intensa tensión.

RESUMEN

CAMARGO, Matías. **Uso de características profundas en reconocimiento facial inteligente.** 2024. 55 f. Proyecto Final – Posgrado en Robótica e Inteligencia Artificial (PRIA). Universidad Tecnológica (UTEC), Rivera.

El reconocimiento facial es crucial para una variedad de aplicaciones en la vida diaria y en entornos empresariales. Sin embargo, la identificación con alto grado de confiabilidad es aún un desafío central para su integración en diversas aplicaciones. En particular, la extracción de características relevantes a este proceso representa un importante desafío considerando la gran variabilidad de las imágenes en términos de posicionamiento, iluminación, presencia o ausencia de accesorios y otros. En este sentido, el presente trabajo propone un sistema de control de acceso basado en la identificación de rostros utilizando redes neuronales. En particular, se propone la utilización del modelo VGG-16 en la extracción de características profundas capaces de maximizar la performance de estas aplicaciones mediante aprendizaje por transferencia, combinado con técnicas de reducción de dimensionalidad como Análisis de Componentes Principales (PCA) y Análisis de Discriminante Lineal (LDA) para extraer características más relevantes. Posteriormente, se realizan comparaciones de proximidad utilizando métricas como la distancia coseno, Mahalanobis y distancia euclíadiana así como la implementación de Máquinas de Vectores de Soporte (SVM) para determinar si un rostro está registrado en una base de datos y sea reconocido para brindar acceso y/o permisos, obteniendo para los mejores casos precision, recall y f1-score de 99.8% lo cual refleja la robustez de la implementación. Este trabajo ha tenido siempre en cuenta el despliegue en dispositivos IoT o dispositivos de borde de una red (*edge*), optando por soluciones que utilizan matrices y cálculos algebraicos en lugar de modelos completos, lo que facilita su implementación en entornos con recursos limitados.

Palabras-clave: Reconocimiento facial, DigiFace1M, Yale, Extended Yale B, Feature extraction.

INDICE DE FIGURAS

Figura 1	Puntos caracteristicos para definir un modelo de cara	10
Figura 2	Arquitectura CNN.	18
Figura 3	Arquitectura VGG16.	18
Figura 4	Este espacio representa las componentes que son similares en cualquier rostro humano, entre estos los ojos, oídos, boca y nariz que poseen una contextura muy parecida entre uno o varios individuos.	20
Figura 5	Linear Discriminant Analysis.	20
Figura 6	Conjunto de similitud de coseno.	21
Figura 7	Hiperplanos de separación.	23
Figura 8	Pipeline de la metodología propuesta para la investigación	27
Figura 9	Ejemplos de imágenes del dataset DigiFace1M	29
Figura 10	Ejemplos de imágenes del dataset YALE	30
Figura 11	Ejemplos de imágenes de Exteded YALE B dataset	31
Figura 12	Diagrama VGG entrenado con sus capas congeladas pre entrenadas (celeste) para un entrenamiento posterior con el dataset DigiFace1M (rosa), teniendo como salida final el resultado del corte (línea roja) de la VGG en la capa block5_conv3	33
Figura 13	Ejemplo de generación de vectores de características y cálculo de distancias para comparación de rostros	35
Figura 14	Accuracy Y Loss del modelo VGG16	37
Figura 15	Evaluación conjunto de pruebas	38
Figura 16	Relación Número de componentes PCA vs Variabilidad de los datos .	39
Figura 17	Número de componentes PCA vs Precisión	40
Figura 18	Matriz de Confusión - Distancia Euclíadiana Cuadrada	41

INDICE DE TABLAS

Tabla 1	Comparativa entre el rendimiento de diferentes técnicas y enfoques para el reconocimiento facial.	16
Tabla 2	Comparación de Componentes y Métricas para dataset YALE	42
Tabla 3	Comparación de Componentes y Métricas para dataset Ext. YALE B .	42
Tabla 4	Comparación de Componentes y Métricas para dataset Ext. YALE B (24 fotos por clase)	42
Tabla 5	Comparación de Componentes y Métricas para dataset YALE + Ext. YALE B	42

LISTA DE ABREVIATURAS Y SIGLAS

VGG	Visual Geometry Group
PCA	Principal Component Analysis
CNN	Convolutional Neural Network
DL	Deep Learning
IoT	Internet of Things
SVM	Maquina de soporte de Vectores
LDA	Análisis Discriminante Lineal
BF	Biometría Facial
CNN	Convolutional Neural Net
RF	Reconocimiento Facial
LBP	Local Binary Patterns
LBP	Local Binary Patterns
DCNN	Deep Convolutional Neural Network
TI	Tecnologías de la Información
IA	Inteligencia Artificial
SaaS	Software as a Service
HOG	Histogramas de Gradientes Orientados
PGM	Portable Gray Map

SUMARIO

1 INTRODUCCIÓN	10
2 OBJETIVOS	12
2.1 Objetivo General	12
2.2 Objetivos Específicos	12
3 REVISIÓN BIBLIOGRÁFICA	13
3.1 Trabajos Relacionados	13
3.2 Fundamentación Teórica	17
3.2.1 CNN	17
3.2.2 VGG	18
3.2.3 PCA	19
3.2.4 LDA	19
3.2.5 Distancia Coseno	21
3.2.6 Distancia euclídea	22
3.2.7 Máquinas de Soporte Vectorial (SVM)	23
3.2.8 Distancia de Mahalanobis	24
3.2.9 Métricas de Evaluación	24
4 METODOLOGÍA	27
4.1 Dataset	28
4.1.1 DigiFace1M	28
4.1.2 YALE	29
4.1.3 Extended YALE B	31
4.1.4 Tratamiento de los datos	31
4.2 Entrenamiento de la red VGG-16	32
4.3 Reducción de dimensiones	34
4.4 SVM y Medidas de similitud	35
5 RESULTADOS Y DISCUSIONES	37
5.1 Entrenamiento de la red VGG	37
5.2 Extracción de características - PCA/LDA	39
5.3 Evaluación	40
6 CONCLUSIONES Y TRABAJOS FUTUROS	43

Bibliografia Consultada **45**

ANEXOS

A Fundamentación matemática **51**

1 INTRODUCCIÓN

La biometría facial (BF), como rama de la visión computacional, ha experimentado un crecimiento exponencial en los últimos años, impulsado por los avances en inteligencia artificial y aprendizaje profundo [Bez et al., 2003]. Si bien la BF es una tecnología relativamente nueva, sus raíces se remontan a principios de la década de 1960. En esa época, Woodrow Wilson Bledsoe [Bledsoe and Browning, 1959] desarrolló un sistema de clasificación manual para reconocer rostros. Este sistema pionero se basaba en la medición de atributos faciales únicos de cada individuo, como la distancia entre los ojos, la forma de la nariz y la posición de la boca [López-Sánchez, 2022]. Figura 1 ilustra un ejemplo claro de estas características.

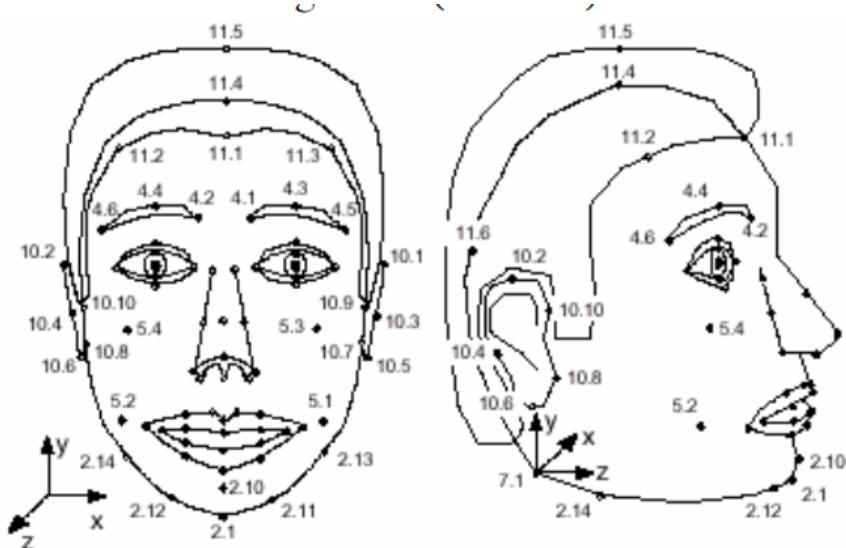


Figura 1: Puntos característicos para definir un modelo de cara
Fuente: [Bez et al., 2003]

El reconocimiento facial (*Facial Recognition*, RF) ha sido un tema de gran interés en el campo de la visión por computadora desde sus inicios. Su objetivo es identificar o verificar a una persona a partir de una imagen o video de su rostro. Sin embargo, lograr un alto rendimiento en esta tarea ha sido un desafío durante muchos años. Avances recientes en inteligencia artificial y aprendizaje profundo han impulsado a la BF a nuevas

alturas. Las técnicas de *Deep Learning* (DL), como las Redes Neuronales Convolucionales Profundas (DCNN), han logrado superar significativamente a los métodos tradicionales, según comenta [Lozano, 2020]. Esta temática incluso tiene un amplio rango de aplicaciones en la actualidad: desbloqueo de dispositivos, como teléfonos inteligentes, ordenadores, tablets, control de acceso físico a oficinas, bancos, aeropuertos, transferencias bancarias, pagos en línea, inicio de sesión en aplicaciones web y redes corporativas afirma [Domínguez Pavón, 2017]. [Lozano, 2020] suma otra variedad de usos, como las aplicaciones en el automovilismo, la medicina y en ámbitos militares.

En el contexto actual, el reconocimiento facial ha emergido como una tecnología clave con una amplia gama de aplicaciones comerciales y de seguridad. Según [Morcillo Vizuete, 2021], el mercado de tecnologías de reconocimiento facial se divide principalmente en tres categorías: motores basados en software como servicio (SaaS), soluciones API REST auto hospedadas y *frameworks* de código abierto. Estas soluciones ofrecen desde conveniencia y facilidad de uso hasta flexibilidad y control total sobre la implementación, cada una con sus propias ventajas y consideraciones. En este mismo trabajo de grado, el cual raliza un abordaje del estudio de las variables a considerar para el desarrollo de un modelo viable de aplicación de un sistema de identificación facial segura en un entorno de organizaciones de servicios evalúa la adopción desde motores gratuitos como *Face Recognition* y *OpenCV* hasta soluciones de pago como *Amazon Recognition* y *Face++*, el mercado ofrece una variedad de opciones para satisfacer las necesidades de diferentes usuarios. Este panorama diversificado refleja el creciente interés y la adopción generalizada del reconocimiento facial en una variedad de entornos comerciales y de seguridad. Sin embargo, el desarrollo de modelos capaces de reconocimiento facial enfrenta dificultades tales como la variabilidad en las condiciones de iluminación, la diversidad en las expresiones faciales y las diferencias entre las imágenes de entrenamiento y las de prueba [Prado et al., 2020].

Este trabajo propone una solución de inteligencia artificial que utiliza características profundas extraídas de redes neuronales convolucionales pre-entrenadas, combinadas con técnicas de reducción de dimensionalidad como el Análisis de Componentes Principales (PCA) y el Análisis Discriminante Lineal (LDA). El objetivo es mejorar la precisión y aciertos del reconocimiento facial frente a las variaciones y desafíos existentes. La estructura del desarrollo sigue un orden específico: primero, se introducirá el tema y los objetivos (Capítulo 1 y 2), seguido de una revisión bibliográfica que abarca trabajos relacionados y fundamentos teóricos esenciales, como CNN, VGG, PCA, LDA, y diversas métricas de evaluación (Capítulo 3). La metodología se detallará en el Capítulo 4, incluyendo el tratamiento de datos, el entrenamiento de la red VGG-16, la reducción de dimensiones, y la implementación de SVM y medidas de similitud. Finalmente, se presentarán los resultados alcanzados y se ofrecerán conclusiones y propuestas de trabajos futuros (Capítulo 5 y 6).

2 OBJETIVOS

2.1 Objetivo General

Este proyecto se centra en desarrollar un sistema de reconocimiento facial eficiente y preciso utilizando algoritmos avanzados de *deep learning* y la extracción de características (*feature extraction*) que permitan identificar y comparar imágenes faciales con alta precisión.

2.2 Objetivos Específicos

Para cumplir con los objetivos generales del proyecto se buscará:

- Desarrollar y entrenar un modelo de deep learning para RF utilizando *dataset* DigiFace1M y técnicas de *transfer learning*.
- Explorar el uso de modelos de aprendizaje profundo en la extracción de características.
- Emplear PCA y LDA en conjunto con métricas de evaluación de características aplicables al RF.
- Emplear y comparar distintas métricas de evaluación.
- Considerar el uso de recursos computacionales en la selección de los modelos propuestos.

3 REVISIÓN BIBLIOGRÁFICA

3.1 Trabajos Relacionados

La investigación de literatura relevante para este estudio fue realizada en bases de datos académicas de renombre tales como PubMed, Google Scholar y IEEE Xplore, entre otras, con el fin de recopilar estudios que, comparten similitudes en cuanto a metodología y enfoque teórico. Se puso especial énfasis en incluir investigaciones que posean elementos metodológicos, teóricos o analíticos que se consideran valiosos para el presente trabajo.

En el ámbito del reconocimiento facial y análisis de imágenes mediante algoritmos de aprendizaje profundo, se ha realizado una variedad de estudios que han contribuido significativamente al progreso tecnológico y metodológico del campo. Estos estudios son buenos ejemplos de cómo los avances en algoritmos de aprendizaje profundo, como VGG-16, junto con técnicas como el aprendizaje por transferencia y el aumento de datos, están mejorando el desarrollo y la precisión de los sistemas de reconocimiento facial y clasificación de imágenes.

[Hong et al., 2021] propone una aplicación basada en RF mediante el reconocimiento de síndromes genéticos a través del modelo desarrollado por el *Visual Geometry Group*, de Oxford de 16 capas neuronales (VGG-16). En un contexto más genérico de clasificación de imágenes, [Tammina, 2019] demostró cómo la incorporación del aprendizaje por transferencia, específicamente utilizando VGG-16 junto con técnicas de aumento de imágenes (*data augmentation*), puede incrementar la precisión en la clasificación hasta un 95.40%. Este estudio resalta la importancia de combinar modelos pre-entrenados con estrategias de procesamiento de datos para mejorar la eficacia en el reconocimiento de patrones. [Liu et al., 2021] presenta una comparación directa entre diferentes arquitecturas de redes neuronales convolucionales profundas para el reconocimiento facial del síndrome de Williams-Beuren¹, destacando que la arquitectura VGG-19 sobresale por su precisión y confiabilidad en relación a VGG-16, ResNet-18, ResNet-34, and MobileNet-V2.

Existen diversos estudios y mecanismos que han logrado, y en ocasiones incluso me-

¹trastorno genético caracterizado por rasgos faciales distintivos

jorado, los resultados obtenidos con técnicas de aprendizaje por transferencia. Estos estudios subrayan la importancia de ajustar modelos preentrenados, como VGG-19 y MobileNet, para mejorar la precisión en las predicciones de tareas específicas. Por ejemplo, trabajos como el de [Zhang et al., 2017] sugieren que seleccionar y adaptar las capas de estos modelos puede ser crucial para obtener un rendimiento óptimo en el reconocimiento de imágenes.

Dentro de este contexto de optimización de la extracción de características, también se han explorado técnicas complementarias, como el análisis de componentes principales (PCA). Este enfoque ha sido ampliamente utilizado para mejorar la representación de características en distintos dominios. Por ejemplo, Castrillón [Castrillón et al., 2008] aplicó PCA en el procesamiento de imágenes para el reconocimiento de expresiones faciales. De manera similar, el trabajo de [Chen and Haoyu, 2019] presenta una propuesta innovadora al fusionar el modelo VGG-16 para la extracción de características faciales con el algoritmo *Support Vector Machine* (SVM) para la clasificación, integrando tanto técnicas de transferencia de aprendizaje como métodos de reducción de dimensionalidad como PCA para mejorar la precisión en la clasificación.

El enfoque del algoritmo se fundamenta en la extracción de características faciales mediante la red VGG-16, seguida de una reducción de dimensionalidad a través de PCA y la clasificación mediante SVM con función de kernel lineal. Esta combinación de técnicas demostró ser altamente efectiva en el reconocimiento facial. En el trabajo de [Barnouti et al., 2016] se presenta un sistema de detección y reconocimiento facial que combina el método de detección facial Viola-Jones con PCA y LDA (Análisis Discriminante Lineal) para la extracción de características y reducción de dimensiones, utilizando además la Distancia Euclídea Cuadrada. El método Viola-Jones se destaca por su eficiencia en la detección de rostros en tiempo real, lo que contribuye significativamente a la rapidez y precisión del proceso de reconocimiento facial.

Mientras que PCA se centra en la reducción de dimensionalidad, Viola-Jones en la detección de objetos, y VGG en el aprendizaje profundo, en el trabajo de Cerna et al. [Cerna et al., 2013], se destaca el uso de los descriptores de Histogramas de Gradientes Orientados (HOG), los cuales capturan los gradientes de intensidad local en una imagen para detectar objetos con formas y apariencias distintas. Al utilizar estos histogramas como características de entrada para entrenar un modelo de SVM, se logra mejorar el rendimiento del clasificador SVM en la distinción entre rostros y no rostros, validando así la eficacia del método propuesto en esta aplicación.

El estudio presentado por Rahmad et al. [Rahmad et al., 2020] se enfoca en comparar dos métodos de detección de rostros: el Clasificador en Cascada de Haar de Viola-Jones y el Histograma de Gradientes Orientados (HOG). Los resultados obtenidos indican que el método basado en HOG supera al Clasificador en Cascada de Haar de Viola-Jones en términos de precisión, sugiriendo que HOG es más efectivo para la detección de rostros.

en diferentes condiciones. Estos resultados pueden observarse en la tabla de resultados.

Después de abordar el concepto HOG, es relevante explorar otra técnica ampliamente utilizada en el campo de la visión por computadora: los Patrones Binarios Locales (LBP - *Local Binary Patterns*). Esta técnica se destaca por su capacidad para describir la textura de una imagen de manera eficiente. En el artículo de [Xie et al., 2019] se analiza el uso de LBP en el reconocimiento facial, resaltando su facilidad de implementación, tolerancia a cambios de iluminación y baja complejidad computacional.

Sin embargo, en el trabajo de [Timo, 2004], se llevaron a cabo pruebas utilizando el sistema *Face Identification Evaluation System* con conjuntos de imágenes que presentaban diversas condiciones. Los resultados indicaron que el método basado en LBP superó ese y otros métodos, como PCA, en todos los conjuntos de pruebas.

Las redes neuronales convolucionales (CNN) con su implementación de redes siamesas han emergido como una arquitectura poderosa para el RF, especialmente en escenarios con conjuntos de datos pequeños. El artículo [Heidari and Fouladi-Ghaleh, 2020] destaca varias ventajas significativas de esta combinación. En primer lugar, la integración de redes siamesas con *transfer learning*, utilizando modelos pre-entrenados como VGG-16, ha mostrado mejorar la precisión del reconocimiento facial hasta un 95.2%, superando a otros métodos tradicionales en entornos con datos limitados.

A sí mismo, [Wu et al., 2017] destaca la ventaja de las redes siamesas en su capacidad para proporcionar resultados concretos en cuanto a similitud entre dos imágenes de entrada. Sin embargo, a pesar de sus beneficios, las redes siamesas presentan algunas limitaciones significativas afirma [Núñez Ortega, 2023]. Entre las algunas de las dificultades que presenta o incluso limitaciones se incluyen la complejidad en el entrenamiento, esta etapa requiere un gran número de pares de imágenes etiquetadas para obtener resultados precisos, y la dificultad para generalizar en conjuntos de datos con variabilidad extrema. Estas desventajas motivaron la exploración de otras técnicas más eficientes y prácticas para nuestro caso de uso específico.

La introducción de *AdaBoost* en el reconocimiento facial destaca su papel en mejorar la precisión de los sistemas de visión por computadora al combinar múltiples clasificadores débiles. Por ejemplo, en [Ruan et al., 2010] proponen un método de selección de características basado en Real *AdaBoost*, realizando distintas comparaciones de modelos, Adaboost.MH, Adaboost.MO y Adaboost.MR con el conjunto de datos FRGC 2.0. Además, [Susheel Kumar et al., 2011] utilizaron *AdaBoost* junto con el clasificador de cascada Haar para la detección facial y técnicas de reducción de dimensionalidad para el reconocimiento facial, considerando también la optimización del consumo de recursos computacionales.

La Tabla 1 resume los trabajos relacionados en términos de datasets empleados, performance y metodología.

Trabajo	Dataset	Metodología	Resultados
[Hong et al., 2021]	Guangdong Provincial People's Hospital	VGG-16 Transfer	Accuracy: 86.60% Precision: 90.79% F1-Score: 88.29%
[Tammina, 2019]	Custom	Fine tuning CNN with pretrained VGG-16 model and image augmentation	Accuracy: 95.40%
[Tian et al., 2022]	VGG Face Dataset	VGG-Face 13 convolutional layers and 3 fully connected layers	Accuracy: 67.80%
[Liu et al., 2021]	Guangdong Provincial People's Hospital	VGG-16	Accuracy: 90.9% Precision: 88.8% F1-Score: 84.7%
[Lv et al., 2023]	AR Database	PCA	Accuracy: 91.40%
[Goel et al., 2021]	Custom	VGG-Cosine Similarity	Accuracy: 100.0% Precision: 100.0% MC Rate: 3.7%
[Kar et al., 2012]	RAV2D FERET	GWT-KDCV	Accuracy: 85.00%
[Zhang et al., 2017]	Custom	VGG-Face Transfer & PCA(80)	AUC: 99.72%
[Raj, 2011]	Yale Face	PCA + Euclidean distance	Accuracy: 67.1%
		PCA + Manhattan Distance	Accuracy: 79.1%
		PCA + Mahalanobis Distance	Accuracy: 91.7%
	AT&T Face	PCA + Euclidean distance	Accuracy: 89.4%
		PCA + Manhattan Distance	Accuracy: 88.8%
		PCA + Mahalanobis Distance	Accuracy: 94.1%
[Dar and Palanivel, 2021]	Yale Dataset	VGG-16	Accuracy: 97.50%
[Susheel Kumar et al., 2011]	Custom	AdaBoost con PCA PCA con LDA	Accuracy: 99%
[Schroff et al., 2015]	Labeled Faces in the Wild (LFW) y Youtube Faces DB	FaceNet	Accuracy: 99.63%
[Cerna et al., 2013]	Base de datos AT&T	SVM y HOG	Accuracy: 99.71%
[Rahmad et al., 2020]	Custom	HOG vs Viola-Jones	Precisión: 79%
[Barnouti et al., 2016]	MUCT, Face94 y Grimace	Viola-Jones con PCA-LDA y Distancia Euclídea Cuadrada	Precisión: 87.5%
[Timo, 2004]	Custom	Local Binary Pattern	Precisión: 97% en fb y 79% en fc
[Chen and Haoyu, 2019]	CelebA dataset & LFW dataset (Labeled Faces in the Wild)	VGG-16 & SVM	Accuracy: 97.47%
[Heidari and Fouladi-Ghaleh, 2020]	LFW dataset	Siamese-VGG	Accuracy: 95.62%

Tabla 1: Comparativa entre el rendimiento de diferentes técnicas y enfoques para el reconocimiento facial.

La bibliografía estudiada en este trabajo, en relación al campo del reconocimiento facial ha experimentado un notable avance gracias a la integración de diversas técnicas de inteligencia artificial y aprendizaje automático. Desde los enfoques tradicionales basados en características geométricas hasta los más recientes métodos de aprendizaje profundo, los investigadores han logrado mejoras significativas en la precisión y robustez de los sistemas de reconocimiento facial. Sin embargo, a pesar de estos avances, aún existen desafíos como la variabilidad en las condiciones de iluminación, las expresiones faciales y las occlusiones que requieren soluciones innovadoras.

A diferencia de los trabajos estudiados, este estudio se denota diferente no solo por el tipo de dataset empleado, que es único en su estructura y contenido, sino también

por la combinación de distintas técnicas de extracción de características y reducción de dimensionalidad, lo que aporta un enfoque novedoso y potencialmente más eficaz para superar los desafíos mencionados.

3.2 Fundamentación Teórica

En esta sección, se abordarán los diferentes métodos y técnicas que forman la base teórica de este trabajo. Comenzaremos con una introducción a las Redes Neuronales Convolucionales (CNN), su estructura y aplicaciones en la clasificación de imágenes. Luego, se describirá en detalle la arquitectura VGG, un modelo específico de CNN ampliamente utilizado en el reconocimiento de imágenes. A continuación, se explicarán los métodos de Análisis de Componentes Principales (PCA) y Análisis Discriminante Lineal (LDA) para la reducción de dimensionalidad y la separación de clases. Finalmente, se discutirán las diferentes métricas de distancia utilizadas para la clasificación, incluyendo la distancia coseno, la distancia euclídea y la distancia de Mahalanobis, así como el uso de Máquinas de Soporte Vectorial (SVM) como técnica de clasificación. En la sección de Anexo se podrá encontrar la fundamentación matemática que refuerzan y justifican las implementaciones de PCA, LDA y SVM.

3.2.1 CNN

Las redes neuronales convolucionales (CNN) se utilizan para procesar imágenes y pueden aprender relaciones entrada-salida, donde la entrada es una imagen y la salida puede variar según el contexto de estudio. Estas redes están basadas en operaciones de convolución y se emplean en tareas comunes como la detección y categorización de objetos, la clasificación de escenas y la clasificación de imágenes en general. [Sánchez et al., 2019]

La convolución consiste en filtrar una imagen usando una máscara, donde diferentes máscaras producen distintos resultados. En la convolución, cada píxel de salida es una combinación lineal de los píxeles de entrada. Las redes CNN se forman utilizando tres tipos de capas: capas convolucionales, que requieren el uso de máscaras y realizan filtrado seguido de una función de activación, capas de *pooling*, y capas totalmente conectadas, que se aplican al final de la red y en las que se pierde la información espacial original.[Loncomilla, 2016]

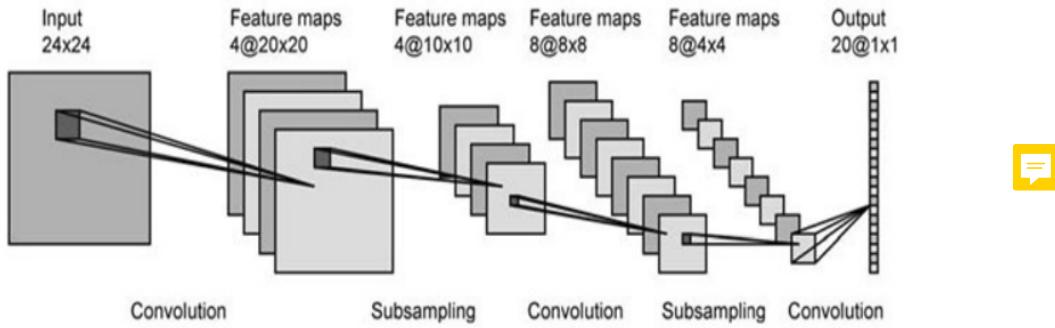


Figura 2: Arquitectura CNN.

Fuente: [Loncomilla, 2016]

3.2.2 VGG

La red neuronal convolucional VGG (*Visual Geometry Group*) es un modelo de aprendizaje profundo ampliamente utilizado en tareas de reconocimiento de imágenes. Fue desarrollado por el grupo VGG de la Universidad de Oxford y se destaca por su arquitectura simple y efectiva. [Simonyan and Zisserman, 2014]

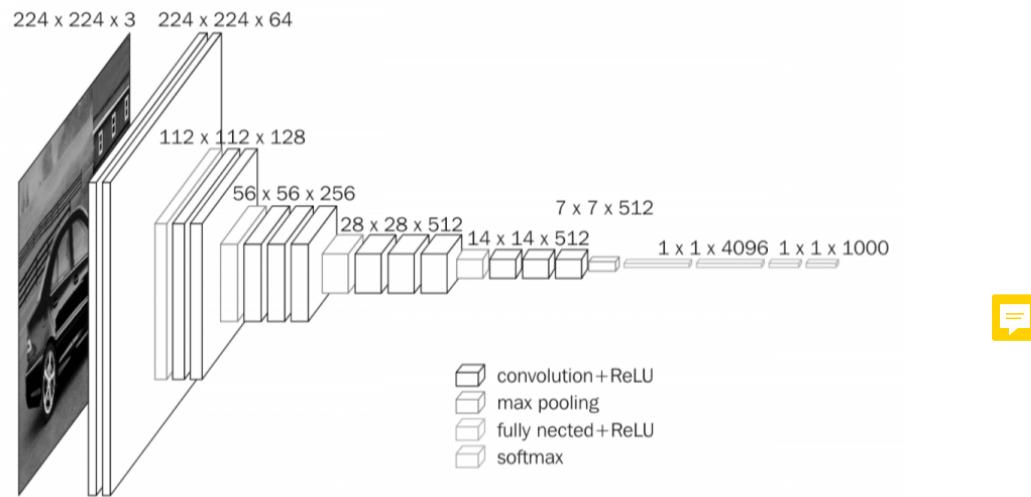


Figura 3: Arquitectura VGG16.

Fuente: [Salih et al., 2019]

La arquitectura VGG consta de 16 capas (VGG-16) o 19 capas (VGG-19), que incluyen principalmente capas convolucionales y capas de agrupamiento, seguidas de capas totalmente conectadas. Estas redes utilizan filtros convolucionales de tamaño reducido (generalmente 3x3) y técnicas de agrupamiento para extraer características de las imágenes de entrada [Tammina, 2019]. En la arquitectura de VGG-16, se intercalan capas de agrupamiento para reducir la dimensionalidad de las características extraídas, manteniendo la información relevante. Al final de la red, se agregan capas totalmente conectadas para la clasificación final, utilizando las características aprendidas por las capas convolucionales para realizar la clasificación en las clases de interés. Se aplican funciones de

activación para introducir no linealidades y facilitar una convergencia más rápida durante el entrenamiento [Salih and Hussein, 2019].

3.2.3 PCA

[Pearson, 1901] introdujo conceptos fundamentales que más tarde se consolidarían en el desarrollo del Análisis de Componentes Principales (PCA). Este artículo se centró en la idea de ajustar líneas y planos a conjuntos de puntos en un espacio tridimensional, como se detalló en la figura 4. Utilizando métodos matemáticos, desarrolló técnicas para minimizar la suma de los cuadrados de las distancias entre los puntos de datos y la línea o el plano ajustado. Este enfoque no solo proporcionó una forma de representar datos de manera más comprensible, sino que también sentó las bases para el análisis estadístico de datos multivariantes. [Faruqe and Hasan, 2009] indica que las técnicas de PCA se utilizan para reducir la dimensionalidad de los datos de imágenes faciales al encontrar los componentes principales que mejor representan la variación en el conjunto de datos. Esto ayuda a extraer las características más importantes de las imágenes faciales, permitiendo una representación y análisis eficientes de los datos. Al transformar las imágenes faciales de alta dimensionalidad en un espacio de menor dimensionalidad, el PCA simplifica los datos mientras retiene la información esencial para el reconocimiento.

El Análisis de Componentes Principales (PCA) se ha convertido en una herramienta fundamental en el ámbito del reconocimiento facial [Faruqe and Hasan, 2009] y funciona proyectando las imágenes faciales en un espacio de características de menor dimensión, denominado "espacio de caras", donde cada imagen se representa por una combinación única de *Eigenfaces*. Estas *Eigenfaces*, son los vectores propios de un conjunto de caras de entrenamiento y representan las características faciales predominantes. Al comparar la composición de *Eigenfaces* en una imagen facial desconocida contra las características de las caras conocidas almacenadas en una base de datos de personas con determinados accesos, un sistema de RF puede identificar con alta precisión a la persona correspondiente [Abdullah, 2012].

Las ventajas de utilizar PCA en el reconocimiento facial son múltiples. En primer lugar, su implementación es sencilla y computacionalmente eficiente, lo que la convierte en una solución atractiva para aplicaciones en tiempo real. Además, PCA sobresale por su precisión en la identificación de rostros, incluso en condiciones de iluminación o ángulos faciales variables. [Kaur and Himanshi, 2015].

3.2.4 LDA

La introducción al trabajo de [Fisher, 1936] es esencial para comprender los fundamentos de la estadística y el análisis discriminante como una técnica para clasificar observaciones en diferentes grupos basándose en características observadas. Fisher buscaba encontrar una combinación lineal de variables que maximizaría la separación entre múltiples

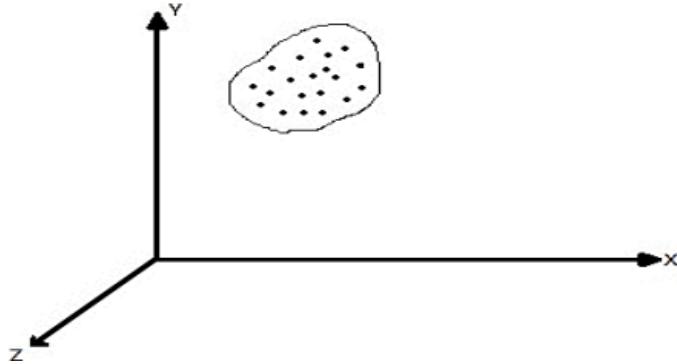


Figura 4: Este espacio representa las componentes que son similares en cualquier rostro humano, entre estos los ojos, oídos, boca y nariz que poseen una textura muy parecida entre uno o varios individuos.

Fuente: [Mejía-Campos et al., 2017]

clases. Su enfoque se centró en la maximización de la razón de la varianza entre las clases a la varianza dentro de las clases. Esto se traduce en la búsqueda de una proyección que permita distinguir de manera efectiva entre diferentes grupos en un conjunto de datos. La figura 5 ilustra el resultado del Análisis Discriminante Lineal (LDA) donde la distancia entre los puntos proyectados indica la separación entre clases en el espacio proyectado. LDA utiliza esta separación para mejorar la clasificación de los datos.

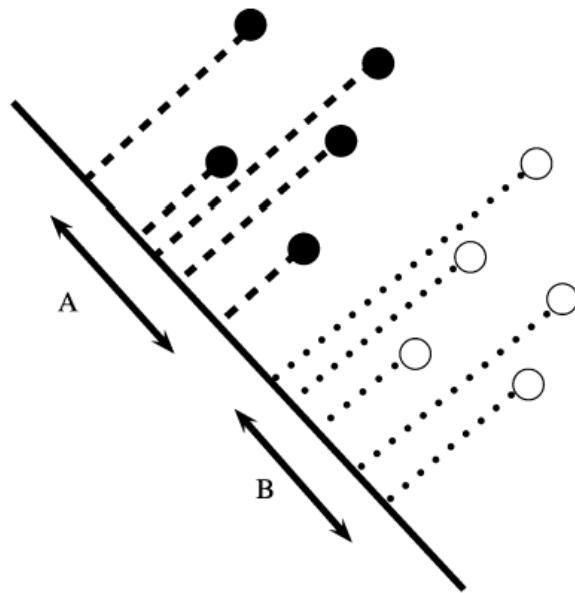


Figura 5: Linear Discriminant Analysis.

Fuente: [Xanthopoulos et al., 2013]

El Análisis Discriminante Lineal (LDA) es comúnmente definido como una técnica de reducción de dimensionalidad supervisada y lineal. No obstante, algunos autores como [Anowar et al., 2021] también consideran que LDA actúa como un clasificador lineal.

LDA identifica un nuevo espacio de características para proyectar los datos con el objetivo de maximizar la separabilidad de las clases. A partir de las características independientes de un conjunto de datos, extrae k nuevas características independientes que separan las clases de manera más efectiva, siendo el número de componentes producidos menor que el número de clases menos uno.

3.2.5 Distancia Coseno

La similitud del coseno utiliza el valor del coseno del ángulo entre dos vectores en el espacio vectorial como una medida de la diferencia entre dos individuos. En comparación con la medida de la distancia, la similitud del coseno se enfoca más en la diferencia en la dirección de los vectores que en su distancia o longitud. [Muñoz Plá, 2023]

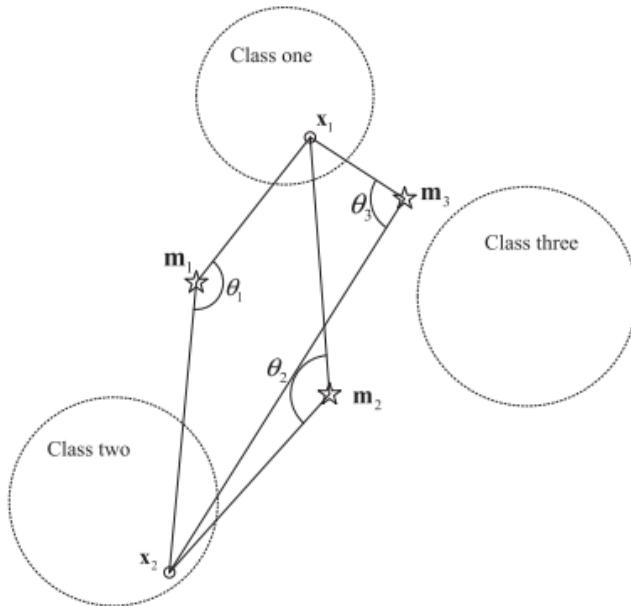


Figura 6: Conjunto de similitud de coseno.

Fuente: [Xia et al., 2015]

La distancia de coseno es una medida de la disimilitud entre dos vectores. En el contexto de inteligencia artificial y aprendizaje automático, se utiliza a menudo para comparar vectores de características. La fórmula para calcular la distancia de coseno entre dos vectores \mathbf{A} y \mathbf{B} es la siguiente:

$$\text{Distancia de Coseno} = 1 - \cos(\theta) = 1 - \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} \quad (1)$$

donde:

- $\mathbf{A} \cdot \mathbf{B}$ es el producto punto (o producto escalar) de los vectores \mathbf{A} y \mathbf{B} .
- $\|\mathbf{A}\|$ y $\|\mathbf{B}\|$ son las normas (o magnitudes) de los vectores \mathbf{A} y \mathbf{B} respectivamente.

- $\cos(\theta)$ es el coseno del ángulo θ entre los dos vectores.

Matemáticamente, el cálculo se puede expresar como:

$$\mathbf{A} \cdot \mathbf{B} = \sum_{i=1}^n A_i B_i \quad (2)$$

$$\|\mathbf{A}\| = \sqrt{\sum_{i=1}^n A_i^2} \quad (3)$$

$$\|\mathbf{B}\| = \sqrt{\sum_{i=1}^n B_i^2} \quad (4)$$

Por lo tanto, la fórmula completa se convierte en:

$$Distancia\ de\ Coseno = 1 - \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad (5) \quad \text{💡}$$

Esta métrica es particularmente útil en aplicaciones de aprendizaje automático y procesamiento de lenguaje natural, donde la orientación de los vectores es más importante que su magnitud. En [Mendoza Olguín et al., 2019] se define el rango de la similitud del coseno como un valor que está entre -1 y 1. En el espacio del producto interno, esta es una medida de similitud en la dirección (y no en el tamaño) entre dos vectores que no son el vector cero. La similitud del coseno es igual al coseno del ángulo entre dos vectores y es lo mismo que un producto interno entre dos vectores normalizados.

- Si dos vectores están en la misma dirección, la similitud del coseno es 1.
- Si dos vectores están en un ángulo de 90 grados, la similitud del coseno es 0.
- Si son opuestos, entonces la similitud del coseno es -1.

3.2.6 Distancia euclíadiana

Esta métrica puede ser utilizada para calcular la distancia entre dos puntos cualesquiera en un espacio bidimensional, como también para medir la distancia absoluta entre dos puntos en un espacio N-dimensional. En el caso del reconocimiento facial, una menor distancia indica mayor similitud [Toro et al., 2023].

La distancia euclíadiana es una medida de la distancia entre dos puntos en un espacio euclíadiano. En el contexto de inteligencia artificial y aprendizaje automático, se utiliza a menudo para calcular la distancia entre vectores de características. La fórmula para calcular la distancia euclíadiana entre dos vectores \mathbf{A} y \mathbf{B} es la siguiente:

$$Distancia\ Euclíadiana = \sqrt{\sum_{i=1}^n (A_i - B_i)^2} \quad (6) \quad \text{💡}$$

donde:

- \mathbf{A} y \mathbf{B} son los dos vectores entre los cuales se está calculando la distancia.
- A_i y B_i son los componentes i -ésimos de los vectores \mathbf{A} y \mathbf{B} , respectivamente.
- n es el número de dimensiones de los vectores.

Esta métrica es particularmente útil en aplicaciones de aprendizaje automático donde se necesita medir la similitud o disimilitud entre puntos de datos en un espacio de características.

La implementación para el método de *EigenFaces* aplica la fórmula matemática de Distancia Euclídea ya que no consume demasiados recursos. Afirma [Espinoza Olguín and Guillen, 2015] que este enfoque es eficiente en términos computacionales, permitiendo un reconocimiento rápido y preciso en aplicaciones de tiempo real. Además, la Distancia Euclídea es fácil de interpretar y calcular, lo que la convierte en una elección popular para sistemas de reconocimiento facial que necesitan procesar grandes volúmenes de datos con rapidez y precisión.

3.2.7 Máquinas de Soporte Vectorial (SVM)

Las Máquinas de Vectores de Soporte (SVM) son sistemas de aprendizaje propuestos por [Vapnik, 2010], que se utilizan para abordar el problema de clasificación de datos mediante la estimación de un hiperplano óptimo. La figura 7 muestra como este hiperplano separa los datos de diferentes clases con un margen máximo, lo que significa que maximiza la distancia entre los datos más cercanos al hiperplano y el propio hiperplano. Los vectores que se encuentran sobre este margen se denominan vectores de soporte.

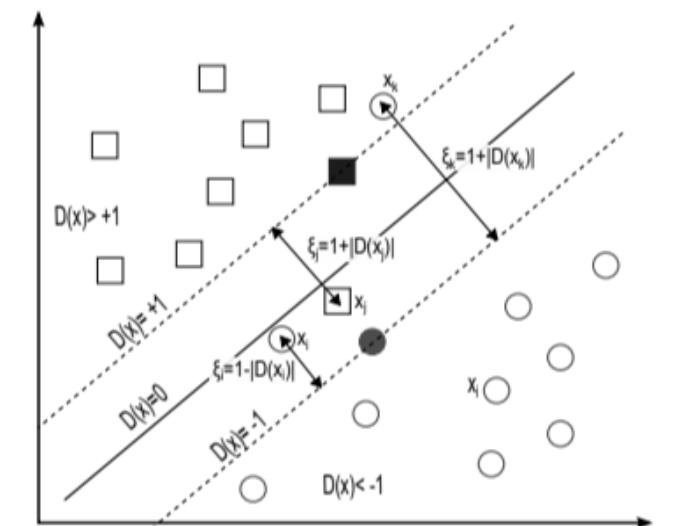


Figura 7: Hiperplanos de separación.

Fuente: [Suárez, 2014]

En [Hsu and Lin, 2002] las SVM son reconocidas por su fuerte capacidad de aprendizaje, incluso con poca información de entrada. La teoría de las SVM se revisa desde el caso más sencillo, donde el hiperplano óptimo se define como el único que separa los datos con un margen máximo, lo que determina la distancia máxima entre las proyecciones de dos vectores de diferentes clases en el conjunto de entrenamiento.

3.2.8 Distancia de Mahalanobis

La distancia de Mahalanobis es una medida alternativa a la distancia euclídea, diseñada para evaluar la similitud entre dos variables aleatorias multidimensionales teniendo en cuenta las correlaciones entre ellas. A diferencia de la distancia euclídea, que trata todas las dimensiones por igual, la distancia de Mahalanobis ajusta la importancia de cada dimensión de acuerdo con su varianza y la relación de correlación entre dimensiones [Lorenzo, 2019]. En el trabajo realizado por [Portillo and Plata, 2008] destaca varias propiedades y particularidades de las aplicaciones del cálculo de distancia Mahalanobis.

Estas propiedades son cruciales para situaciones donde las variables tienen diferentes escalas o varianzas, como en el caso de clasificar manzanas por su diámetro y peso. La distancia de Mahalanobis asegura que ambas variables, independientemente de su varianza, contribuyan equitativamente al cálculo de la distancia, permitiendo una comparación más precisa. Para que una medida sea considerada distancia de Mahalanobis, debe cumplir con propiedades fundamentales como semipositividad, simetría y desigualdad triangular. Su capacidad para integrar la matriz de covarianza en el cálculo de distancias la convierte en una herramienta poderosa para el análisis de datos multivariados, donde la consideración de correlaciones entre variables es esencial para obtener resultados precisos y significativos.

La distancia de Mahalanobis entre dos puntos \mathbf{x} y \mathbf{y} se define como:

$$D_M(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T \mathbf{S}^{-1} (\mathbf{x} - \mathbf{y})} \quad (7)$$

donde:

- \mathbf{x} y \mathbf{y} son los vectores de datos.
- \mathbf{S} es la matriz de covarianza de los datos.
- \mathbf{S}^{-1} es la inversa de la matriz de covarianza.



3.2.9 Métricas de Evaluación

Para evaluar el rendimiento de los modelos de clasificación, se utilizan varias métricas, entre ellas la exactitud (accuracy), la precisión (precision), el recall y el F1-score, como se describe en [Parra, 2015]. Estas métricas se calculan a partir de la matriz de confusión, que muestra la comparación entre las predicciones del modelo y las clases verdaderas.

El recall mide la capacidad del modelo para identificar todas las instancias positivas, mientras que el F1-score proporciona una medida combinada de precisión y recall, útil en escenarios donde el equilibrio entre estas dos métricas es crucial.

Matriz de Confusión

La matriz de confusión para un problema binario se representa de la siguiente manera:

<i>Actual/Predicción</i>	<i>Positivo</i>	<i>Negativo</i>
<i>Positivo</i>	<i>TP</i>	<i>FN</i>
<i>Negativo</i>	<i>FP</i>	<i>TN</i>

donde:

- TP = Verdaderos Positivos
- TN = Verdaderos Negativos
- FP = Falsos Positivos
- FN = Falsos Negativos

Exactitud (Accuracy)

La exactitud mide la proporción de predicciones correctas realizadas por el modelo:

$$\text{Exactitud} = \frac{TP + TN}{TP + FP + TN + FN} \quad (8)$$

Precisión (Precision)

La precisión mide la proporción de verdaderos positivos sobre el total de predicciones positivas:

$$\text{Precisión} = \frac{TP}{TP + FP} \quad (9)$$

Recall

El recall mide la proporción de verdaderos positivos que el modelo ha identificado correctamente de todas las instancias positivas reales. Se calcula de la siguiente manera:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (10)$$

Donde TP son los verdaderos positivos y FN son los falsos negativos.

F1-Score

El F1-score es la media armónica de la precisión y el recall, proporcionando una medida combinada que considera tanto la precisión como el recall del modelo. Se calcula utilizando la siguiente fórmula:

$$F1 - Score = 2 \cdot \frac{Precisión \cdot Recall}{Precisión + Recall} \quad (11)$$

Donde Precisión y Recall se calculan según las fórmulas mencionadas anteriormente.

Estas métricas son fundamentales para evaluar el rendimiento de un modelo de clasificación. La exactitud (*accuracy*) ofrece una medida general de las predicciones correctas, mientras que la precisión (*precision*) indica la proporción de predicciones positivas que son verdaderas. El *recall* evalúa la capacidad del modelo para identificar todas las instancias positivas reales, y el *F1-score* es una combinación entre precisión y recall en una sola métrica, lo que concluye en conjunto estas métricas permiten una comprensión más completa de las fortalezas y debilidades del modelo.

4 METODOLOGÍA

El presente trabajo se centra en la implementación de un pipeline para el reconocimiento facial, comenzando con la selección del dataset DigiFace1M, un conjunto sintético con amplia variabilidad en condiciones de imagen. El modelo VGG16 pre-entrenado se ajusta mediante Transfer Learning. Para la reducción de las dimensiones de las características extraídas, se realiza el ajuste de PCA y LDA utilizando imágenes del dataset YALE, optimizando así la representación y clasificación de las características. Finalmente, se aplican diversas medidas de similitud para evaluar y comparar el rendimiento del modelo. La Figura 8 ilustra el pipeline de procesamiento propuesto.

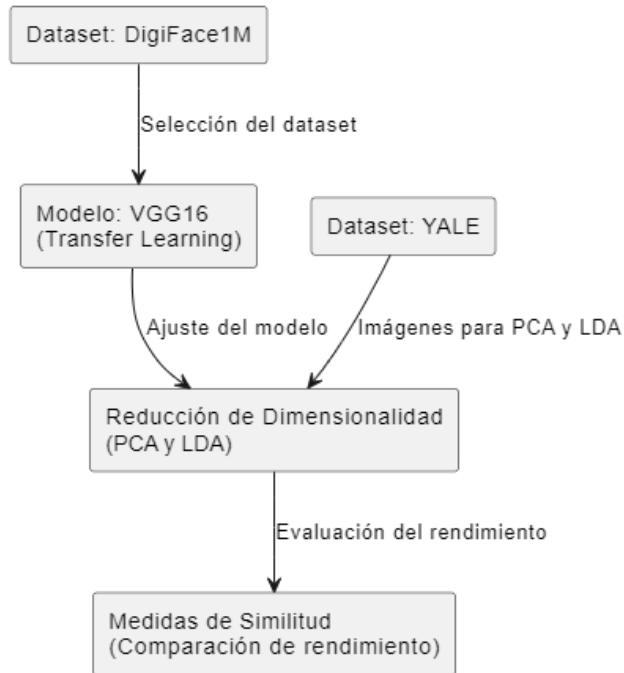


Figura 8: Pipeline de la metodología propuesta para la investigación
Fuente: Elaboración propia

4.1 Dataset

El primer paso fundamental en nuestro trabajo fue la obtención de un dataset adecuado. La selección de un dataset equilibrado y balanceado es crucial para garantizar la precisión y la validez de los resultados en RF. No obstante, esta tarea presenta varios desafíos debido a las condiciones variables en las que se capturan las imágenes. Estas variabilidades incluyen la distancia entre la cámara y el sujeto, las diferencias en las condiciones de iluminación, y las variaciones en las expresiones faciales y oclusiones entre otros.

Los conjuntos de datos disponibles en el ámbito del RF suelen presentar varios problemas que deben ser abordados. El RF se considera uno de los sistemas biométricos más complejos dentro del campo del reconocimiento de patrones [Oloyede et al., 2020]. Un problema común es el desbalanceo en los conjuntos de datos, donde algunas clases tienen muchas más imágenes que otras, lo que puede sesgar el entrenamiento del modelo. Además, las imágenes pueden ser tomadas en diversos escenarios, con diferentes cámaras, ángulos, calidad de imagen y distancias, e incluso en distintos momentos para las mismas personas, lo que puede contribuir a un entrenamiento erróneo del modelo [Oloyede et al., 2020].

En particular, el desafío de encontrar un conjunto de datos extenso y relevante que incluya personas con rasgos faciales latinos, especialmente de Brasil, es considerable. Para este trabajo, se seleccionó un conjunto de datos que abarca una amplia gama de condiciones. Este conjunto incluye imágenes de personas en diversas poses, con diferentes expresiones faciales y bajo variadas condiciones de iluminación, sombras, y accesorios, como lentes, lunares y tipos de peinado. El conjunto de datos utilizado para el entrenamiento de la red VGG-16 es el DigiFace1M [Bae et al., 2023b], el análisis se centrará en 10 clases seleccionadas aleatoriamente de este conjunto, cada una con aproximadamente 72 imágenes por clase. Los dataset YALE y Extended Yale B, son utilizados para el uso de PCA/LDA y así medir el rendimiento de la solución.

4.1.1 DigiFace1M

Es un dataset de imágenes de rostros generadas sintéticamente mediante Inteligencia Artificial Genética, como se describe en el documento oficial [Bae et al., 2023a]. El dataset está compuesto por setecientas veinte mil imágenes, distribuidas entre diez mil identidades (clases), con setenta y dos imágenes por identidad. Para cada identidad, se seleccionaron cuatro conjuntos diferentes de accesorios, y se generaron dieciocho imágenes para cada conjunto. Este dataset ofrece varias ventajas significativas para el desarrollo y la evaluación de modelos de reconocimiento facial. Su tamaño y variedad, con 720000 imágenes, permite entrenar modelos con una amplia gama de datos, mejorando así su capacidad de generalización. Al ser un dataset sintético, DigiFace1M incluye una di-

versidad de condiciones de iluminación, poses, expresiones faciales y otros factores que afectan el reconocimiento facial, lo que aumenta la robustez de los modelos frente a variaciones reales en los datos. Además, los datos generados ofrecen la ventaja de evitar los sesgos presentes en conjuntos de datos de rostros reales, lo que contribuye a resultados más equilibrados. Al no involucrar imágenes de personas reales, se eliminan los desafíos relacionados con la privacidad y la necesidad de obtener el consentimiento, un aspecto que suele ser complejo al crear conjuntos de datos a gran escala. Esto simplifica considerablemente el proceso de recopilación de datos y permite un enfoque más controlado en la generación de las imágenes. La figura 9 es una muestra de las distintas imágenes y poses generadas para un individuo.



Figura 9: Ejemplos de imágenes del dataset DigiFace1M

Fuente: [Bae et al., 2023a]

Una vez seleccionado el dataset para esta etapa inicial de entrenamiento del modelo VGG-16 y posterior extracción de características, procedemos a su organización y separación en los diferentes conjuntos para la construcción del modelo: entrenamiento, validación y prueba. Este conjunto está compuesto por el 70% de las imágenes del dataset original, seleccionadas de manera proporcional en cada clase ,es decir, se mantiene la misma cantidad y variedad de clases que en el dataset original, pero con un 70% de las muestras disponibles en cada clase, destinado a ser el conjunto de datos de entrenamiento de la red. Un 15% de las imágenes totales, también distribuidas equitativamente entre las mismas clases conformar el conjunto de datos de validación, el 15% restante de las imágenes, igualmente distribuidas entre todas las clases, se reservó para evaluar la precisión y robustez del modelo final conformando el conjunto de prueba.

4.1.2 YALE

La base de datos Yale Face Database ¹ (tamaño de 6,4 MB) contiene 165 imágenes en escala de grises en formato GIF de 15 individuos reales. Hay 11 imágenes por sujeto, cada una correspondiente a una expresión facial o configuración diferente: luz centrada, con gafas, feliz, luz a la izquierda, sin gafas, normal, luz a la derecha, triste, somnoliento,

¹<https://www.kaggle.com/datasets/zerghamwarraich1/yale-images-dataset-for-image-classifications/data>



sorprendido y guiño.

Este dataset fue seleccionado para la etapa de reducción de dimensiones mediante PCA/LDA debido a su amplia aplicabilidad y eficacia, como se ha analizado en trabajos relacionados. La elección de este conjunto de datos es representativa de un escenario real en el que se pretende otorgar permisos de acceso a un grupo de personas. Cada clase en el dataset simula a un usuario con permisos específicos dentro de un sistema de acceso. Además, se manipuló el dataset YALE original para agregar una nueva clase (individuo) con 11 imágenes adicionales, permitiendo experimentar con casos reales que involucran personas conocidas. A continuación de exemplifica en la figura 10 un extracto del dataset YALE.

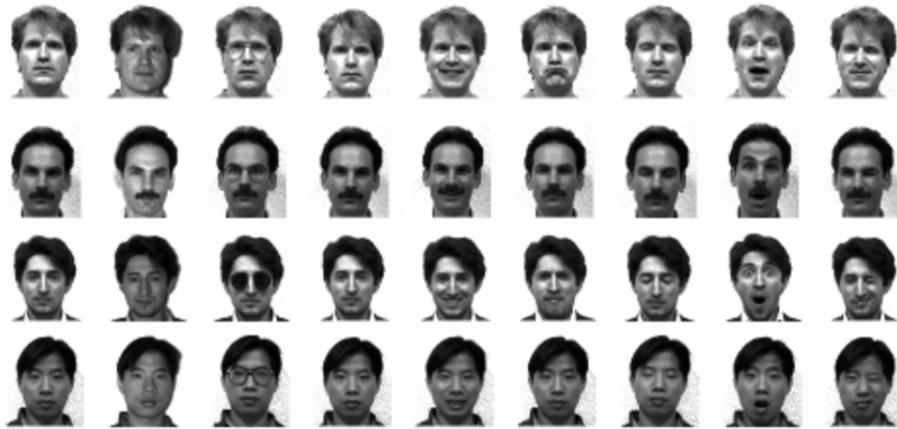


Figura 10: Ejemplos de imágenes del dataset YALE

Fuente: [Harandi et al., 2007]

4.1.3 Extended YALE B

El Extended Yale Face Database B ²fue creado para superar las limitaciones del dataset original de Yale, que tenía un número limitado de imágenes y variaciones en las condiciones de iluminación. Fue desarrollado como parte de un esfuerzo de investigación dirigido por la Universidad de Yale, con el objetivo de proporcionar un recurso más robusto para la evaluación de algoritmos de reconocimiento facial, especialmente en condiciones de iluminación variable. El dataset incluye imágenes de 38 sujetos, cada uno fotografiado bajo 9 posiciones de luz diferentes, lo que genera un total de 64 condiciones de iluminación, resultando en 2,414 imágenes en total. Las imágenes están en escala de grises y generalmente se almacenan en formato PGM (Portable Gray Map), con una resolución de 640x480 píxeles.

El uso de este dataset en la investigación se debe a que es similar a otros datasets que ya hemos utilizado, lo que nos permite mantener consistencia en cuanto a las características de las imágenes, como el hecho de que sean en escala de grises. La figura 11 ilustra como este dataset introduce una variación importante en el número de imágenes por clase y en las condiciones de iluminación, lo que es ideal para probar el modelo en un escenario distinto pero comparable. Esto nos ayuda a ver cómo se comporta el modelo cuando enfrenta situaciones similares pero con algunas diferencias clave, especialmente en términos de iluminación, lo que refuerza la validez de las pruebas



Figura 11: Ejemplos de imágenes de Extended YALE B dataset

Fuente: [Jacob et al., 2001]

4.1.4 Tratamiento de los datos

El flujo de trabajo desarrollado para entrenar un modelo de clasificación de imágenes se basó en el uso de *Transfer Learning* utilizando el modelo VGG16, aplicando técnicas de *fine-tuning* como la congelación selectiva de capas, la adición de capas densas con regularización y *dropout*, el ajuste dinámico de la tasa de aprendizaje, y el uso de *early stopping* y *model checkpointing* para la evaluación del rendimiento.

Para la carga y preparación de imágenes, se configuraron generadores de datos que permitieron cargar las imágenes en memoria de manera eficiente y aplicar aumentos de

²<https://www.kaggle.com/datasets/souvadrahati/extended-yale-dataset-b/data>

datos en tiempo real. Estas operaciones incluyen el re-escalado de los píxeles de las imágenes a un rango de [0, 1], rotaciones aleatorias de hasta 20 grados, desplazamientos horizontales y verticales con un rango de 0.2, así como cortes (*shear*) y *zoom* aleatorios, los cuales alteran la perspectiva y la escala de las imágenes. Se ajusta el brillo dentro de un rango de [0.8, 1.2] para simular diferentes condiciones de iluminación. Finalmente, se utiliza un modo de llenado *nearest* para completar los píxeles fuera de los límites tras las transformaciones geométricas (como rotaciones o desplazamientos) tomando el valor del píxel más cercano dentro de los límites originales de la imagen. Estas técnicas permiten al modelo que sea entrenado con un conjunto de imágenes más variado, mejorando su capacidad para generalizar y reconocer patrones en datos que no han sido observados anteriormente.

4.2 Entrenamiento de la red VGG-16

El objetivo del entrenamiento es hacer que el modelo extraiga características faciales relevantes para su posterior procesamiento. Se utilizó el modelo VGG16 preentrenado de *tensorflow.keras* con los pesos de *ImageNet*, excluyendo sus capas superiores lo que permitió preservar el conocimiento (pesos) aprendidos en *ImageNet* mientras se adaptaba el modelo a las nuevas tareas de clasificación. Se entrenó la red VGG16 como un problema clásico de clasificación, añadiendo capas densas con funciones de activación ReLU y softmax para clasificar las imágenes en las clases definidas por el dataset DigiFace1M. Una vez completado este entrenamiento, se eliminaron las capas densas adicionales, reteniendo únicamente las características clave extraídas por las capas convolucionales del modelo, que son fundamentales para la identificación de personas, la figura 12 ilustra la red resultante.



Para obtener esta red VGG resultante, se congelaron las capas del modelo para preservar su conocimiento preentrenado. Las capas congeladas incluyeron las del primer bloque de convolución (block1_conv1, block1_conv2), del segundo bloque (block2_conv1, block2_conv2), del tercer bloque (block3_conv1, block3_conv2, block3_conv3), del cuarto bloque (block4_conv1, block4_conv2, block4_conv3) y del quinto bloque de convolución (block5_conv1, block5_conv2) dejando sin congelar la última capa de convolución de VGG16 (block5_conv3) sumado a las capas densas añadidas posteriormente, permitiendo de esta manera que se entrenen durante el proceso de fine-tuning del modelo creado.

El modelo se compiló utilizando el optimizador Adam y la función de pérdida de entropía cruzada categórica. Durante el entrenamiento, se utilizaron datos generados aplicando *callbacks* que monitorearon y ajustaron el rendimiento del modelo. Estos *callbacks* incluyeron *ModelCheckpoint*, que permitió guardar el mejor modelo basado en la precisión de validación; *EarlyStopping*, que detuvo el entrenamiento si la precisión de validación no mejoraba durante un número específico de épocas (50); y *ReduceLROnPla-*

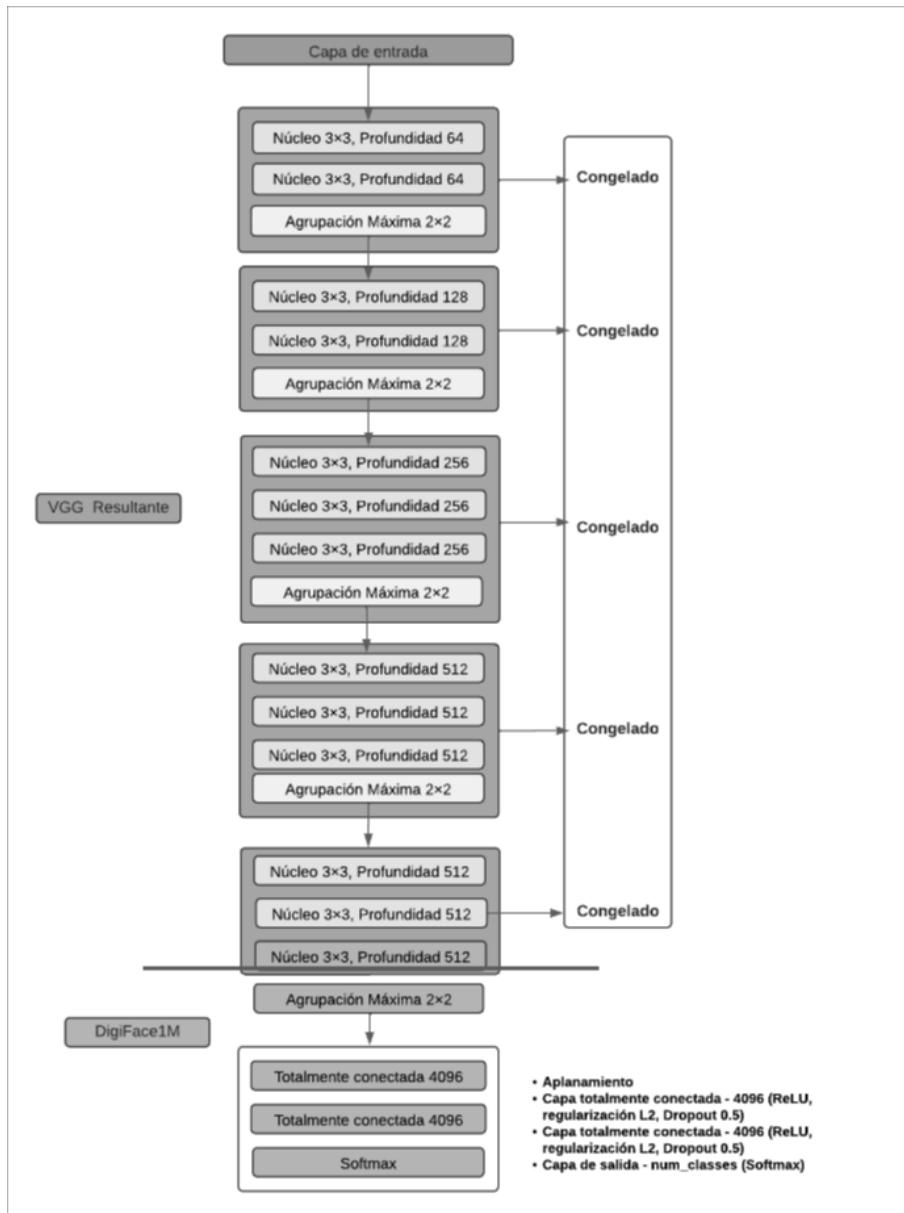


Figura 12: Diagrama VGG entrenado con sus capas congeladas pre entrenadas (celeste) para un entrenamiento posterior con el dataset DigiFace1M (rosa), teniendo como salida final el resultado del corte (linea roja) de la VGG en la capa block5_conv3

Fuente: Elaboración propia

teau, que permite un ajuste dinámico de la taza de aprendizaje, favoreciendo la convergencia del modelo. Se utilizan técnicas de regularización como *dropout* y *early stopping* para prevenir el sobreajuste, y realizamos un ajuste de hiperparámetros para optimizar el rendimiento del modelo, ajustando la tasa de aprendizaje, el tamaño del *batch* a 32 y el número de *epochs* a 120.

Se ha realizado un corte en el modelo VGG16 antes de la capa de pooling final, específicamente después de la capa *block5_conv3* siendo esta la capa de salida de nuestro modelo pre-entrenado VGG16. Este corte permite obtener un mapa de características de tamaño $7 \times 7 \times 512$ para cada imagen de entrada. En este punto del modelo, cada imagen se representa mediante un tensor de 25,088 elementos ($7 * 7 * 512$). Este vector de características se utiliza como base para realizar otras operaciones, como aplicar PCA o LDA, según los objetivos del trabajo en reconocimiento facial.

4.3 Reducción de dimensiones

Se procede a aplicar PCA utilizando la biblioteca *sklearn* para reducir la dimensionalidad de los mapas de características del modelo VGG16 anteriormente entrenado y rectificado -sin las capas de clasificación-. Se seleccionaron cuidadosamente distintas cantidades de componentes principales que capturaron la variabilidad más significativa en los datos, con el objetivo de encontrar el mejor valor para cada uno de los algoritmos utilizados, permitiendo así obtener una representación eficiente de las imágenes.

La estandarización de las características extraídas del conjunto de datos de entrenamiento y prueba es un paso crucial para que estas tengan media cero y varianza uno, lo que mejora la convergencia y eficiencia de los algoritmos posteriores. Después de estandarizar las características, se aplica PCA para reducir la dimensionalidad del espacio de características. En este proceso, se realizó una evaluación exhaustiva del número de componentes principales para asegurar que se capturara la mayor variabilidad posible de los datos mientras se minimizaba la pérdida de información. Para ello, se probaron diferentes números de componentes principales, desde 1 hasta 50. Este análisis se llevó a cabo mediante un ciclo *for* que ajustaba un modelo de PCA para cada número de componentes. Posteriormente, los datos transformados se utilizaban como entrada para un modelo de LDA, cuyo número de componentes estaba limitado por el número de clases presentes en el conjunto de entrenamiento. Este enfoque permitió seleccionar, para la mayoría de los casos estudiados, un valor óptimo de 26 componentes PCA y 14 componentes para LDA, logrando un equilibrio entre la reducción de dimensionalidad y la preservación de información clave.

Se utiliza LDA como un análisis complementario para refinar la selección del número de componentes. LDA no solo maximiza la separabilidad entre clases al proyectar los datos en un espacio de menor dimensión, sino que también se emplea para evaluar cómo

la elección del número de componentes principales impacta en la discriminación entre clases. Este enfoque permite validar y afinar la selección del número de componentes determinados por PCA, garantizando que las características resultantes no solo capturen la mayor variabilidad, sino que también sean las más discriminativas para la tarea de clasificación.

Una vez completado el entrenamiento de los modelos PCA y LDA, se procedió a guardar estos modelos para su posterior uso, asegurando la reproducibilidad de los resultados y permitiendo su aplicación a nuevos datos y en dispositivos finales sin necesidad de reentrenamiento. El modelo PCA/LDA entrenado se guardó en un archivo denominado `pca_model.pkl` utilizando la biblioteca *pickle*, que permite la serialización de objetos de *Python*.

4.4 SVM y Medidas de similitud

Una vez extraídas y reducidas las características de las imágenes en sus componentes principales, se propone investigar tanto el método SVM cuanto medidas de similitud en la comparación de imágenes y determinación de clases o individual como representado en la Figura 13

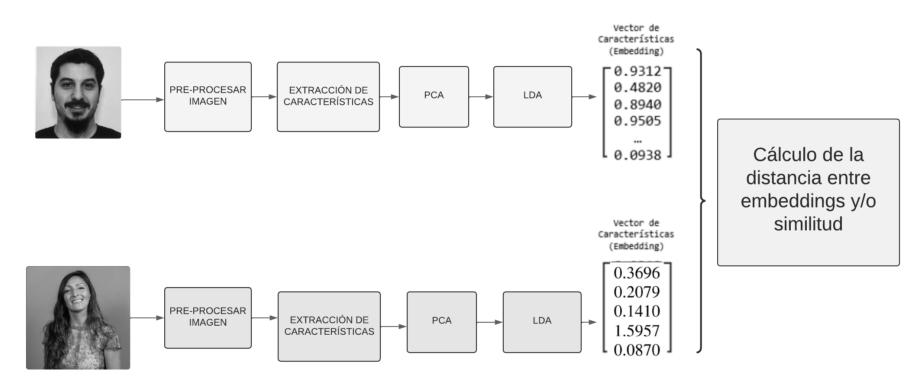


Figura 13: Ejemplo de generación de vectores de características y cálculo de distancias para comparación de rostros

Fuente: Elaboración propia

Para cada extracción de características y la aplicación de PCA y LDA, se emplearon varios algoritmos de comparación de distancias de vectores con el fin de obtener la menor distancia posible y así predecir con mayor exactitud la clase a la que pertenece el individuo. La clase del punto de entrenamiento más cercano se asigna como la predicción para cada punto de prueba. Este enfoque simple pero efectivo toma como idea central que distintos puntos similares en el espacio de características transformadas deben pertenecer a la misma clase.

Los métodos utilizados para la comparación de distancias y clasificación fueron:

- **Algoritmos de comparación de distancias:**

- Distancia Coseno
- Distancia Euclidiana
- Distancia Mahalanobis

- **Clasificador:**

- *Support Vector Machine (SVM)*

Inicialmente, en este trabajo se evaluó la opción de modificar el conjunto de datos original de manera que se crearan subconjuntos a partir de un gran dataset (DigiFace1M completo). En cada etapa de entrenamiento, se utilizaron conjuntos de datos de entrenamiento, validación y prueba, tanto en el entrenamiento del modelo VGG para realizar transferencia de aprendizaje como en el ajuste que realiza la extracción de características mediante PCA. La idea de crear subconjuntos era poder utilizar el mismo dataset pero con diferentes configuraciones para el entrenamiento de nuestra red VGG16 y el entrenamiento de PCA/LDA.

Sin embargo, se decidió emplear un nuevo conjunto de datos: YALE Dataset, para la etapa de ajuste de la extracción de características con PCA/LDA. Este nuevo dataset ha demostrado ser de gran relevancia a lo largo de varios estudios, según lo analizado en el estado del arte y ejemplificado en la tabla de comparación de modelos anteriores. Varios autores han utilizado el dataset YALE en sus investigaciones, debido a su representatividad y a las características que ofrece. La evaluación con este dataset mostró un rendimiento significativamente mejor en comparación con escenarios anteriores, proporcionando resultados más elevados y acercándose más a un posible escenario real.

Adicionalmente, se amplió el análisis incluyendo el Extended Yale B dataset, conocido por su mayor complejidad debido a las variaciones extremas en condiciones de iluminación. Esta extensión permitió una evaluación más exhaustiva de los modelos PCA/LDA, confirmando su robustez en entornos más desafiantes. Los resultados obtenidos con Extended Yale B reforzaron las pruebas ofreciendo una perspectiva más realista de la aplicabilidad de los modelos en situaciones del mundo real.

5 RESULTADOS Y DISCUSIONES

5.1 Entrenamiento de la red VGG

Los resultados presentados se centran principalmente en el la precisión y el *accuracy* (Aciertos), en la precisión de la inferencia, y la métrica *f1-score* para todos los casos en que se trabaje con la aplicación de PCA/LDA. En este punto tenemos que diferenciar y comprender los valores obtenidos en las distintas etapas del pipeline, por un lado prueba del modelo VGG y por otro las reducciones de las dimensiones de las características y los valores de predicción del modelo en su esquema completo.

Se evalúa el comportamiento de la red VGG-16 para el conjunto de imágenes correspondientes al conjunto de prueba del dataset creado a partir del dataset original DigiFace1M. Obteniendo resultados de accuracy de 88% según la siguiente figura 14 la cual ilustra los resultados alcanzados en la etapa de validación.

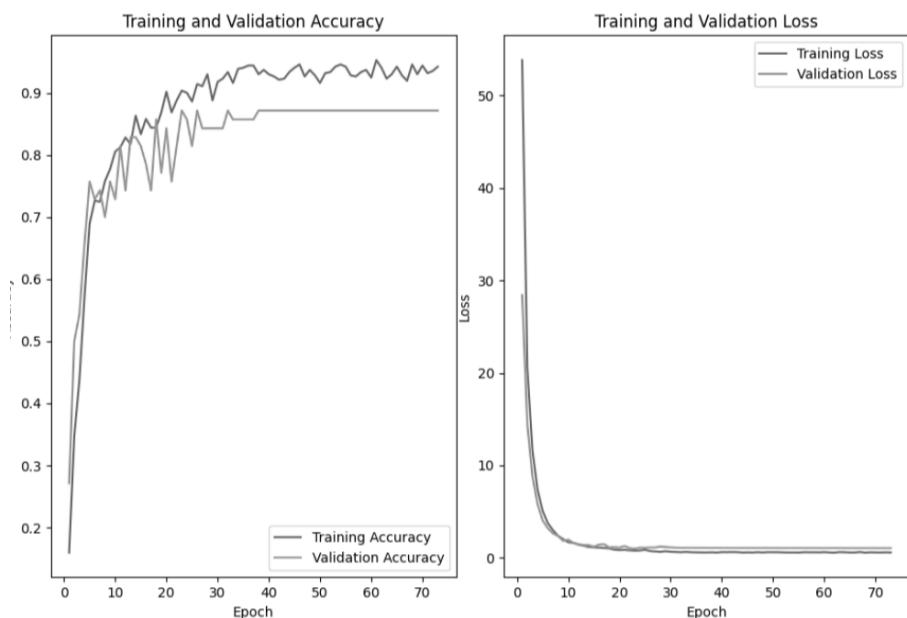


Figura 14: Accuracy Y Loss del modelo VGG16

Fuente: Elaboración propia

Tras el entrenamiento del modelo, se llevó a cabo una evaluación exhaustiva utilizando



el conjunto de datos de prueba, que consta de imágenes no vistas durante el proceso de entrenamiento. Esta evaluación se realizó con el objetivo de medir el rendimiento del modelo en condiciones de generalización obteniendo muy buenos resultados. El `classification_report` proporciona una evaluación detallada del rendimiento del modelo, mostrando métricas clave como precisión, *recall* y *F1-score* para cada clase dentro del conjunto de prueba. Estas métricas permiten una visión clara de la capacidad del modelo para clasificar correctamente las imágenes en sus respectivas categorías. A continuación, se presenta la imagen del informe de clasificación que resume estas métricas obtenida de nuestro código de implementación.

Classification Report:				
	precision	recall	f1-score	support
300	1.00	1.00	1.00	8
301	0.86	0.75	0.80	8
302	1.00	0.75	0.86	8
303	0.89	1.00	0.94	8
304	1.00	1.00	1.00	8
305	0.89	1.00	0.94	8
306	0.89	1.00	0.94	8
307	1.00	1.00	1.00	8
308	1.00	1.00	1.00	8
309	1.00	1.00	1.00	8
 accuracy				0.95
macro avg				0.95
weighted avg				0.95

Figura 15: Evaluación conjunto de pruebas

Fuente: Elaboración propia

5.2 Extracción de características - PCA/LDA

Los resultados de la aplicación de PCA y LDA en este dataset revelaron mejoras en el rendimiento del modelo, evidenciando la efectividad de estas técnicas para manejar la variabilidad en las imágenes faciales.

El gráfico de la figura 16 muestra cómo cada componente principal captura una proporción de la varianza total de los datos originales, ascendiendo rápidamente al principio y luego aplanándose a medida que se agregan más componentes. PCA es útil para reducir la dimensionalidad, pero es menester hacer una buena selección en el número de componentes para asegurar que no se comprometa la capacidad predictiva del modelo.

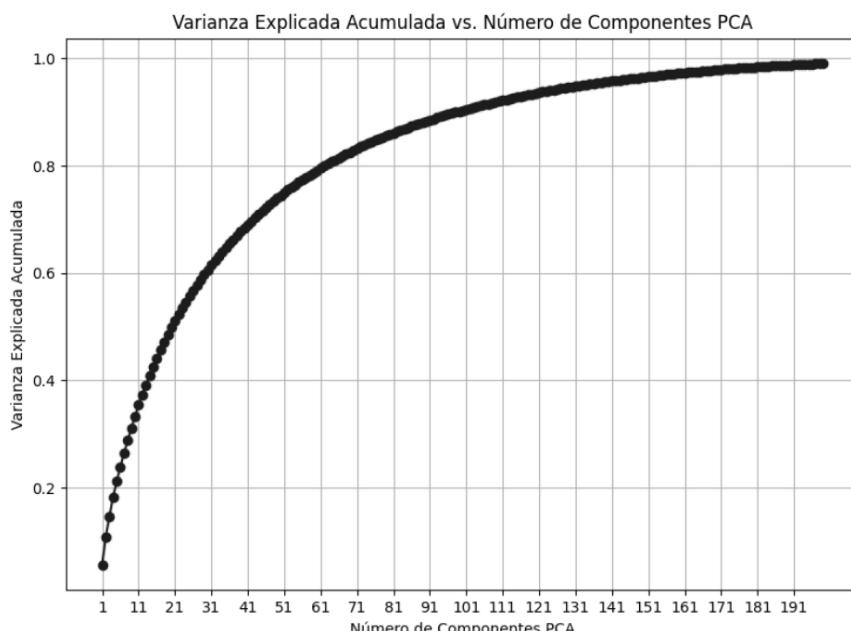


Figura 16: Relación Número de componentes PCA vs Variabilidad de los datos
Fuente: Elaboración propia

En este estudio se puede apreciar como el hecho de aplicar LDA complementa a PCA al maximizar la separación entre clases, ayudando a mitigar el ruido que PCA podría no eliminar.

Si bien la combinación de PCA y LDA puede ser poderosa, es crucial no asumir que un mayor número de componentes siempre resultará en una mejor precisión o accuracy. La relación entre la varianza explicada y el rendimiento del modelo es compleja, y en este trabajo se ha concluido, a través de observación, ensayo y error, que el número óptimo de componentes PCA no es necesariamente proporcional a una mejora en la evaluación del accuracy. Este hallazgo subraya la importancia de validar cuidadosamente el número de componentes mediante técnicas de validación. La figura 17 nos ayuda a entender la relación entre PCA y la selección de numero de componentes para medir las inferencias mediante el cálculo de distancia euclíadiana cuadrada.

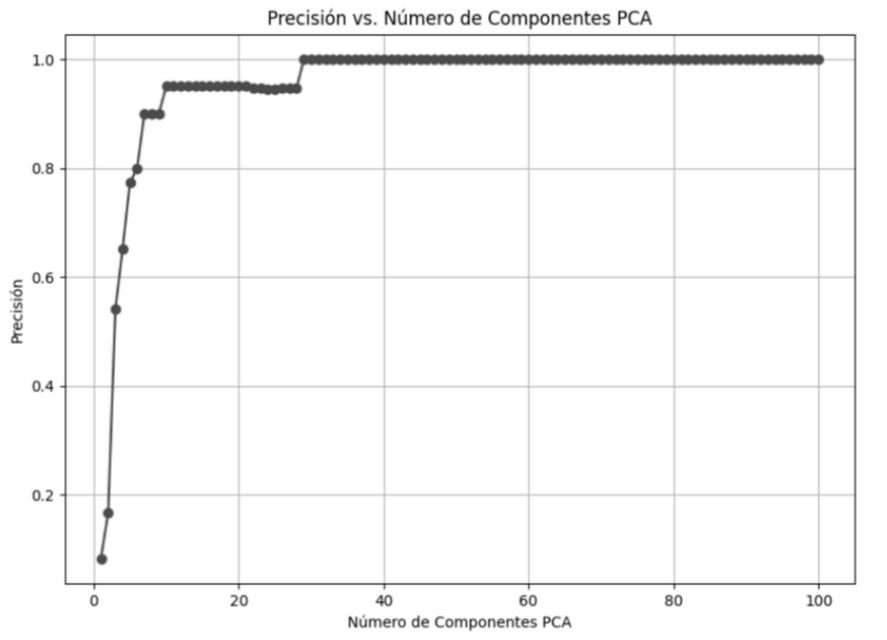


Figura 17: Número de componentes PCA vs Precisión

Fuente: Elaboración propia

5.3 Evaluación

Al evaluar el rendimiento de diferentes métodos de clasificación en varios conjuntos de datos, se han observado algunas tendencias notables. Para el dataset YALE, el uso SVM con solo 6 componentes principales de PCA logró una alta precisión del 96%. Sin embargo, cuando se incluyó el dataset YALE B completo, la precisión del modelo bajó considerablemente a un 73%. Se interpreta que aunque el modelo SVM funciona bien con un número reducido de componentes en conjuntos de datos simples, tiene dificultades con datos más complejos.

Por otro lado, la métrica de distancia coseno demuestra un buen desempeño en el manejo de conjuntos de datos más grandes y complejos. Con el dataset YALE, esta métrica alcanzó una precisión del 95% usando 26 componentes PCA. En el dataset YALE B Completo y en las combinaciones de YALE y YALE B, la precisión se mantuvo alta e incluso llegó al 100% en algunos casos. Esto sugiere que la distancia coseno es muy eficaz en espacios de alta dimensión, donde un mayor número de componentes mejora la representación.

La distancia euclíadiana también mostró un rendimiento excelente en dimensiones altas, con una precisión del 99% en YALE y hasta el 100% en las combinaciones de YALE y YALE B usando 26 componentes PCA.

El método Mahalanobis presentó resultados más modestos. Con una precisión del 82% en YALE usando 9 componentes PCA, y una caída a 76% en YALE B Completo, muestra que aunque es útil, la distancia Mahalanobis no es la mejor opción para conjuntos

de datos complejos o con muchas dimensiones.

Para un subconjunto de 24 fotos del dataset YALE, se obtuvieron resultados excepcionales, con una precisión del 100% usando distancia coseno, euclíadiana y Mahalanobis con 30, 26 y 9 componentes PCA, respectivamente. Finalmente, al combinar los datasets YALE y YALE B, tanto la distancia coseno como la distancia euclíadiana demostraron ser muy efectivas, alcanzando precisiones del 99% y 100% con 30 componentes PCA y 15 componentes LDA.

Estos hallazgos destacan la importancia de ajustar tanto el número de componentes PCA como el método de clasificación según las características del dataset. La selección adecuada de estos parámetros es crucial para optimizar el rendimiento del modelo. La figura 18 ilustra la matriz de confusión para la evaluación de similitud mediante distancia de coseno para el conjunto de datos YALE + YALE B.

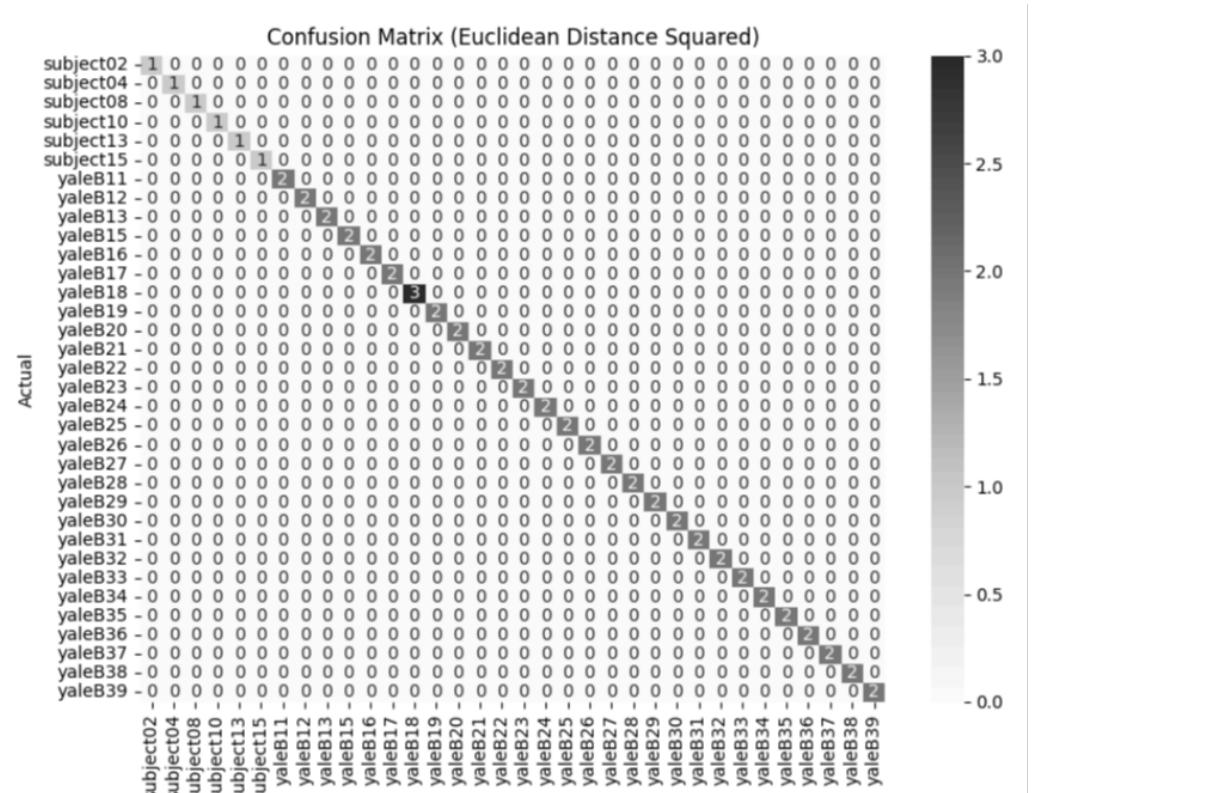


Figura 18: Matriz de Confusión - Distancia Euclíadiana Cuadrada

Fuente: Elaboración propia

A continuación se presenta una matriz de resultados alcanzados con valores extremadamente notorios en cuanto a su precisión y acierto. Las tablas 2, 3 4 y 5 describen los valores alcanzados en la comparación de los distintos algoritmos y datasets empleados.

Método	Num. PCA	Num. LDA	Acc %	Precisión	Recall	F1-Score
SVM	6	6	96	0.98	0.95	0.96
Mahalanobis	9	14	82	0.91	0.90	0.91
Coseno	26	14	95	1.00	0.98	0.99
Euclíadiana	26	14	99	1.00	1.00	1.00

Tabla 2: Comparación de Componentes y Métricas para dataset YALE

Método	Num. PCA	Num. LDA	Acc %	Precisión	Recall	F1-Score
Coseno	30	15	92	0.95	0.97	0.98
Euclíadiana	26	14	92	0.95	0.98	0.99
Mahalanobis	9	24	76	0.79	0.79	0.78
SVM	6	6	73	0.78	0.78	0.78

Tabla 3: Comparación de Componentes y Métricas para dataset Ext. YALE B

Método	Num. PCA	Num. LDA	Acc %	Precisión	Recall	F1-Score
Coseno	30	15	100	1.00	0.99	0.99
Euclíadiana	26	14	100	1.00	1.00	1.00
Mahalanobis	9	24	100	1.00	0.98	0.99
SVM	6	6	74	0.75	0.78	0.76

Tabla 4: Comparación de Componentes y Métricas para dataset Ext. YALE B (24 fotos por clase)

Método	Num. PCA	Num. LDA	Acc %	Precisión	Recall	F1-Score
Coseno	30	15	99	1.00	0.99	1.00
Euclíadiana	26	14	100	1.00	1.00	1.00
Mahalanobis	9	24	94	0.98	0.95	0.96
SVM	6	14	85	0.86	0.84	0.82

Tabla 5: Comparación de Componentes y Métricas para dataset YALE + Ext. YALE B

6 CONCLUSIONES Y TRABAJOS FUTUROS

El uso de la red VGG16 en transferencia de aprendizaje, con el corte en la capa block5_conv3, ha demostrado ser una estrategia efectiva para la extracción de características de alto nivel en proyectos de clasificación de imágenes. Esta capa intermedia proporciona una clara y efectiva representación de las imágenes, lo cual es crucial para la reducción efectiva de las dimensiones de la salida de ésta capa. La aplicación de PCA y LDA sobre estas características extraídas permite una compresión significativa de la información, preservando la variabilidad esencial mientras se simplifica el modelo. La combinación de estas técnicas de reducción de dimensiones con la red VGG16 no solo mejora la eficiencia del modelo, sino que también facilita su aplicación en escenarios prácticos, donde la gestión de recursos y la precisión en la clasificación son variables consideradas de interés.

En este trabajo, se ha demostrado la factibilidad y eficacia de entrenar una red VGG mediante transfer learning utilizando un dataset generado digitalmente. Este enfoque es altamente ventajoso, ya que elimina la necesidad de utilizar cientos de miles de imágenes de personas reales, lo cual sería una tarea compleja. La capacidad de entrenar modelos precisos sin recurrir a datos sensibles es un avance significativo, y es importante destacar que, según la bibliografía consultada, no existen estudios previos que validen el uso de datasets digitales en este contexto, lo que subraya relevancia de este enfoque.

Otro aspecto relevante es la simplicidad y eficiencia del proceso para actualizar la "base de datos" de usuarios con acceso. Al tratarse de un modelo basado en la reducción dimensional mediante PCA/LDA, la actualización se reduce a modificar los archivos de escalado resultantes de estas técnicas, lo que lo convierte en un procedimiento rápido y sencillo. Esto no solo facilita el mantenimiento del sistema, sino que también asegura que los modelos de reconocimiento facial se mantengan actualizados con un mínimo esfuerzo computacional.

En conclusión, la combinación de técnicas de transfer learning con datasets digitales y la eficiencia en la actualización de modelos representa un avance significativo en la implementación de sistemas de reconocimiento facial. Este enfoque no solo garantiza una alta precisión y aciertos, sino que también optimiza el recall, evidenciado por el rendimiento

sobresaliente en la identificación de usuarios, con puntuaciones consistentemente altas en precisión y recall en diversos métodos de clasificación.

Para trabajos futuros, ahora que se ha demostrado el éxito de este enfoque, se podría ampliar la investigación aplicando esta misma técnica con el uso de otras redes neuronales que, aunque más livianas, presentan una mayor complejidad en su manipulación. Además, se plantea estudiar la extracción de características utilizando redes U-Net, con el objetivo de optimizar aún más el modelo. Esto permitiría su implementación en dispositivos con recursos cada vez más limitados, orientando el desarrollo hacia el campo de TinyML, donde la eficiencia y la capacidad de procesamiento en el dispositivo son fundamentales sin perder de vista las precauciones a nivel de la seguridad informática y la seguridad de la información a la hora de procesar y almacenar los datos.

BIBLIOGRAFIA CONSULTADA

- [Abdullah, 2012] Abdullah, M. (2012). Optimizing face recognition using PCA. *Int. J. Artif. Intell. Appl.*, 3(2):236–231.
- [Anowar et al., 2021] Anowar, F., Sadaoui, S., and Selim, B. (2021). Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kPCA, lda, mds, svd, lle, isomap, le, ica, t-sne). *Computer Science Review*, 40:100378.
- [Auria and Moro, 2008] Auria, L. and Moro, R. A. (2008). Support vector machines (SVM) as a technique for solvency analysis. *SSRN Electron. J.*
- [Bae et al., 2023a] Bae, G., de La Gorce, M., Baltrušaitis, T., Hewitt, C., Chen, D., Valentin, J., Cipolla, R., and Shen, J. (2023a). Digiface-1m: 1 million digital face images for face recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3526–3535.
- [Bae et al., 2023b] Bae, G., de La Gorce, M., Baltrušaitis, T., Hewitt, C., Chen, D., Valentin, J., Cipolla, R., and Shen, J. (2023b). Digiface-1m: 1 million digital face images for face recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3526–3535.
- [Barnouti et al., 2016] Barnouti, N. H., Al-Dabbagh, S. S. M., Matti, W. E., and Naser, M. A. S. (2016). Face detection and recognition using Viola-Jones with PCA-LDA and square euclidean distance. *International Journal of Advanced Computer Science and Applications*, 7(5).
- [Bez et al., 2003] Bez, M. R., Igelmo, A., and Perales, F. J. (2003). *Análisis Estadístico Multivariante de Medidas de la Cara para Aplicaciones de Animación Facial*.
- [Bledsoe and Browning, 1959] Bledsoe, W. W. and Browning, I. (1959). Pattern recognition and reading by machine. In *Papers presented at the December 1-3, 1959, eastern joint IRE-AIEE-ACM computer conference on - IRE-AIEE-ACM '59 (Eastern)*, New York, New York, USA. ACM Press.

- [Castrillón et al., 2008] Castrillón, W. A., Alvarez, D. A., and López, A. F. (2008). Técnicas de extracción de características en imágenes para el reconocimiento de expresiones faciales. *Scientia Et Technica*, 14(38):7–12.
- [Cerna et al., 2013] Cerna, L. R., Camara-Chavez, G., and Menotti, D. (2013). Computer vision, and pattern recognition (IPCV) (p. 1). the steering committee of the world congress in computer science. In *Proceedings of the International Conference on Image Processing*.
- [Chen and Haoyu, 2019] Chen, H. and Haoyu, C. (2019). Face recognition algorithm based on VGG network model and SVM. *J. Phys. Conf. Ser.*, 1229(1):012015.
- [Dar and Palanivel, 2021] Dar, S. A. and Palanivel, S. (2021). Performance evaluation of convolutional neural networks (CNNs) and VGG on real time face recognition system. *Adv. Sci. Technol. Eng. Syst. J.*, 6(2):956–964.
- [Domínguez Pavón, 2017] Domínguez Pavón, S. (2017). *Reconocimiento facial mediante el Análisis de Componentes Principales (PCA)*. (*Trabajo Fin de Grado Inédito*). Sevilla.
- [Espinoza Olguín and Guillen, 2015] Espinoza Olguín, D. E. and Guillen, P. I. (2015). *Reconocimiento facial*.
- [Faruqe and Hasan, 2009] Faruqe, M. O. and Hasan, M. A. M. (2009). Face recognition using PCA and SVM. In *2009 3rd International Conference on Anti-counterfeiting, Security, and Identification in Communication*. IEEE.
- [Fisher, 1936] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Ann. Eugen.*, 7(2):179–188.
- [Goel et al., 2021] Goel, R., Mehmood, I., and Ugail, H. (2021). A study of deep learning-based face recognition models for sibling identification. *Sensors (Basel)*, 21(15):5068.
- [Harandi et al., 2007] Harandi, M. T., Ahmadabadi, M. N., and Araabi, B. N. (2007). A hierarchical face identification system based on facial components. In *2007 IEEE/ACS International Conference on Computer Systems and Applications*. IEEE.
- [Heidari and Fouladi-Ghaleh, 2020] Heidari, M. and Fouladi-Ghaleh, K. (2020). Using siamese networks with transfer learning for face recognition on small-samples datasets. In *2020 International Conference on Machine Vision and Image Processing (MVIP)*. IEEE.

- [Hong et al., 2021] Hong, D., Zheng, Y.-Y., Xin, Y., Sun, L., Yang, H., Lin, M.-Y., Liu, C., Li, B.-N., Zhang, Z.-W., Zhuang, J., Qian, M.-Y., and Wang, S.-S. (2021). Genetic syndromes screening by facial recognition technology: VGG-16 screening model construction and evaluation. *Orphanet J. Rare Dis.*, 16(1):344.
- [Hsu and Lin, 2002] Hsu, C.-W. and Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425.
- [Jacob et al., 2001] Jacob, M., Blu, T., and Unser, M. (2001). An exact method for computing the area moments of wavelet and spline curves. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(6):633–642.
- [Kar et al., 2012] Kar, A., Bhattacharjee, D., Basu, D. K., Nasipuri, M., and Kundu, M. (2012). *A Gabor-Block-Based Kernel Discriminative Common Vector Approach Using Cosine Kernels for Human Face Recognition*. Indian Statistical Institute, Kolkata 700108, India, volume 700108. Indian Statistical Institute, Kolkata; Kolkata; India.
- [Kaur and Himanshi, 2015] Kaur, R. and Himanshi, E. (2015). Face recognition using principal component analysis. en. In *IEEE International Advance Computing Conference (IACC)*.
- [Kurita, 2019] Kurita, T. (2019). *Principal Component Analysis (PCA)*, pages 1–4. Springer International Publishing, Cham.
- [Liu et al., 2021] Liu, H., Mo, Z.-H., Yang, H., Zhang, Z.-F., Hong, D., Wen, L., Lin, M.-Y., Zheng, Y.-Y., Zhang, Z.-W., Xu, X.-W., Zhuang, J., and Wang, S.-S. (2021). Automatic facial recognition of Williams-Beuren syndrome based on deep convolutional neural networks. *Front. Pediatr.*, 9:648255.
- [Loncomilla, 2016] Loncomilla, P. (2016). Deep learning: Redes convolucionales. *Recuperado de <https://ccc.inaoep.mx/~pgomez/deep/presentations>*.
- [Lorenzo, 2019] Lorenzo, J. (2019). Distancia de mahalanobis.
- [Lozano, 2020] Lozano, G. (2020). Sistema de reconocimiento facial con deep learning (trabajo fin de grado, grado en ingeniería informática). *Universidad*.
- [Lv et al., 2023] Lv, X., Su, M., and Wang, Z. (2023). Face recognition method under adaptive image matching and dictionary learning algorithm. *Comput. Intell. Neurosci.*, 2023:8225630.
- [López-Sánchez, 2022] López-Sánchez, M. (2022). Introducción a los sistemas de reconocimiento facial utilizando deep learning.

- [Mejía-Campos et al., 2017] Mejía-Campos, R., Néjer-Haro, D., Recalde-Avincho, S., Rosero-Montalvo, P., and Peluffo-Ordóñez, D. (2017). Face detection and classification using eigenfaces and principal component analysis: Preliminary results. In *2017 International Conference on Information Systems and Computer Science (INCISCOS)*, pages 309–315.
- [Mendoza Olguín et al., 2019] Mendoza Olguín, G. E., Laureano de Jesús, Y., and Pérez de Celis Herrero, M. d. l. C. (2019). Métricas de similaridad y evaluación para sistemas de recomendación de filtrado colaborativo. *Revista de Investigación en Tecnologías de la Información*, 7(14):224–240.
- [Morcillo Vizuete, 2021] Morcillo Vizuete, F. (2021). *Estudio de las variables a considerar para el desarrollo de un modelo viable de aplicación de un sistema de identificación facial segura en un entorno de organizaciones de servicios (Doctoral dissertation)*.
- [Muñoz Plá, 2023] Muñoz Plá, M. (2023). *Biometría facial con Deep Learning: Reconocimiento facial con redes neuronales convolucionales (Master's thesis)*. Hernández Jiménez, E. (Tutor).
- [Núñez Ortega, 2023] Núñez Ortega, J. E. (2023). Detección visual automática de cadeñas de aisladores dañadas en líneas de transmisión a través de imágenes aéreas.
- [Oloyede et al., 2020] Oloyede, M. O., Hancke, G. P., and Myburgh, H. C. (2020). A review on face recognition systems: recent approaches and challenges. *Multimed. Tools Appl.*, 79(37-38):27891–27922.
- [Pearson, 1901] Pearson, K. (1901). LIII. on lines and planes of closest fit to systems of points in space. *Lond. Edinb. Dublin Philos. Mag. J. Sci.*, 2(11):559–572.
- [Portillo and Plata, 2008] Portillo, M. T. E. and Plata, J. A. S. (2008). P. ch. mahalanobis y las aplicaciones de su distancia estadística. *CULCyT: Cultura Científica y Tecnológica*, 5(27):13–20.
- [Prado et al., 2020] Prado, N., González-Castro, V., Alegre, E., and Fernández, E. (2020). Comparación de métodos de detección de rostros en imágenes digitales. pages 976–982.
- [Rahmad et al., 2020] Rahmad, C., Asmara, R. A., Putra, D. R. H., Dharma, I., Darmono, H., and Muhiqin, I. (2020). Comparison of Viola-Jones haar cascade classifier and histogram of oriented gradients (HOG) for face detection. *IOP Conf. Ser. Mater. Sci. Eng.*, 732(1):012038.

- [Raj, 2011] Raj, D. (2011). A realtime face recognition system using pca and various distance classifiers, CS676: Computer vision and image processing. pages 1–11.
- [Ruan et al., 2010] Ruan, C., Ruan, Q., and Li, X. (2010). Real adaboost feature selection for face recognition. In *IEEE 10th INTERNATIONAL CONFERENCE ON SIGNAL PROCESSING PROCEEDINGS*. IEEE.
- [Salih et al., 2019] Salih, N., Akgün, Ö., and Hussein, N. (2019). Building smart algorithm to extract features of topographic images of a human eye.
- [Salih and Hussein, 2019] Salih, N. and Hussein, N. (2019). Building Smart Algorithm To Extract Features Of Topographic Images Of A Human Eye.
- [Sánchez et al., 2019] Sánchez, W., Sigua, E., and Valladarez, D. (2019). Reconocimiento facial en video usando deep learning. *INTELIGENCIA ARTIFICIAL*, 1.
- [Schroff et al., 2015] Schroff, F., Kalenichenko, D., and Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- [Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition.
- [Suárez, 2014] Suárez, E. J. C. (2014). Tutorial sobre máquinas de vectores soporte (svm). *Tutorial sobre Máquinas de Vectores Soporte (SVM)*, 1:1–12.
- [Susheel Kumar et al., 2011] Susheel Kumar, K., Prasad, S., Bhaskar Semwal, V., and Tripathi, R. C. (2011). Real time face recognition using AdaBoost improved fast PCA algorithm. *Int. J. Artif. Intell. Appl.*, 2(3):45–58.
- [Tammina, 2019] Tammina, S. (2019). Transfer learning using VGG-16 with deep convolutional neural network for classifying images. *Int. J. Sci. Res. Publ. (IJSRP)*, 9(10):9420.
- [Tian et al., 2022] Tian, F., Xie, H., Song, Y., Hu, S., and Liu, J. (2022). The face inversion effect in deep convolutional neural networks. *Front. Comput. Neurosci.*, 16:854218.
- [Timo, 2004] Timo, A. (2004). Face recognition with local binary patterns. In *Euro. Conf. on Computer Vision*.
- [Toro et al., 2023] Toro, G., Chang Camacho, P. J., Saavedra Rondo, V., and Bustos Cárdenas, J. (2023). Evaluación de métodos de reconocimiento facial para análisis automático de reportes de PREY (tesis de magíster). universidad de chile, facultad de

ciencias físicas y matemáticas, departamento de ciencias de la computación. *Barrios Núñez, J. (Prof. Guía)*.

[Vapnik, 2010] Vapnik, V. (2010). *Estimation of dependences based on empirical data. Information Science and Statistics*. Springer, New York, NY.

[Wu et al., 2017] Wu, H., Xu, Z., Zhang, J., Yan, W., and Ma, X. (2017). Face recognition based on convolution siamese networks. In *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE.

[Xanthopoulos et al., 2013] Xanthopoulos, P., Pardalos, P. M., and Trafalis, T. B. (2013). Linear discriminant analysis. In *SpringerBriefs in Optimization*, pages 27–33. Springer New York, New York, NY.

[Xia et al., 2015] Xia, P., Zhang, L., and Li, F. (2015). Learning similarity with cosine similarity ensemble. *Inf. Sci. (Ny)*, 307:39–52.

[Xie et al., 2019] Xie, Z., Zhang, S., Yu, X., and Liu, G. (2019). Infrared and visible face fusion recognition based on extended sparse representation classification and local binary patterns for the single sample problem. *J. Opt. Technol.*, 86(7):408.

[Zhang et al., 2017] Zhang, Y., Wang, H., Xu, F., and Jia, K. (2017). A deep face recognition method based on model fine-tuning and principal component analysis. In *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE.

A FUNDAMENTACIÓN MATEMÁTICA

Análisis de Componentes Principales (PCA)

El Análisis de Componentes Principales (PCA) se basa en los siguientes pasos matemáticos. [Kurita, 2019]:

Paso 1: Normalización de los Datos

Primero, los datos se normalizan para tener media cero y desviación estándar uno:

$$\mathbf{X}_{normalizado} = \frac{\mathbf{X} - \mu}{\sigma} \quad (12)$$

donde:

- \mathbf{X} es la matriz de datos original.
- μ es el vector de medias de cada característica.
- σ es el vector de desviaciones estándar de cada característica.

Paso 2: Cálculo de la Matriz de Covarianza

Se calcula la matriz de covarianza \mathbf{C} :

$$\mathbf{C} = \frac{1}{n-1} \mathbf{X}_{normalizado}^T \mathbf{X}_{normalizado} \quad (13)$$

donde n es el número de muestras.

Paso 3: Descomposición en Valores Propios

La matriz de covarianza \mathbf{C} se descompone en sus valores propios (λ) y vectores propios (\mathbf{v}):

$$\mathbf{C}\mathbf{v}_i = \lambda_i \mathbf{v}_i \quad (14)$$

Paso 4: Selección de Componentes Principales

Se ordenan los vectores propios según sus valores propios de mayor a menor y se seleccionan los primeros k vectores propios.

Paso 5: Proyección de los Datos

Finalmente, los datos originales se proyectan sobre el subespacio de las componentes principales seleccionadas:

$$\mathbf{Z} = \mathbf{X}_{\text{normalizado}} \mathbf{W} \quad (15)$$

donde \mathbf{W} es la matriz de los primeros k vectores propios.

Análisis Discriminante Lineal (LDA)

El Análisis Discriminante Lineal (LDA) se basa en los siguientes pasos matemáticos. [Xanthopoulos et al., 2013]

Paso 1: Calcular la Media de Cada Clase

Para cada clase i , se calcula el vector de media μ_i :

$$\mu_i = \frac{1}{N_i} \sum_{x \in C_i} x \quad (16)$$

donde:

- N_i es el número de muestras en la clase i .
- C_i es el conjunto de muestras pertenecientes a la clase i .

Paso 2: Calcular la Media Total

Se calcula el vector de media total μ usando todas las muestras:

$$\mu = \frac{1}{N} \sum_{i=1}^k \sum_{x \in C_i} x \quad (17)$$

donde:

- N es el número total de muestras.
- k es el número de clases.

Paso 3: Calcular las Matrices de Dispersión Dentro de la Clase y Entre Clases

La matriz de dispersión dentro de la clase (\mathbf{S}_W) se calcula como la suma de las matrices de co-varianza de cada clase:

$$\mathbf{S}_W = \sum_{i=1}^k \sum_{x \in C_i} (x - \mu_i)(x - \mu_i)^T \quad (18)$$

La matriz de dispersión entre clases (\mathbf{S}_B) se calcula como la suma de las desviaciones de las medias de las clases con respecto a la media total:

$$\mathbf{S}_B = \sum_{i=1}^k N_i (\mu_i - \mu)(\mu_i - \mu)^T \quad (19)$$

Paso 4: Calcular los Vectores Propios y Valores Propios

Se resuelve el problema de valores propios para la matriz $\mathbf{S}_W^{-1}\mathbf{S}_B$:

$$\mathbf{S}_W^{-1}\mathbf{S}_B \mathbf{v} = \lambda \mathbf{v} \quad (20)$$

donde:

- λ son los valores propios.
- \mathbf{v} son los vectores propios.

Paso 5: Seleccionar los Vectores Propios

Se seleccionan los vectores propios correspondientes a los mayores valores propios para formar la matriz de transformación \mathbf{W} .

Paso 6: Transformar los Datos

Los datos originales se transforman proyectándolos sobre el subespacio definido por los vectores propios seleccionados:

$$\mathbf{Z} = \mathbf{X}\mathbf{W} \quad (21)$$

Support Vector Machine (SVM)

El Support Vector Machine (SVM) se basa en los siguientes pasos matemáticos. [Auria and Moro, 2008]

Paso 1: Hiperplano Separador

Un hiperplano en n -dimensiones puede ser escrito como:

$$\mathbf{w} \cdot \mathbf{x} - b = 0 \quad (22)$$

donde:

- \mathbf{w} es un vector normal al hiperplano.

- b es el sesgo.

Paso 2: Márgenes y Hiperplano Óptimo

El objetivo del SVM es encontrar el hiperplano que maximiza el margen entre las dos clases. El problema de optimización se formula como:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2$$

sujeto a las restricciones:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1 \quad \forall i \quad (23)$$

donde:

- y_i son las etiquetas de clase (+1 o -1).
- \mathbf{x}_i son las muestras de entrenamiento.

Paso 3: Función de Pérdida y Dualidad

La función de Lagrange se introduce para permitir errores en la clasificación:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^n \alpha_i [y_i(\mathbf{w} \cdot \mathbf{x}_i - b) - 1] \quad (24)$$

donde α son los multiplicadores de Lagrange.

Paso 4: Solución del Problema Dual

El problema dual se resuelve maximizando la función de Lagrange con respecto a α :

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad (25)$$

sujeto a:

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad y \quad \alpha_i \geq 0 \quad \forall i \quad (26)$$

Paso 5: Clasificación

Una vez que se encuentran los valores óptimos de α , el vector \mathbf{w} se puede calcular como:

$$\mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i \quad (27)$$

y la clasificación de una nueva muestra \mathbf{x} se realiza mediante:

$$f(\mathbf{x}) = sign(\mathbf{w} \cdot \mathbf{x} - b) \quad (28)$$