

Trabajo Fin de Grado

Grado en Ingeniería de las Tecnologías de
Telecomunicación

Reconocimiento facial mediante el Análisis de
Componentes Principales (PCA)

Autora: Sara Domínguez Pavón

Tutor: Rubén Martín Clemente

Dep. Teoría de la Señal y Comunicaciones
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2017



Trabajo Fin de Grado
Grado en Ingeniería de las Tecnologías de Telecomunicación

Reconocimiento facial mediante el Análisis de Componentes Principales (PCA)

Autora:

Sara Domínguez Pavón

Tutor:

Rubén Martín Clemente

Profesor titular

Dep. de Teoría de la Señal y Comunicaciones

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2017

Trabajo Fin de Grado: Reconocimiento facial mediante el Análisis de Componentes Principales (PCA)

Autora: Sara Domínguez Pavón

Tutor: Rubén Martín Clemente

El tribunal nombrado para juzgar el Trabajo arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2017

El Secretario del Tribunal

A mi familia

Agradecimientos

Con este trabajo pongo fin a una etapa muy importante y especial de mi vida. A pesar de que han sido muchos años de trabajo y sacrificio, prevalecen los buenos momentos, en los que han influido muchas personas a las que les agradezco todo lo que me han aportado.

Esta etapa no habría sido posible llevarla a cabo sin los amigos que he conocido durante ella y que me llevo para siempre, a los cuales tengo que agradecerles todo su apoyo y ayuda. También me acuerdo de todos esos profesores y compañeros de clase que han ido dejando huella en mí y que, de una manera u otra, me han conducido por las diferentes decisiones que he ido tomando.

Agradecer también a mi tutor, Rubén, por su entrega y por ayudarme a sacar este trabajo adelante. Pero sobre todo, por confiar en mí y darme la oportunidad de trabajar junto a él en uno de sus proyectos.

Finalmente, dar las gracias a mi familia por apoyarme durante todos estos años. A mis padres y hermana, por su cariño, ánimo y consejos, por alegrarse de mis triunfos sabiendo que también son los suyos, por valorar mi trabajo incluso más que yo y por haberme hecho llegar a donde estoy y a lo que soy. Y por último, y no por ello menos especial, a Ángel, por su infinita paciencia conmigo.

Sara Domínguez Pavón

Sevilla, 2017

Resumen

Los sistemas de reconocimiento facial han recibido un fuerte impulso en la actualidad gracias al avance en la tecnología. Dado que este tipo de técnicas tienen muchas aplicaciones útiles en campos muy diversos como la biometría, la clasificación de imágenes o la seguridad, se han destinado muchos esfuerzos tanto económicos como científicos para tratar de mejorarlas.

El proceso de reconocimiento facial se divide en dos tareas. La primera de ellas, la detección, comprende la localización de una o varias caras dentro de una imagen, ya sea una imagen fija o una secuencia de video. La segunda tarea, el reconocimiento, consiste en la comparación de la cara detectada con anterioridad con otras almacenadas previamente en una base de datos. Estos dos procesos no deben ser totalmente independientes, ya que un buen reconocimiento depende fuertemente de la previa detección, la cual está condicionada por la posición y orientación de la cara del sujeto con respecto a la cámara y las condiciones de iluminación.

En este trabajo se estudia, implementa y evalúa un sistema automático de reconocimiento facial, tanto para trabajar con imágenes como con videos, además de en tiempo real.

Como punto de partida se realizará un estudio de las técnicas de reconocimiento facial ya existentes en el estado del arte. Tras ese primer análisis, se seleccionará una de las técnicas analizadas para su posterior implementación en Python en un sistema capaz de detectar y reconocer rostros de personas introducidas previamente en el sistema en la fase de entrenamiento. En este caso, se implementará el método *Eigenfaces*, construido sobre técnicas de Análisis de Componentes Principales (PCA). Finalmente, se harán un conjunto de pruebas sobre diferentes bases de datos de imágenes para analizar y verificar los resultados obtenidos tras aplicar el algoritmo implementado.

Abstract

Facial recognition systems have received a strong boost at presents thanks to advances in technology. All these techniques have many useful applications and can be applied in many different areas such as biometrics, image classification or security. That's the reason why society has invested a lot of economic and investigation efforts in improving them.

Facial recognition process is divided into two tasks. The first of these, detection, comprises the location of one or more faces within an image or a video sequence. The second task, recognition, consists of the comparison of the previously detected face with others previously saved in a database. These two processes should not be totally independent, since a good recognition depends strongly on the previous detection, which is conditioned by the position and orientation of the face of the subject in respect to the camera and the lighting conditions.

In this project, an automatic real time face recognition system is studied, implemented and evaluated, both to work with images and with videos.

First of all, a study of facial recognition techniques in the state of the art will be carried out. Once we will have finished this first analysis, we will choose one of the analysed techniques in order to develop them. We will develop in Python a system able to detect and recognize faces from people who have been entered on the system previously in a training phase. In this case, the *Eigenfaces* method built over techniques of Principal Component Analysis (PCA), will be implemented. Finally, we will make some tests on different image databases to analyse and check the results got from using the algorithms developed.

Índice

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de Tablas	xvii
Índice de Figuras	xix
Notación	xxi
1 Introducción	1
1.1 <i>Motivación del trabajo</i>	1
1.2 <i>Objetivos y enfoque</i>	2
1.3 <i>Organización de la memoria</i>	2
2 Estado del arte	3
2.1 <i>Historia del reconocimiento facial</i>	3
2.2 <i>Fundamentos del reconocimiento facial</i>	4
2.3 <i>Ventajas e inconvenientes del reconocimiento facial</i>	5
2.4 <i>Aplicaciones de los sistemas de reconocimiento facial</i>	6
2.5 <i>Sistemas comerciales en la actualidad</i>	7
3 Fases de un sistema de reconocimiento facial	9
3.1 <i>Detección</i>	9
3.1.1 Algoritmo de Viola-Jones	10
3.1.2 Detección de caras	11
3.1.3 Detección de ojos	11
3.2 <i>Preprocesado</i>	12
3.3 <i>Extracción de características</i>	13
3.3.1 Técnicas basadas en apariencia	13
3.3.2 Técnicas basadas en modelos	14
3.4 <i>Comparación y decisión</i>	17
3.4.1 Medidas de similitud o distancias	17
3.4.2 Clasificadores	18

4	Análisis de Componentes Principales	19
4.1	<i>Motivación</i>	19
4.2	<i>Fundamentos teóricos de PCA</i>	20
4.3	<i>Aplicación de PCA para la representación de las caras: Eigenfaces</i>	20
4.4	<i>Proyección y reducción de la dimensionalidad</i>	22
5	Implementación del sistema	25
5.1	<i>Entorno de desarrollo</i>	25
5.1.1	Lenguaje de programación	25
5.1.2	Entorno de trabajo	25
5.1.3	Librerías y extensiones	25
5.2	<i>Etapas del sistema</i>	26
5.2.1	Etapa de entrenamiento	27
5.2.2	Etapa de test	27
5.3	<i>Detección</i>	28
5.3.1	Detección de caras	28
5.3.2	Detección de ojos	28
5.4	<i>Preprocesado</i>	29
5.4.1	Rotación	30
5.4.2	Escalado	30
5.4.3	Recorte	31
5.4.4	Ecuilibración del histograma	31
5.5	<i>Extracción de Características</i>	32
5.5.1	Cálculo de las <i>Eigenfaces</i>	32
5.5.2	Proyección sobre las <i>Eigenfaces</i>	32
5.6	<i>Comparación y decisión</i>	32
6	Experimentos realizados y resultados	33
6.1	<i>Bases de datos utilizadas</i>	33
6.1.1	ORL Face Database	33
6.1.2	Normalized Yale Face Database	34
6.1.3	Base de datos propia	34
6.2	<i>Evaluación de la detección</i>	36
6.3	<i>Evaluación del reconocimiento</i>	38
6.3.1	Variación del número de imágenes de entrenamiento	38
6.3.2	Variación del número de <i>eigenfaces</i>	40
6.4	<i>Representación de los resultados</i>	42
7	Conclusiones y líneas futuras	45
7.1	<i>Conclusiones</i>	45
7.2	<i>Líneas futuras</i>	46
	Referencias	47
	Glosario	51

ÍNDICE DE TABLAS

Tabla 6-1. Datos sobre la base de datos ORL	33
Tabla 6-2. Datos sobre la base de datos Yale	34
Tabla 6-3. Datos sobre la base de datos propia	34
Tabla 6-4. Resultados de la variación de imágenes de entrenamiento en la base de datos ORL	38
Tabla 6-5. Resultados de la variación de imágenes de entrenamiento en la base de datos YALE	39
Tabla 6-6. Resultados de la variación de imágenes de entrenamiento en la base de datos propia	39
Tabla 6-7. Resultados de la variación del número de <i>eigenfaces</i> en la base de datos ORL	40
Tabla 6-8. Resultados de la variación del número de <i>eigenfaces</i> en la base de datos YALE	41
Tabla 6-9. Resultados de la variación del número de <i>eigenfaces</i> en la base de datos propia	41

ÍNDICE DE FIGURAS

Figura 2-1. Rasgos faciales	4
Figura 2-2. Sistema de reconocimiento facial a la entrada al estadio Centenario, Uruguay	6
Figura 2-3. Sistema de reconocimiento facial en la aduana del aeropuerto JFK, NY	6
Figura 2-4. Software NeoFace de NEC para confirmar la identidad de sospechosos	6
Figura 2-5. Entrenamiento de Face Unlock en smartphone Samsung	7
Figura 3-1. Diagrama general de un sistema de reconocimiento facial	9
Figura 3-2. Características tipo <i>Haar</i>	10
Figura 3-3. Ejemplos de características <i>Haar</i> comunes de un detector de cara	10
Figura 3-4. Detección de la región de la cara	11
Figura 3-6. Detección de los ojos	11
Figura 3-5. Ejemplos de características <i>Haar</i> comunes de un detector de ojos	11
Figura 3-7. Preprocesamiento de una imagen	12
Figura 3-8. Taxonomía del reconocimiento facial	13
Figura 3-9. Eigenfaces de un conjunto de imágenes de la base Extended Yale Face Database B	14
Figura 3-10. Fisherfaces de un conjunto de imágenes de la base Extended Yale Face Database B	14
Figura 3-11. Vectores de distancias obtenidos a partir de puntos característicos	15
Figura 3-12. LBP sobre imágenes con diferente intensidad	15
Figura 3-13. Ejemplo de cara representada como grafo mediante EBGGM	16
Figura 3-14. Parametrización y extracción de bloques de imagen facial mediante HMM	16
Figura 3-15. Ejemplo del clasificador K-Nearest Neighbours	18
Figura 4-1. Ejemplo simplificado del espacio de las imágenes (3D) y región ocupada por las caras	20
Figura 4-2. Ejemplo de cara media y 7 primeras <i>eigenfaces</i>	22
Figura 5-1. Diagrama general del sistema implementado	26
Figura 5-2. Resultado de la detección de caras del sistema implementado	28
Figura 5-3. Resultado de la detección de caras y ojos del sistema implementado	29

Figura 5-4. Ejemplo de imagen de un rostro para el preprocesado	29
Figura 5-5. Resultado de la rotación	30
Figura 5-6. Resultado del escalado	30
Figura 5-7. Resultado del recorte	31
Figura 5-8. Resultado de la ecualización del histograma	31
Figura 6-1. Imágenes de tres sujetos de la base de datos ORL	33
Figura 6-2. Imágenes de cuatro sujetos de la base de datos Yale	34
Figura 6-3. Algunas de las imágenes utilizadas para cada individuo de la base de datos propia	35
Figura 6-4. Imágenes de individuos con los rostros despejados, aptas para la detección	36
Figura 6-5. Imágenes de individuos de perfil, no aptas para la detección	36
Figura 6-6. Imágenes de individuos con el rostro sin despejar, no aptas para la detección	37
Figura 6-7. Imágenes de individuos con condiciones de iluminación no aptas para la detección	37
Figura 6-8. Salida que muestra el sistema al finalizar el reconocimiento	42
Figura 6-9. Resultado del reconocimiento para una imagen con más de un individuo	42
Figura 6-10. Resultado del reconocimiento en caso de error	43

Notación

d_E	Distancia Euclídea
P, Q	Puntos del espacio n -dimensional
D_B	Distancia Bhattacharyya
BC	Coefficiente de Bhattacharyya
p, q	Distribuciones de probabilidad
\ln	Logaritmo neperiano
w	Píxeles de ancho de una imagen
h	Píxeles de alto de una imagen
\cdot	Multiplicación
N	Número de imágenes del conjunto de entrenamiento
X	Vector aleatorio de dimensión $w \cdot h$
E	Esperanza
μ	Media
σ^2	Varianza
σ_{ij}^2	Covarianza entre las componentes i y j
Σ_X	Matriz de covarianza de X
M_X	Matriz formada por las imágenes de entrenamiento con su media restada
M_X'	Traspuesta
λ	Autovalores
v	Autovectores
V	Matriz que contiene los autovectores de $M_X M_X'$

1 INTRODUCCIÓN

1.1 Motivación del trabajo

En los últimos años, la necesidad de mejorar la seguridad en diferentes contextos de la sociedad actual, dan lugar al desarrollo de nuevo hardware y software informático empleando técnicas de reconocimiento automático basadas en rasgos biométricos.

Los sistemas biométricos son sistemas automatizados que permiten la identificación y verificación de individuos de forma rápida y segura, puesto que solo se analizan determinados patrones biométricos que no pueden ser alterados, manipulados, falsificados o robados para recrear información personal o acceder a la información de otros.

Estos patrones son características tanto físicas como de comportamiento. Las huellas dactilares, la retina, el iris, los patrones faciales, de venas de la mano o la geometría de la palma de la mano, representan ejemplos de características físicas, mientras que en las características del comportamiento se incluye el reconocimiento de voz, firma, dinámica del tecleo, cadencia del paso y conducta gestual. Todos estos son rasgos inherentes a la persona, es por ello que estos sistemas suponen una ventaja frente a sistemas tradicionales que suelen requerir información adicional como documentos de identidad, tarjetas de acceso, etc. que puede ser perdida o robada provocando brechas de seguridad.

Son muchas las empresas que optan hoy en día por sistemas biométricos de identificación, para llevar un mejor control de asistencia de sus trabajadores. En general, toda aplicación que requiera seguridad, control de acceso, control de presencia e identificación o comprobación del usuario, puede ser beneficiaria por este tipo de sistemas.

En la actualidad, sistemas biométricos como los basados en huella dactilar o iris obtienen unas tasas de reconocimiento formidables siendo la elección preferida en muchos campos de aplicación. Uno de los inconvenientes de estas técnicas es el problema de la adquisición, ya que a menudo resultan demasiado intrusivas para algunas aplicaciones. El reconocimiento facial es una técnica muy estudiada y que, debido a su baja intrusividad y buenos resultados, la convierten en una gran candidata para ser aplicada en entornos donde otras técnicas no son viables. La cara es un rasgo biométrico con gran capacidad discriminativa y con el cual los humanos nos identificamos entre nosotros. Por todo ello, es un rasgo comúnmente aceptado y de fácil adquisición.

La motivación que me ha llevado a la selección de este trabajo es la curiosidad por entender el funcionamiento de los sistemas de reconocimiento facial, cada vez más presentes en nuestro día a día. No hay mejor manera para comprender el funcionamiento de un sistema, que llevar a cabo su implementación.

1.2 Objetivos y enfoque

El objetivo principal de este trabajo es el estudio, implementación y evaluación de un sistema de reconocimiento facial. No se crearán nuevos algoritmos ni nuevas formas de reconocimiento, sino que se implementará uno de los métodos ya existentes estudiados en el estado del arte, en este caso el método *Eigenfaces*, que es una de las técnicas más conocidas y utilizadas en la actualidad.

El usuario dispondrá de un sistema que le permitirá reconocer rostros tanto en imágenes como en videos en tiempo real. Para poder reconocer, se necesitará primero una fase previa de entrenamiento, donde el usuario entrenará con fotos de individuos el algoritmo para que sea capaz de reconocer a dichos sujetos en cualquier imagen o video.

1.3 Organización de la memoria

El presente documento se divide en siete capítulos:

- **Capítulo 1: Introducción.** Se trata del capítulo actual, en el cual se exponen las razones que han motivado a la realización del trabajo así como los objetivos perseguidos con el mismo.
- **Capítulo 2: Estado del arte.** En primer lugar se explican los fundamentos del reconocimiento facial añadiendo un trasfondo histórico exponiendo los hitos más relevantes al reconocimiento facial hasta el presente. Se compara además el reconocimiento facial con otros sistemas de reconocimiento biométrico mostrando las ventajas e inconvenientes del mismo. Por último y debido al enfoque de desarrollo e implementación del trabajo, se incluye un breve repaso de aplicaciones y sistemas comerciales más importantes en la actualidad.
- **Capítulo 3: Fases de un sistema de reconocimiento facial.** En este capítulo se exponen las etapas que todo sistema de reconocimiento facial sigue, detallando las técnicas más usadas hoy día en cada una de las fases.
- **Capítulo 4: Análisis de Componentes Principales.** Se explica todo el fundamento teórico de la técnica de reconocimiento empleada en el sistema del presente trabajo.
- **Capítulo 5: Implementación del sistema.** En este capítulo se detallan los pasos seguidos para llevar a cabo la implementación del sistema. La explicación se divide según las diferentes fases vistas en el capítulo 3. También se expone el entorno de trabajo utilizado para llevar a cabo dicha implementación.
- **Capítulo 6: Experimentos realizados y resultados.** Los experimentos que se llevan a cabo para evaluar diferentes características del funcionamiento del sistema se dividen en dos secciones, experimentos relacionados con la detección y experimentos relacionados con el reconocimiento, ambas sobre distintas bases de datos de imágenes.
- **Capítulo 7: Conclusiones y líneas futuras.** En este capítulo se resume el trabajo realizado en el proyecto extrayendo las conclusiones más importantes y se proponen líneas de trabajo futuras que hagan posible la continuación y mejora del trabajo aquí expuesto.

2 ESTADO DEL ARTE

2.1 Historia del reconocimiento facial

Una de las formas más comunes que desde siempre han tenido los humanos para identificarse entre ellos es mediante sus rostros. Al ser un rasgo muy característico y de fácil acceso para la vista humana, somos capaces de discernir entre diferentes personas sólo con la información del mismo. Desde los comienzos de la visión artificial, el reconocimiento facial ha sido estudiado debido a su importancia práctica e interés teórico de científicos cognitivos.

El reconocimiento facial automatizado es un concepto relativamente nuevo. El comienzo de las investigaciones en esta técnica se remonta a los años 60, cuando W. W. Bledsoe [1] y su equipo de investigación desarrollaron los primeros sistemas de reconocimiento. Estos sistemas eran semiautomáticos, ya que requerían un administrador para localizar rasgos característicos en las fotografías.

En 1988, L. Sirobich y M. Kirby [2] aplicaron análisis de componentes principales (PCA), una técnica estándar del álgebra lineal, al problema del reconocimiento facial. Esto fue considerado como un hito al mostrar que eran requeridos menos de 100 componentes para cifrar acertadamente la imagen de una cara convenientemente alineada y normalizada. Un año más tarde, T. Kohonen [3] definió esta técnica de reconocimiento basada en la caracterización de la cara por la extracción de los autovectores de la matriz de autocorrelación como *Eigenfaces*.

En 1991 Turk y Pentland [4] demostraron que el error residual de codificar las *Eigenfaces* se podía utilizar para detectar caras en las imágenes, un descubrimiento que permitió sistemas automatizados de reconocimiento facial en tiempo real. Si bien la aproximación era un tanto forzada, creó sin embargo un interés significativo en posteriores desarrollos de estos sistemas.

La tecnología capturó la atención del público a partir de la reacción de los medios a una prueba de implementación en el Super Bowl de la NFL en enero de 2001, la cual capturó imágenes de vigilancia y las comparó con una base de datos de fotos digitales. Esta demostración inició un muy requerido debate sobre cómo usar la tecnología para satisfacer necesidades nacionales, mientras se tomaban en consideración las preocupaciones sociales y de privacidad del público.

A raíz de estas investigaciones, el interés sobre las técnicas de reconocimiento facial aumentó sensiblemente y, desde entonces, numerosas técnicas han sido desarrolladas en esta área consiguiendo altos niveles de perfeccionamiento.

2.2 Fundamentos del reconocimiento facial

La cara humana nos proporciona gran cantidad de información sobre un sujeto, permitiéndonos discernir e identificar a simple vista diferentes individuos. Alberga un conjunto de rasgos (Figura 2-1) localizados en posiciones similares a lo largo de la población, por lo que un sistema de reconocimiento facial puede beneficiarse de esta característica. En concreto, como se verá en capítulos posteriores, los sistemas de reconocimiento facial que emplean técnicas basadas en modelos son los que aprovechan dichos rasgos. Por tanto, aclarar que la técnica empleada en este trabajo, PCA, al pertenecer al grupo de técnicas basadas en apariencia no hace uso de estas características.

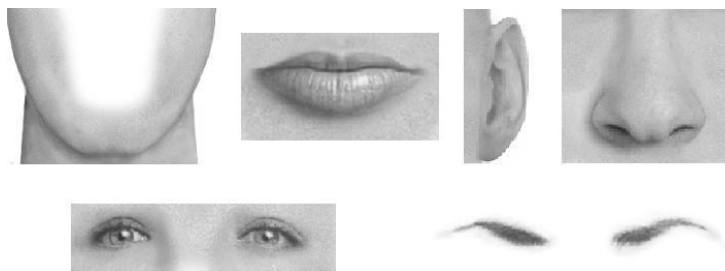


Figura 2-1. Rasgos faciales

A continuación se describen los rasgos más significativos que componen el rostro humano.

- **Orejas.** Habitualmente la variabilidad que presentan entre individuos es eminentemente geométrica, siendo el tamaño la característica que mejor las define. Las orejas al estar situadas en los laterales de la cara, pueden estar ocluidas por el pelo, generando variaciones no deseadas. Es por ello por lo que en muchos sistemas de reconocimiento la región de la cara que se extrae las excluye para evitar esta variabilidad.
- **Cejas.** Compuestas por vello situado en la parte superior de la cara justo encima de los ojos, ofrecen diferentes características a tomar en cuenta como son el grosor, la forma, el espesor y el color del vello. Su localización puede estar modificada por la expresión aunque por lo general no existe mucha variación del resto de características frente a diferentes gestos.
- **Ojos.** Los ojos, dada su complejidad, son quizá unos de los rasgos más discriminativos de la cara. Situados en la mitad superior de la cara, están compuestos por pestañas, párpados y el globo ocular que a su vez se diferencia en córnea, iris y pupila. Ofrecen gran variabilidad entre sujetos puesto que su geometría es diferente para cada uno y el iris, un rasgo biométrico por sí solo, dota a los ojos de gran información discriminativa. El inconveniente de los ojos es que en ocasiones los párpados ocluyen parcial o totalmente este rasgo, y además son bastante sensibles a cambios de expresión.
- **Nariz.** La nariz está situada aproximadamente en el centro de la cara. Su forma varía en gran medida entre los usuarios y la misma no suele ser afectada en los cambios de expresión. Los dos orificios nasales suelen ser un buen punto característico cuando se miden distancias.
- **Boca.** Por último, la boca situada en la parte inferior de la cara, es otro rasgo característico que facilita información del individuo. Como característica particular, debido a la gran flexibilidad y diversidad de movimientos que puede realizar este rasgo, es posible encontrar gran variabilidad en un mismo sujeto dependiendo de si está sonriendo, si tiene la boca abierta, está sacando la lengua, etc. Los labios son el componente que siempre está visible y que suelen definir el aspecto de la boca.

Adicionalmente a estos rasgos, también cabe destacar que la forma de la cara también es una característica discriminante, así como otras zonas de la misma como pueden ser los pómulos, la frente o la barbilla.

Existen, sin embargo, ciertas características de la cara que pueden introducir mayor variabilidad en el mismo individuo. En este sentido, el elemento más destacado es el pelo. El pelo puede contribuir a la oclusión de los rasgos (ya sea cabello o barba) y a cambiar el aspecto de una persona. Otros elementos artificiales comunes que suelen contribuir a la pérdida de fiabilidad en el sistema son las gorras, bufandas y gafas, siendo las gafas de sol las que mayor oclusión generan.

2.3 Ventajas e inconvenientes del reconocimiento facial

El reconocimiento facial presenta ciertas características favorables que la convierten en una técnica más viable para ser utilizada en ciertos ámbitos que otras técnicas biométricas [5]. Los aspectos más determinantes se definen a continuación.

- **Simetría.** La cara es un rasgo con una componente simétrica bastante elevada, algo de lo que las tareas de localización o extracción de distancias se pueden beneficiar.
- **Baja intrusividad.** Al ser la cara un rasgo humano muy visible, facilita la tarea de su obtención lo que conlleva que la misma sea poco intrusiva para los individuos. La cooperación que conlleva esta obtención es también mínima puesto que con que una cámara sea capaz de capturar la cara de una persona con una calidad aceptable es suficiente. Esto permite diseñar sistemas de reconocimiento sin que los individuos se percaten de que están siendo identificados.
- **Gran poder discriminante.** Aunque la cara no sea uno de los rasgos con mayor poder discriminativo, sí que es un rasgo con una cantidad suficiente de información y variabilidad para que tenga un alto poder discriminante. Una alta cantidad de sujetos en una base de datos no llega a deteriorar en gran medida la eficacia del sistema, obteniéndose también en estos casos buenas tasas de reconocimiento.
- **Disponibilidad extendida.** El ser humano ha utilizado como método de reconocimiento básico la cara de las personas en gran variedad de ámbitos. Esto hace que haya una gran disponibilidad de bases de datos y que este rasgo haya sido obtenido de la mayor parte de la población con anterioridad. Al haber sido este rasgo muy utilizado, se han llevado a cabo multitud de investigaciones referentes al reconocimiento facial que han resultado en diferentes técnicas y algoritmos de los cuales hoy en día nos podemos beneficiar.

Al igual que las ventajas aquí descritas, el reconocimiento facial, como cualquier otra técnica biométrica, presenta ciertos inconvenientes que deben ser mencionados.

- **Variabilidad.** Uno de los grandes inconvenientes que presenta esta técnica, es la plasticidad del rostro humano respecto a los gestos y a que es un rasgo que no presenta invariabilidad temporal [6], teniendo varios rostros de un sujeto a lo largo de los años diferentes a simple vista.
- **Entornos no controlados.** Cuando el entorno no está controlado, los cambios en la pose, oclusiones (gafas, gorro etc.) y baja calidad de las imágenes tomadas, hacen que el sistema pierda fiabilidad presentando inconvenientes a la hora de reconocer personas en un entorno real.
- **Situaciones particulares.** Por último, existen ciertas situaciones particulares en las que el reconocimiento facial no es viable, ya que debido al fácil acceso del rasgo, un impostor podría utilizar imágenes de rostros de otras personas sin mucha dificultad. En este sentido, en los últimos años se está prestando mucho interés estudiando y generando soluciones.

Cabe destacar que respecto a varios de los inconvenientes anteriormente descritos, año tras año se están desarrollando nuevas técnicas e investigando formas de solventarlos que hacen disminuir el impacto que tienen sobre los sistemas de reconocimiento facial. En el actual estado del arte se están consiguiendo tasas de reconocimiento razonablemente altas para imágenes tomadas en entornos no controlados [7].

2.4 Aplicaciones de los sistemas de reconocimiento facial

Existe una gran cantidad de aplicaciones prácticas que pueden beneficiarse del reconocimiento facial. A continuación se muestran algunas de las más conocidas.

1. **Control de acceso.** Una de las aplicaciones de los sistemas de reconocimiento facial más extendidas es el uso de los mismos en lugares donde se requiere la verificación de identidad. Un ejemplo es la verificación de la identidad de una persona en la aduana de un aeropuerto (Figura 2-2) o el control a la entrada de un estadio (Figura 2-3). La información contenida en el documento de identidad de la persona puede ser contrastada automáticamente con una imagen de su cara. Existen a su vez empresas con áreas de acceso restringido en las que los sistemas de reconocimiento facial también son útiles ya que en este caso existiría una base de datos previa con las fotos de las personas autorizadas a las cuales el sistema solo garantiza el acceso por medio de la obtención de la imagen de su cara en el lugar de acceso.

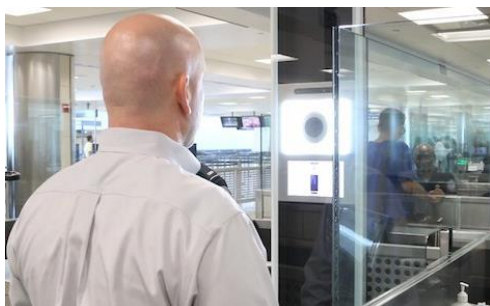


Figura 2-3. Sistema de reconocimiento facial en la aduana del aeropuerto JFK, NY



Figura 2-2. Sistema de reconocimiento facial a la entrada al estadio Centenario, Uruguay

2. **Sistemas de seguridad.** Estos sistemas son utilizados para añadir una capa más de seguridad al sistema de videovigilancia. Las cámaras son capaces de detectar las caras de los individuos que aparecen en la imagen y realizar tareas de reconocimiento en tiempo real contra una base de datos de sospechosos y alertar en caso de obtener alguna coincidencia (Figura 2-4). La utilización de estos sistemas está facilitada por la gran extensión de los sistemas de vigilancia a infinidad de áreas.



Figura 2-4. Software NeoFace de NEC para confirmar la identidad de sospechosos

- 3. Aplicaciones móviles y redes sociales.** Con la llegada de los smartphones al mercado, los teléfonos móviles son cada día capaces de realizar una mayor variedad de tareas debido al desarrollo de la tecnología. Algunos teléfonos con cámara integrada son capaces de beneficiarse de sistemas de reconocimiento facial para permitir el acceso al dispositivo por parte del dueño (Figura 2-5) o para, por ejemplo, detectar las caras en una foto y reconocer si se trata de gente guardada en la lista de contactos para utilizar esa información como clasificador. Muchos programas de fotografía y redes sociales como Facebook, también llevan integrados sistemas de reconocimiento facial para automatizar las tareas de etiquetar y compartir fotos.

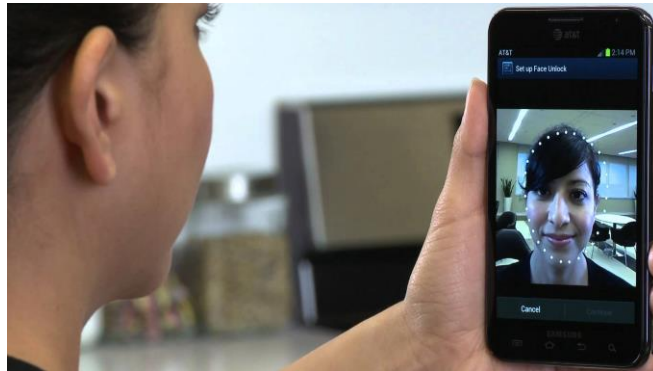


Figura 2-5. Entrenamiento de Face Unlock en smartphone Samsung

2.5 Sistemas comerciales en la actualidad

Hoy en día gran cantidad de sistemas de reconocimiento facial en tiempo real están en uso para muy diversas aplicaciones, ya sean comerciales, públicas, de uso general o privadas. Existen muchas compañías compitiendo por conseguir el mejor algoritmo de reconocimiento facial, cuyo objetivo es conseguir un nivel de precisión prácticamente igual al de los humanos.

En 2014, Facebook anunció que había conseguido obtener un 97,3% de porcentaje de acierto con su algoritmo DeepFace [8] probado en la base de datos de imágenes LFW (Labeled Face in the Wild). Ese resultado mejoraba en un 27% el mejor resultado hasta el momento. Google, por su parte, con su algoritmo FaceNet [9] obtenía un porcentaje superior con un 99,6% de aciertos.

En Enero de 2015, Facebook presenta un nuevo método de reconocimiento llamado PIPER [10] capaz de reconocer a personas en poses inusuales, incluso si su cara no es visible. Este algoritmo ya no es solamente capaz de fijarse en la cara, sino que además analiza aspectos como el peinado, la ropa o los complementos.

Así pues, el reconocimiento facial, entendido como método para identificar la identidad de una persona, evoluciona hacia nuevos niveles. Ahora existen algoritmos capaces de identificar a personas por características muy diferentes y no solo por su rostro.

3 FASES DE UN SISTEMA DE RECONOCIMIENTO FACIAL

A pesar de la gran variedad de sistemas existentes, todo sistema de reconocimiento facial puede dividirse en cuatro fases principales (Figura 3-1). La primera fase comprende la detección y localización de la cara en la imagen para su posterior preprocesado. Esta fase de preprocesado se lleva a cabo para preparar el rostro mediante la alineación, escalado, recorte y ecualización de la imagen. Una vez que la imagen del rostro está preparada, se realiza la tercera fase de extracción de características para obtener la información relevante de la imagen. Por último, se realiza la comparación donde se aplican los algoritmos de reconocimiento y se extrae una decisión respecto a la base de datos y la imagen de entrada.

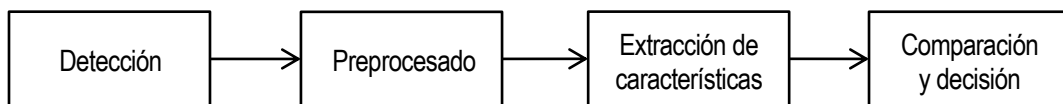


Figura 3-1. Diagrama general de un sistema de reconocimiento facial

Estas cuatro fases se aplican a cada *imagen de test* que queramos reconocer. No obstante, existe una **etapa previa de entrenamiento** para preparar el sistema que solo se ejecuta una vez y en primer lugar. En esta etapa, las tres primeras fases se aplican a las *imágenes del conjunto de entrenamiento* de la base de datos para extraer las características más relevantes de dichas imágenes. Una vez entrenado el sistema, en la **etapa de test** ya se podrán introducir imágenes para ser reconocidas.

Veremos ahora con más detalle en qué consisten cada una de estas cuatro fases y cuáles son las técnicas más utilizadas hoy en día, sin hacer distinción entre si se trata de la etapa de entrenamiento o de test, pues el proceso a aplicar a la imagen es el mismo indistintamente. En el capítulo 5, donde se hablará del desarrollo del sistema implementado en este trabajo, será donde se expliquen en qué consisten cada una de estas dos etapas.

A la hora de implementar el sistema se elegirá una de las técnicas nombradas a continuación para cada fase. Sin embargo el sistema es completamente modular, por lo que se pueden cambiar fácilmente dichas técnicas por otras en un futuro para comparar resultados y buscar mejoras.

3.1 Detección

La fase de detección consiste en encontrar áreas de la imagen que contengan una cara para aislarlas del resto. Este es uno de los objetivos del trabajo, y una de las fases más importante de la aplicación ya que una mala detección conllevaría un error en el resto de las fases.

En la actualidad, existen distintos algoritmos que pueden ser implementados para una detección facial, muchos de ellos un tanto complejos o específicos según el área donde se desean emplear. Estos pueden llegar a tratarse de redes neuronales, grafos, entre otros. Sin embargo, en este trabajo se va a comprender uno de los métodos más populares y eficientes que permiten detectar objetos en tiempo real.

3.1.1 Algoritmo de Viola-Jones

El algoritmo de Viola-Jones [11] es un método de detección de objetos que destaca por su bajo coste computacional, lo que permite que sea empleado en tiempo real. Su desarrollo fue motivado por el problema de la detección de caras, donde sigue siendo ampliamente utilizado, pero puede aplicarse a otras clases de objetos que, como las caras, estén caracterizados por patrones típicos de iluminación.

El algoritmo se basa en una serie de clasificadores débiles denominados *Haar-like features* que se pueden calcular eficientemente a partir de una imagen integral. Estos clasificadores, que por sí mismos tienen una probabilidad de acertar solo ligeramente superior a la del azar, se agrupan en una cascada empleando un algoritmo de aprendizaje basado en AdaBoost para conseguir un alto rendimiento en la detección así como una alta capacidad discriminativa en las primeras etapas.

3.1.1.1 Haar-like features

Las *Haar-like features* son los elementos básicos con los que se realiza la detección. Reciben este nombre por similitud a los *wavelets de Haar* [12]. Estos clasificadores son características muy simples que se buscan en las imágenes y que consisten en la diferencia de intensidades luminosas entre regiones rectangulares adyacentes. Las características quedan por tanto definidas por unos rectángulos y su posición relativa a la ventana de búsqueda y adquieren un valor numérico resultado de la comparación que evalúan.

En el trabajo presentado por Viola-Jones existen tres tipos de características, representadas en la Figura 3-2. Este algoritmo considera regiones rectangulares en una ventana de detección, suma las intensidades de los píxeles en cada región y calcula la diferencia entre estas sumas. En la Figura 3-3 pueden verse algunos ejemplos de características *Haar* comunes de un detector de caras.

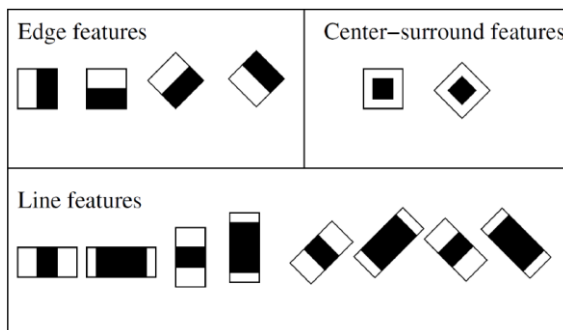


Figura 3-3. Características tipo *Haar*



Figura 3-2. Ejemplos de características *Haar* comunes de un detector de cara

3.1.1.2 Imagen Integral

La suma de los píxeles de un rectángulo puede ser calculada de manera muy eficiente empleando una representación intermedia denominada imagen integral. La imagen integral en un punto contiene la suma de todos los píxeles que están arriba y hacia la izquierda de ese punto en la imagen original. La imagen integral total se puede calcular fácilmente en un solo barrido de la imagen.

3.1.1.3 Proceso de aprendizaje

Es necesario realizar un proceso de entrenamiento supervisado para crear el clasificador mediante una gran cantidad de imágenes positivas (caras) e imágenes negativas (imágenes sin caras). Este proceso se realiza mediante un algoritmo basado en AdaBoost [13], un algoritmo adaptativo de machine learning cuyo nombre es una abreviatura de *adaptive boosting*.

3.1.1.4 Cascada de clasificadores

En vez de construir un único clasificador mediante el proceso anterior, se construyen clasificadores más pequeños y eficientes. Estos clasificadores más simples se utilizan para rechazar la mayoría de las ventanas de búsqueda y solo en aquellas en las que hay mayores probabilidades de encontrar caras se llama a clasificadores más complejos. Se obtiene así una cascada de clasificadores, cada uno de los cuales es entrenado con AdaBoost.

3.1.2 Detección de caras

Empleando el algoritmo de Viola-Jones explicado en el apartado anterior y mediante un entrenamiento correcto, se hace posible un detector de caras robusto basado en un clasificador en cascada como se muestra en la Figura 3-4.

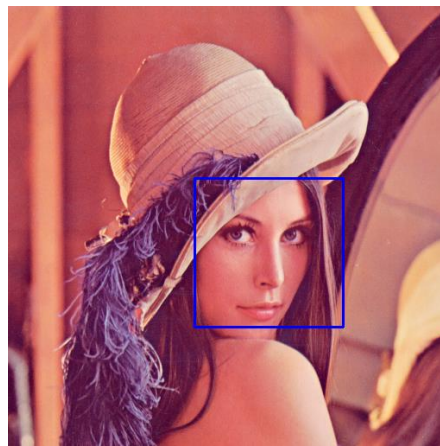


Figura 3-4. Detección de la región de la cara

3.1.3 Detección de ojos

Para realizar una correcta alineación de la imagen en el preprocesado, es necesario determinar las coordenadas de los ojos en la misma. Para detectar los ojos existen varios métodos. El método más directo, y también el más usado, es utilizar los clasificadores *Haar* usados en la detección de caras pero esta vez entrenados con imágenes de ojos [14] (Figura 3-5 y 3-6).



Figura 3-5. Ejemplos de características *Haar* comunes de un detector de ojos

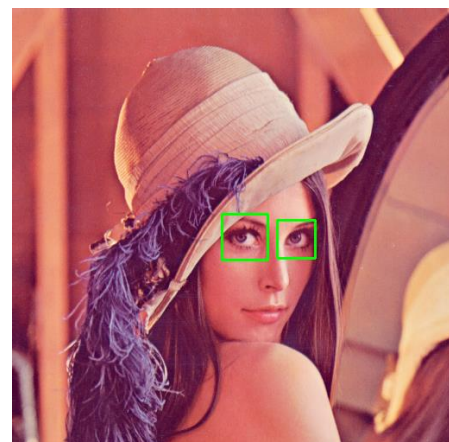


Figura 3-6. Detección de los ojos

El segundo método consiste en acotar la región en la que sabemos que van a estar los ojos por geometría facial y obtener la gráfica que representa la suma por columnas de los valores de la imagen binarizada. De esta forma se pueden buscar los picos de la imagen que corresponderán a los ojos.

Por último, con métodos basados en momentos invariantes [15] también se obtienen buenos resultados. Para poder detectar los ojos de esta forma, la imagen se binariza y se etiquetan las regiones inconexas de la misma. Posteriormente, se utilizan los momentos invariantes para conseguir obtener las dos regiones más parecidas.

3.2 Preprocesado

La etapa de preprocesado [16] se lleva a cabo a partir de la información obtenida en la detección. Esta etapa realiza una serie de transformaciones geométricas sobre la imagen dejándola preparada para la correcta extracción de características. En el preprocesado se utilizan cuatro fases para normalizar y alinear la imagen tal y como muestra la Figura 3-7.

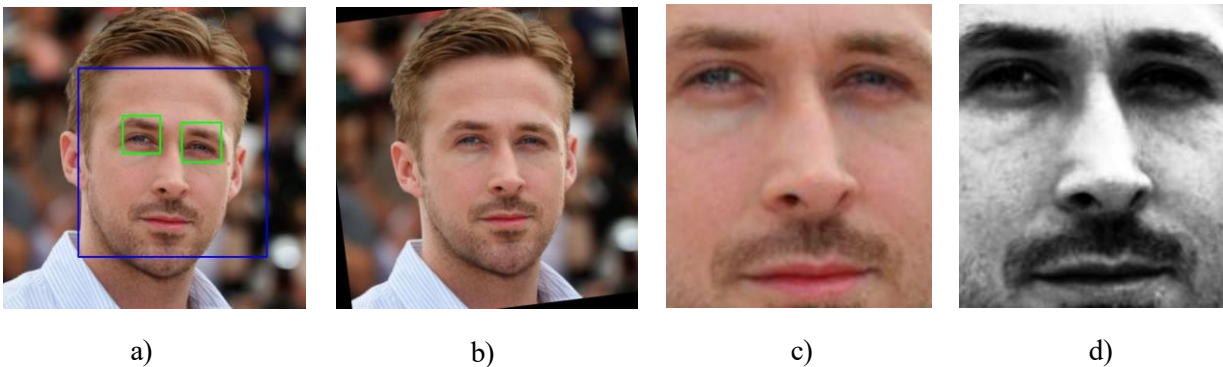


Figura 3-7. Preprocesamiento de una imagen: a) Imagen de entrada. b) Rotación a partir de las coordenadas de los ojos. c) Recorte y escalado. d) Ecuilización del histograma.

- **Rotación.** Una de las utilidades de calcular las coordenadas de los ojos, radica en poder determinar el ángulo de giro de una cara en una imagen y compensarlo. Al tener caras sin giro, el proceso de reconocimiento dará mejores resultados.
- **Escalado.** Para conseguir que todos los rostros de las imágenes tengan las mismas proporciones, se utiliza la distancia entre los centros de los ojos para conseguir un ratio por el cual la imagen debe ser aumentada o reducida. Esto es necesario puesto que muchas técnicas de reconocimiento requieren que todos los datos de entrada tengan el mismo tamaño (en nuestro caso la matriz de píxeles).
- **Recorte.** Una vez la imagen ha sido rotada y escalada, se procede al recorte de la misma para obtener sólo la región de interés. Se le da a todas las imágenes las mismas dimensiones con el fin de que todas ellas tengan el mismo tamaño para que sea posible la comparación entre ellas.
- **Ecuilización del histograma.** Las imágenes pueden presentar variabilidad en la luminosidad y en el contraste lo que produce que imágenes similares sean muy diferentes respecto al valor de intensidad de sus píxeles. Mediante la ecuilización de su histograma, se pretende que las imágenes que tienen la mayor parte de sus valores de intensidad concentrados en una zona reducida del histograma, pasen a extenderse por todo el rango de valores del histograma. Esto resulta en imágenes con mayor contraste y con menor variabilidad lumínica entre ellas.

3.3 Extracción de características

La extracción de características se emplea para obtener la información que resulta relevante de cara a realizar una comparación. Durante las últimas décadas se han desarrollado un gran número de algoritmos para llevar a cabo dicha extracción en el ámbito del reconocimiento facial. Como se puede ver en la Figura 3-8, las técnicas dependientes de pose se dividen en dos grandes grupos [17]. En primer lugar, las técnicas basadas en apariencia, que analizan la textura de la imagen a partir de la cual se aplicarán diferentes técnicas estadísticas y se extraerá la información. En segundo lugar, las técnicas basadas en modelos, que extraen las características tanto de la forma del rostro como de la textura. En este trabajo no se estudiarán las técnicas independientes de pose, puesto que la mayoría de ellas requieren equipamiento especializado para la captura de imágenes o capturas muy específicas controladas.

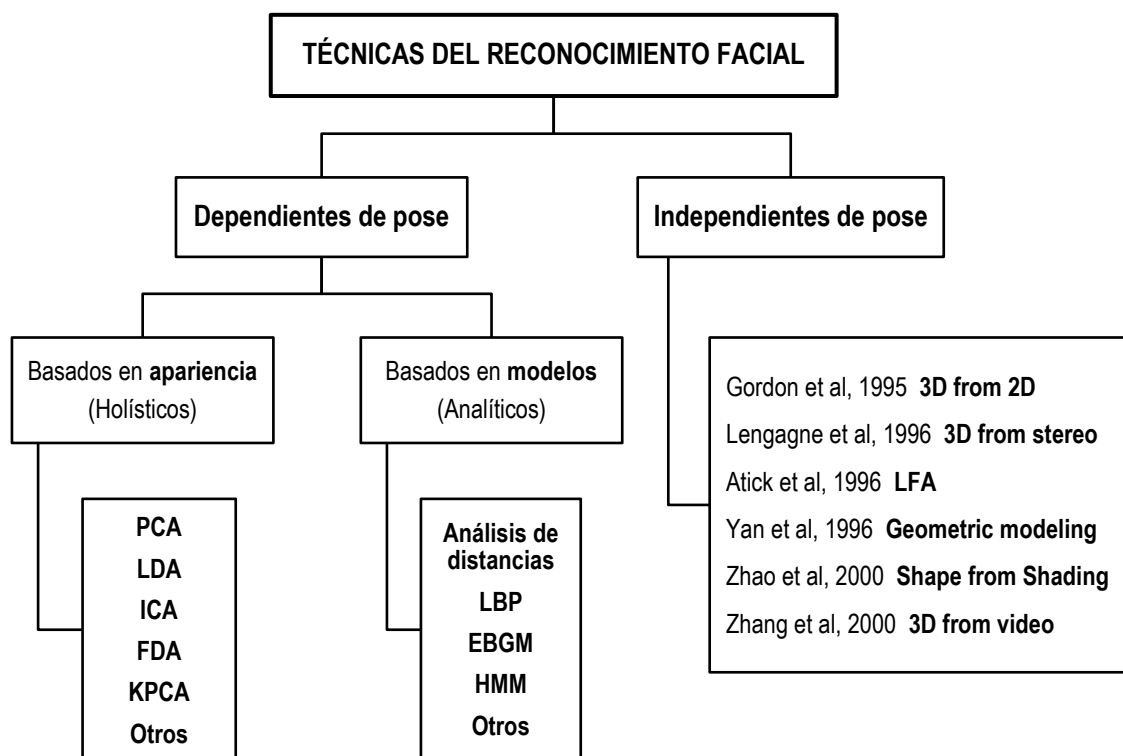


Figura 3-8. Taxonomía del reconocimiento facial

3.3.1 Técnicas basadas en apariencia

Estas técnicas basadas en apariencia transforman el problema de reconocimiento facial en un problema de análisis de espacio donde se pueden aplicar diferentes técnicas estadísticas. De este tipo de técnicas destaca su aplicabilidad en imágenes de baja resolución o mala calidad, su rapidez y ejecución (se pueden implantar en sistemas en tiempo real) o su baja complejidad. Sin embargo también tienen varios inconvenientes. Uno de los inconvenientes es que para conseguir buenos resultados se requiere un conjunto de muestras considerable para la fase de entrenamiento. También aspectos como los cambios en la iluminación, la pose o la expresión de la cara tienen un gran impacto en los resultados finales. Dependiendo del método empleado, estos inconvenientes tendrán un impacto mayor o menor.

3.3.1.1 Principal Component Analysis

El método *Eigenfaces* [4] es uno de los más populares, construido sobre técnicas de *Principal Component Analysis* (PCA) [18]. Dicho método transforma la imagen a un subespacio (ejemplo en Figura 3-9) por el cual es posible obtener vectores de características de menor dimensionalidad sin una pérdida de información discriminativa importante.

Al ser la técnica usada en el presente trabajo, se describe detalladamente su funcionamiento en el capítulo 4.



Figura 3-9. Eigenfaces de un conjunto de imágenes de la base Extended Yale Face Database B

3.3.1.2 Linear Discriminant Analysis

Derivada de PCA, se implementó una técnica que aúna PCA con *Linear Discriminant Analysis* (LDA) denominada *Fisherfaces* [19], con la que se consigue mayor robustez frente a cambios de iluminación pero resulta computacionalmente más costosa. Este método, junto con el anterior *Eigenfaces*, son los dos más usados en la actualidad en reconocimiento facial.



Figura 3-10. Fisherfaces de un conjunto de imágenes de la base Extended Yale Face Database B

3.3.1.3 Frequency Domain Analysis

Por último, las técnicas de análisis en el dominio de la frecuencia [20] ofrecen una representación de la imagen en función de las componentes de baja frecuencia que presentan alta energía. Tales técnicas como DFT, DCT o DWT son independientes de los datos por lo que no requieren de un entrenamiento. Adicionalmente, existen algoritmos optimizados que facilitan la implementación y reducen el coste computacional.

3.3.2 Técnicas basadas en modelos

Las técnicas basadas en modelos, tratan de obtener características biométricas de las imágenes para realizar el reconocimiento. Se tienen en cuenta aspectos como la distancia entre los ojos, el grosor de la nariz, el tamaño de la boca, etc. Estos sistemas requieren un conocimiento previo de las imágenes, además de ser más lentos y complejos que los sistemas basados en apariencia. Sin embargo, son más robustos frente a cambio de orientación o expresión de la cara y se ven menos afectados por cambios en la iluminación o las sombras.

3.3.2.1 Análisis de distancias a puntos característicos

Uno de los primeros sistemas de reconocimiento se basa en la técnica a partir de puntos geométricos de la cara [21]. A partir de la detección de diferentes puntos característicos (por ejemplo, como se muestra en la Figura 3-11) se crean vectores que contienen datos de distancias entre los mismos. Cuantos más puntos característicos son detectados, mayor número de distancias podrán ser calculadas obteniendo así mejores resultados en el reconocimiento.

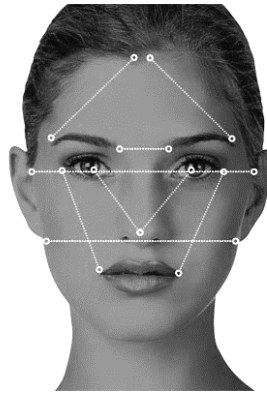


Figura 3-11. Vectores de distancias obtenidos a partir de puntos característicos

3.3.2.2 Local Binary Patterns

El algoritmo LBP [22] [23] es conocido como un buen descriptor de texturas a nivel local, utilizado en muchas aplicaciones de tratamiento de imágenes y reconocimiento de patrones. LBP etiqueta cada píxel de la imagen de acuerdo a los valores de sus píxeles vecinos. Para esto, se define un grado de vecindad y se les da el valor de 1 o 0 a estos píxeles según su nivel de intensidad sea mayor o menor que el valor del píxel central. A continuación, se recorren los vecinos y se genera una etiqueta binaria para el píxel central. Este proceso se repite sucesivamente para todos los píxeles de la imagen.

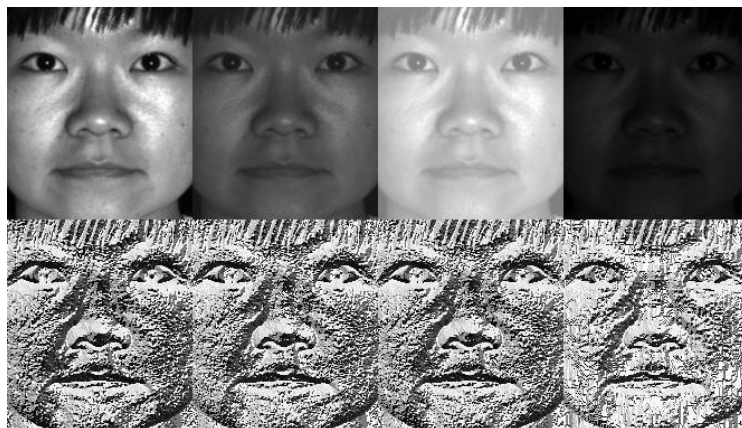


Figura 3-12. LBP sobre imágenes con diferente intensidad

En un enfoque para el reconocimiento facial, se trabaja con la imagen del rostro dividida en regiones (sub-imágenes). Con el operador LBP se codifica cada píxel de la sub-imagen y se recogen en un histograma regional. Posteriormente, se concatenan todos los histogramas regionales en un solo histograma global, para obtener una representación de la cara. Este proceso tiene la gran ventaja de ser sencillo de implementar y rápido de aplicar en una imagen, además, ofrece una gran robustez frente a cambios de iluminación como se muestra en Figura 3-12.

3.3.2.3 Elastic Bunch Graph Matching

EBGM [24] es una técnica que fue diseñada para realizar tareas de reconocimiento facial utilizando solamente algunos puntos de interés y no la cara en su totalidad, aprovechando la estructura topológica similar que estas presentan. La implementación de este algoritmo requiere el uso de las *wavelets de Gabor*, que son filtros paso banda que permiten alcanzar la resolución conjunta de información máxima en los espacios bidimensionales espacial y frecuencial. De esta manera, el uso de estas funciones bidimensionales proporciona una descripción de la información en frecuencia, en una región específica de la imagen.

La primera parte del algoritmo trata la extracción de las características locales. Para ello se define una estructura de grafo sobre la cara (Figura 3-13), cuyos nodos son puntos de interés que se puedan localizar fácilmente y que posean la misma estructura en todos los rostros, además, estos puntos deben mantener la simetría del rostro. Posteriormente, cada nodo del grafo es caracterizado utilizando un banco de filtros de Gabor de diferentes frecuencias y orientaciones. En cada nodo, se calcula la respuesta de todos los filtros, a lo que se le da el nombre de *Jet*. Finalmente para una imagen nueva se busca en la base de datos el conjunto de *jets* que sean más similares.

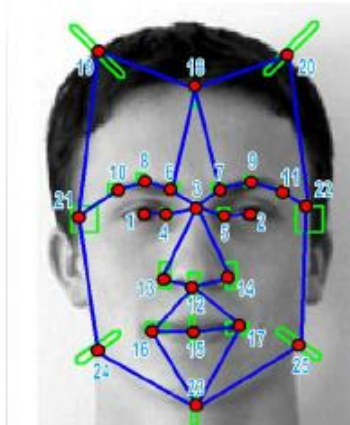


Figura 3-13. Ejemplo de cara representada como grafo mediante EBGM

3.3.2.4 Hidden Markov Models

Los HMMs [25] también han sido usados con éxito para el reconocimiento facial. Estas técnicas presentan robustez frente a cambios de iluminación, expresión y orientación, otorgando así una ventaja frente a los métodos holísticos. Las técnicas basadas en HMM utilizan regiones horizontales de píxeles (Figura 3-14) que albergan a la frente, ojos, nariz, boca y barbilla sin obtener la posición exacta de cada rasgo. Cada una de estas regiones es asignada a un estado del HMM para el reconocimiento.

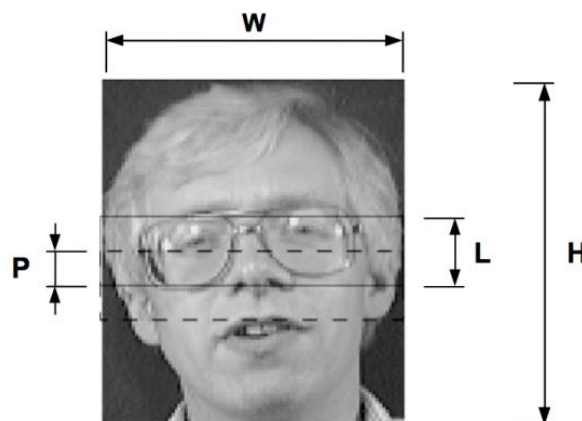


Figura 3-14. Parametrización y extracción de bloques de imagen facial mediante HMM

3.4 Comparación y decisión

En la última fase, se compara la información extraída del rostro que se quiere reconocer con los datos que hay almacenados en la base de datos. Para obtener una puntuación por la cual tomar una decisión, existen varios métodos de comparación. En este apartado se describen algunos de los más utilizados en la actualidad. Los métodos se pueden dividir en medidas de similitud y clasificadores [26].

3.4.1 Medidas de similitud o distancias

3.4.1.1 Distancia Euclídea

La distancia euclídea es una de las medidas más básicas para calcular distancias. Esta distancia se define como la distancia directa entre dos puntos en un plano. El ejemplo más claro es la distancia entre dos puntos en un plano de dos dimensiones de coordenadas x e y . Si tuviéramos dos puntos P_1 y P_2 con coordenadas (x_1, y_1) y (x_2, y_2) respectivamente, el cálculo de la distancia euclídea entre los mismos sería

$$d_E(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.1)$$

En general, la distancia euclídea entre dos puntos $P = (p_1, p_2, \dots, p_n)$ y $Q = (q_1, q_2, \dots, q_n)$ en el espacio euclídeo n -dimensional vendría definida como

$$d_E(P, Q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (3.2)$$

3.4.1.2 Distancia Bhattacharyya

La distancia Bhattacharyya [27] es una medida estadística que cuantifica la similitud entre dos distribuciones de probabilidad discretas o continuas. Está estrechamente relacionada con el coeficiente de Bhattacharyya, que es una medida de la superposición entre dos muestras estadísticas o poblaciones. Esta distancia es particularmente conocida por su bajo coste computacional y baja sensibilidad al ruido. Para las distribuciones de probabilidad p y q definidas sobre el mismo dominio \mathbf{X} , la distancia Bhattacharyya es definida como

$$D_B(p, q) = -\ln(BC(p, q)) \quad (3.3)$$

donde BC es el coeficiente de Bhattacharyya, definido como (3.4) para distribuciones de probabilidad discretas y como (3.5) para distribuciones de probabilidad continuas. En ambos casos, $0 \leq BC \leq 1$ y $0 \leq D_B \leq \infty$.

$$BC(p, q) = \sum_{x \in \mathbf{X}} \sqrt{p(x)q(x)} \quad (3.4) \quad BC(p, q) = \int \sqrt{p(x)q(x)} dx \quad (3.5)$$

En su formulación más simple, la distancia Bhattacharyya entre dos clases que siguen una distribución normal puede calcularse a partir de la media (μ) y varianza (σ^2) de las mismas como

$$D_B(p, q) = \frac{1}{4} \ln \left(\frac{1}{4} \left(\frac{\sigma_p^2}{\sigma_q^2} + \frac{\sigma_q^2}{\sigma_p^2} + 2 \right) \right) + \frac{1}{4} \left(\frac{(\mu_p - \mu_q)^2}{\sigma_p^2 + \sigma_q^2} \right) \quad (3.6)$$

3.4.2 Clasificadores

3.4.2.1 K-Nearest Neighbours

El método k-NN [28] es una técnica no paramétrica de clasificación de objetos basado en las muestras de entrenamiento más cercanas del espacio de características. El algoritmo se basa en encontrar los k vecinos más cercanos al objeto para, en función de la cantidad de los mismos, clasificarlo en el conjunto que tenga un mayor número de muestras cercanas. En el ejemplo de la Figura 3-15 podemos ver como para $k=3$ el objeto se clasificaría como triángulo y para $k=5$ como cuadrado.

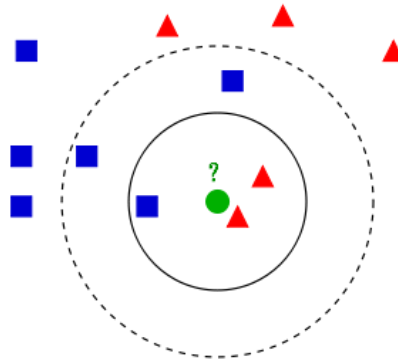


Figura 3-15. Ejemplo del clasificador K-Nearest Neighbours

3.4.2.2 Support Vector Machines

Los SVMs [29] son modelos de aprendizaje usados para clasificación y regresión. El objetivo de este método es el de representar en un espacio una serie de clases y tratar de encontrar un hiperplano que divida ambas clases de una forma óptima, es decir, que la distancia entre ambas sea máxima.

El cálculo de este hiperplano depende en gran medida de la distribución de las clases, es por ello que, dependiendo de la técnica utilizada para extraer características, los SVMs funcionarán mejor, o en ciertos casos no serán viables.

4 ANÁLISIS DE COMPONENTES PRINCIPALES

Tras haber llevado a cabo la fase de búsqueda de información, es necesario decidir qué técnica va a desarrollarse. Se decide implementar una de las técnicas de reconocimiento más usadas, como es el caso de PCA, que aplicada al ámbito del reconocimiento facial, es también conocida como *Eigenfaces*. Una de las razones es que este tipo de sistemas tiene probablemente el mejor compromiso entre complejidad, rapidez de ejecución y resultados. Como método de comparación se decide utilizar la distancia euclídea, ya que a pesar de ser una de las medidas más básicas, se verá como ofrece muy buenos resultados.

El primer paso para comprender los sistemas de reconocimiento basados en *Eigenfaces* es conocer la filosofía que se oculta detrás de PCA, que se utiliza para generarlas y que permite un mejor entendimiento de la representación de las imágenes de caras mediante *Eigenfaces*.

4.1 Motivación

Consideremos el conjunto de todas las imágenes posibles cuyas dimensiones son w píxeles de ancho por h píxeles de alto. El número de componentes de estas imágenes es $w \cdot h$. Si concatenamos las filas de estas imágenes podemos considerarlas como vectores de dimensión $w \cdot h$. Utilizando esta representación vectorial, el conjunto de imágenes forman un espacio vectorial de dimensión $w \cdot h$ que se puede generar a partir de la base canónica $(1,0,0, \dots, 0)$, $(0,1,0,0, \dots, 0)$, \dots , $(0,0, \dots, 0,1)$.

Consideremos ahora el conjunto de las imágenes de caras. En relación al resto de todas las posibles imágenes son todas relativamente semejantes: todas tienen los mismos elementos, situados de forma semejante, con textura parecida, etc. Además, los píxeles están altamente correlados con sus vecinos. Es razonable, por lo tanto, suponer que el conjunto de las imágenes de caras ocupa sólo una pequeña región de este espacio. Una consecuencia de esto es que la representación de las imágenes de caras utilizando la base del espacio de las imágenes es claramente ineficiente.

Además, debido a que todas las imágenes de caras son relativamente semejantes, podemos suponer que las imágenes de caras están en una zona reducida del espacio de las imágenes (llamémoslo el espacio de las caras). En ese caso, podríamos intentar hallar una base de vectores cuyo número sería mucho menor que el de una base del espacio de las imágenes. En la Figura 4-1 se muestra un ejemplo simplificado en el que el espacio de las imágenes es de dimensión tres (imágenes de tres píxeles).

Finalmente, llegamos a la conclusión de que debería ser posible generar las imágenes de caras (o al menos una muy buena aproximación) con muchos menos vectores que los necesarios para representar el espacio de las imágenes completo. PCA permite hallar esta nueva base de vectores con un máximo de eficiencia.

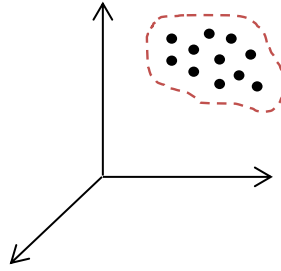


Figura 4-1. Ejemplo simplificado del espacio de las imágenes (3D) y región ocupada por las caras

4.2 Fundamentos teóricos de PCA

Como ya se ha dicho, las componentes (los píxeles) de las imágenes de caras están altamente correladas entre sí. PCA se basa precisamente en las propiedades estadísticas de las imágenes y es un método óptimo para reducir el número de dimensiones necesarias para representar un conjunto de vectores (en nuestro caso, imágenes de caras).

Consideremos un vector aleatorio \mathbf{X} de dimensión n . Su correspondiente media $\boldsymbol{\mu}$ viene definida como

$$\boldsymbol{\mu} = E\{\mathbf{X}\} \quad (4.1)$$

La matriz de covarianza $\boldsymbol{\Sigma}_X$ se define como se muestra en (4.2), donde σ_{ij}^2 representa la covariancia entre la componente número i y el número j del vector \mathbf{X} . El análisis de componentes principales consiste en hallar los autovectores de esta matriz y expresar \mathbf{X} en función de los mismos. De esta manera, el vector aleatorio \mathbf{X} se puede representar sin error mediante una combinación lineal de vectores de la forma (4.3) donde los vectores A_i son los autovectores de la matriz de covariancia. La cualidad más importante de esta representación es que si queremos representar \mathbf{X} con sólo m (menor que n) componentes, la mejor elección posible en términos del error que se comete son los m autovectores de la matriz de covariancia con mayores autovalores asociados.

$$\boldsymbol{\Sigma}_X = E\{(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})'\} = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \cdots & \sigma_{1n}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \cdots & \sigma_{2n}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1}^2 & \sigma_{n2}^2 & \cdots & \sigma_{nn}^2 \end{bmatrix} \quad (4.2)$$

$$\mathbf{X} = \sum_{i=1}^n y_i \mathbf{A}_i \quad (4.3)$$

4.3 Aplicación de PCA para la representación de las caras: Eigenfaces

Como se acaba de ver, la técnica PCA se reduce básicamente a encontrar los autovectores de la matriz de covariancia $\boldsymbol{\Sigma}_X$ del vector al que aplicamos dicha técnica. Apliquemos toda esta teoría a nuestro problema particular de reconocimiento de caras: nuestro objetivo es caracterizar un conjunto de N imágenes de caras (que serán las del conjunto de entrenamiento) en un espacio de menor dimensionalidad.

Tal y como se ha expuesto anteriormente, consideremos las imágenes de caras de este conjunto (de anchura w y altura h) como realizaciones del vector aleatorio \mathbf{X} de dimensión $w \cdot h$, con su correspondiente vector media $\boldsymbol{\mu}$. A este conjunto de imágenes de entrenamiento, deseamos aplicarle PCA.

La media del conjunto no la podemos obtener al no conocer la función de densidad de probabilidad, pero la podemos estimar a partir las imágenes que tenemos (4.4). La matriz de covarianza tampoco la conocemos pero también se puede estimar como se muestra en (4.5).

$$\boldsymbol{\mu} \approx \frac{1}{N} \sum_{i=1}^N \mathbf{X}_i \quad (4.4)$$

$$\boldsymbol{\Sigma}_X \approx \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})' \quad (4.5)$$

Podemos expresar la ecuación (4.5) en forma de producto de matrices (4.6), donde \mathbf{M}_X será la matriz formada por las imágenes de entrenamiento puestas en columnas con su media restada.

$$\boldsymbol{\Sigma}_X \approx \frac{1}{N} \mathbf{M}_X \mathbf{M}_X' \quad (4.6)$$

Llegados a este punto, en teoría sólo queda el cálculo de los autovectores de la matriz de covarianza, para lo que existen varios algoritmos automáticos fácilmente implementables. Pero la implementación de estos algoritmos plantea un problema: la cantidad de memoria necesaria para almacenar la matriz es demasiado grande. Si los vectores cara de la base de entrenamiento son de dimensión $w \cdot h$, las dimensiones de la matriz de covarianza estimada $\boldsymbol{\Sigma}_X$ serán de $w \cdot h$ de ancho y $w \cdot h$ de alto. Por ejemplo, si utilizamos imágenes de entrenamiento con unas dimensiones de $w = 128$ y $h = 128$, el número de elementos de la matriz de covarianza sería de 128^4 , o sea más de 268 millones, lo que está fuera del alcance de las memorias de los ordenadores actuales.

Una de las soluciones más recurrentes es aplicar una descomposición en valores simples (SVD, *Singular Value Decomposition*) [30] a la matriz \mathbf{M}_X , con el que se consiguen calcular los autovectores de la matriz de covarianza de forma indirecta. El método consiste en descomponer dicha matriz en un producto de tres matrices (4.7). La propiedad que nos es útil de la SVD es que la matriz \mathbf{V} contendrá los autovectores de la matriz $\mathbf{M}_X \mathbf{M}_X'$. Si recordamos la estimación de la matriz $\boldsymbol{\Sigma}_X$ (4.6) vemos que esta no era exactamente $\mathbf{M}_X \mathbf{M}_X'$, sino que además se dividía este producto por N . Pero esto no importa, ya que los autovectores de una matriz y los de la misma matriz dividida por un escalar son iguales.

$$\mathbf{M}_X = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V} \quad (4.7)$$

Cuando se trabaja con imágenes, este método supone un gran coste computacional debido a trabajar con una matriz \mathbf{M}_X de dimensiones muy grandes. Es por ello que J. E Solem propone en [31] un método compacto para calcular los autovectores de la matriz de covarianza en los casos en los que la dimensión de las imágenes $w \cdot h$ sea mayor que el número de imágenes de entrenamiento N , lo que siempre será nuestro caso. Este método consiste en lo siguiente:

1. Expresar \mathbf{M}_X de forma que las imágenes de entrenamiento estén puestas en filas y restarle su media.
2. Calcular su matriz de covarianza con algún algoritmo automático. Al expresar \mathbf{M}_X de esta forma, ahora su dimensionalidad ($N \cdot N$) sí será manejable por un ordenador.
3. Hallar los autovalores ($\boldsymbol{\lambda}$) y autovectores (\mathbf{v}) de la matriz de covarianza.
4. Por último, formar la matriz \mathbf{V} (4.8) y dividir sus columnas por la raíz cuadrada del vector $\boldsymbol{\lambda}$ elemento a elemento.

$$\mathbf{V} = (\mathbf{M}_X' \cdot \mathbf{v})' \quad (4.8)$$

- Los autovectores que buscamos (los de la verdadera matriz de covarianza (4.6), la que provenía de M_X con las imágenes de entrenamiento puestas en columnas) serán las N filas de la matriz finalmente obtenida.

Sin embargo este método no calcula todos los autovectores de Σ_X (4.6) ya que esta matriz, al ser real y simétrica, tiene siempre $w \cdot h$ autovectores. Pero se puede demostrar que los N autovectores conseguidos con este método compacto son los únicos autovectores de Σ_X que pueden tener un autovalor distinto de cero. Desde el punto de vista teórico de PCA, esto quiere decir que el resto de autovectores no capturan ninguna información del vector X al que estamos aplicando dicha técnica.

Volviendo a nuestro problema de representación, todo esto significa que los N autovectores que genera el método compacto son capaces de caracterizar a los vectores correspondientes al conjunto de imágenes de caras de entrenamiento, y el resto de autovectores sólo nos podrían servir para abarcar el resto del espacio original de X (el espacio de las imágenes $w \cdot h$), pero no nos servirían para representar las caras. De entre estos autovectores el número de ellos que contienen información útil (autovalor mayor de cero) es igual al número de vectores linealmente independientes del conjunto de entrenamiento menos uno, ya que al restar la media se reduce siempre en uno el número de vectores linealmente independientes.

De esta manera obtenemos los deseados autovectores que definen nuestro nuevo espacio, el espacio de las caras. Estos vectores tienen realmente apariencia de caras, y por ello han sido llamados *Eigenfaces*. En la Figura 4-2 se muestran la cara media y las 7 primeras eigenfaces de un total de 59 generadas con un conjunto de 60 imágenes de entrenamiento.

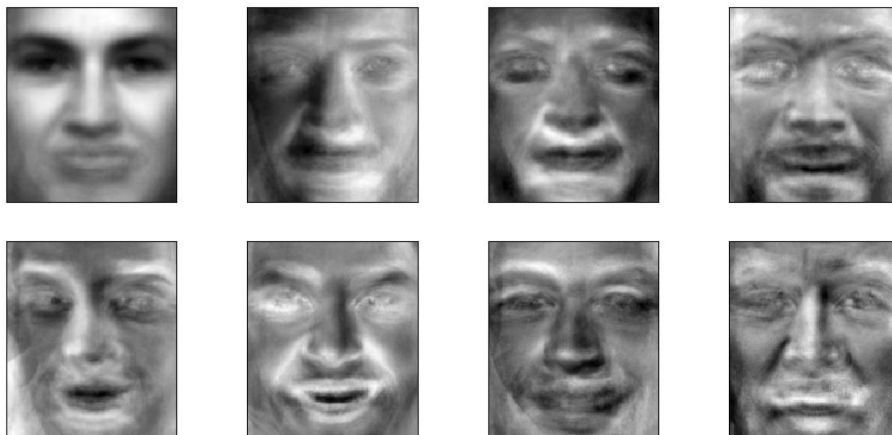


Figura 4-2. Ejemplo de cara media y 7 primeras *eigenfaces*

4.4 Proyección y reducción de la dimensionalidad

Llegados a este punto, la situación es la siguiente: tenemos un conjunto de N imágenes de caras X (que utilizamos como conjunto de entrenamiento) que queremos representar en un espacio de menor dimensión. Para ello hemos considerado estas imágenes como realizaciones del proceso aleatorio vectorial “imágenes de caras” y hemos aplicado PCA a este vector. Como resultado obtenemos N autovectores (*eigenfaces*) ortonormales, que como se ha comentado anteriormente, al haber restado la media se reducen a $N - 1$ vectores linealmente independientes, con los cuales se pueden generar sin error las N caras del conjunto de entrenamiento.

Para hallar el vector \mathbf{Y}_i que contiene las coordenadas de una cara \mathbf{X}_i sobre este nuevo espacio no tenemos más que proyectar \mathbf{X}_i sobre las *eigenfaces*. Para hallar la proyección de \mathbf{X}_i sobre cada *eigenface* bastará con calcular el producto escalar, ya que las *eigenfaces* son ortonormales.

Al haber $N - 1$ eigenfaces la dimensión de los vectores \mathbf{Y}_i también es $N - 1$ (recordemos que N es el número de imágenes de entrenamiento). Esto supone una fuerte reducción de dimensionalidad frente al espacio de las imágenes ($w \cdot h$). Sin embargo, la verdadera reducción de dimensionalidad de la representación mediante *eigenfaces* consiste en utilizar sólo las *eigenfaces* con mayor autovalor asociado. De esta manera se puede reducir notablemente el número de dimensiones de la representación sin perder apenas calidad en la representación de las caras. En este aspecto las *eigenfaces* consiguen la máxima eficiencia que se puede conseguir ya que las primeras *eigenfaces* (las de mayor autovalor asociado) consiguen capturar la información más importante de las caras, mientras que las últimas apenas aportan información (sus autovalores son muy bajos). Dicho de otra manera: la mejor representación posible de un conjunto de N imágenes de caras utilizando sólo M dimensiones (con $M < N$) se consigue proyectándolas sobre las M eigenfaces con mayor autovalor.

5 IMPLEMENTACIÓN DEL SISTEMA

5.1 Entorno de desarrollo

5.1.1 Lenguaje de programación

El lenguaje de programación elegido para la implementación es Python [32], ya que es un lenguaje de programación orientado a objetos muy utilizado en el desarrollo de algoritmos de inteligencia artificial, por lo que existen diferentes librerías y herramientas que pueden ser muy útiles para conseguir los objetivos de una manera más sencilla.

La versión utilizada es la 2.7, la última versión estable dentro de las 2.x. A pesar de que el lenguaje ya ha dado el salto a las versiones 3.x, la página oficial de Python [33] recomienda trabajar todavía con la versión estable 2.7 debido a que las 3.x ofrecen actualmente un soporte de bibliotecas ligeramente menor, por lo que este trabajo podría verse afectado. Otro de los motivos es que la gran mayoría de documentos y códigos Open Source aún se encuentran escritos en Python 2.x.

No obstante, en un futuro no habría problema de traspasar este trabajo a versiones 3.x si fuera necesario, ya que a pesar de que sí existe diferencia internamente, en cuanto a sintaxis la diferencia es muy pequeña.

5.1.2 Entorno de trabajo

El entorno de trabajo utilizado es Spyder, un entorno de programación para Python que incluye la distribución multiplataforma Anaconda desarrollada por Continuum Analytics [34]. Esta distribución se caracteriza por ser libre y por incluir una gran colección de paquetes y librerías para análisis de datos, computación científica e ingeniería. La versión utilizada es Anaconda 4.3.0, la última versión estable en la fecha del comienzo del trabajo.

5.1.3 Librerías y extensiones

Para el desarrollo de la aplicación se hace uso de diferentes librerías que facilitarán la implementación de los algoritmos descritos. Las principales son las siguientes:

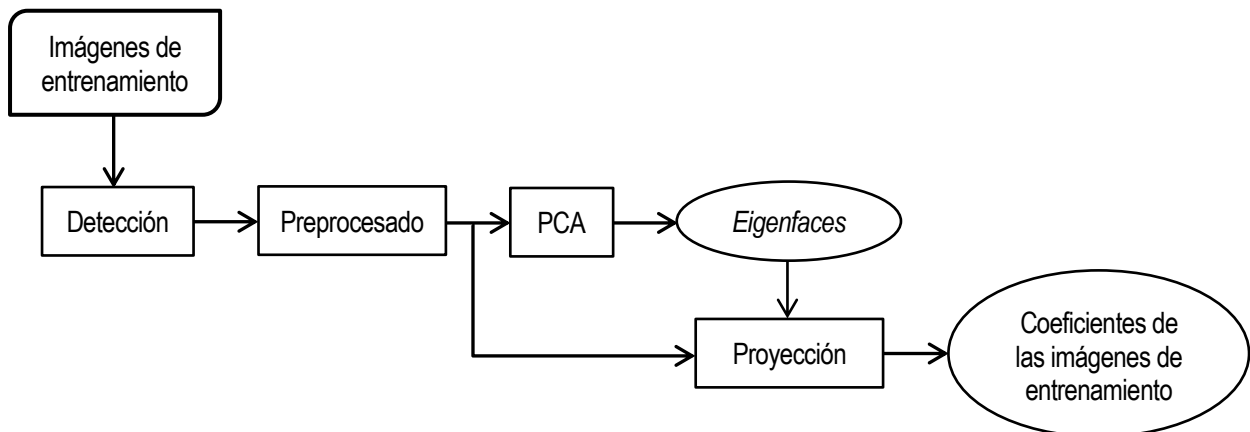
- **OpenCV** [35] (versión 4.3.0) es una librería Open Source originalmente desarrollada por Intel para procesamiento de imágenes y visión por computador. Dispone de más de 500 funcionalidades, entre las que destacan la detección y el reconocimiento de objetos.
- **Numpy** [36] (versión 1.11.3) librería también Open Source que ofrece un gran conjunto de funciones matemáticas. Destaca también por su utilidad a la hora de operar con vectores y matrices.

5.2 Etapas del sistema

Como se vio en el capítulo 3, para que el sistema esté preparado para realizar la fase de comparación es necesario disponer de las características de las imágenes de entrenamiento. Por ello es necesario realizar una vez una etapa de entrenamiento para preparar el sistema. A partir de entonces se puede efectuar el reconocimiento con un conjunto de imágenes de test.

El esquema general del sistema implementado se muestra en la Figura 5-1.

ETAPA DE ENTRENAMIENTO



ETAPA DE TEST

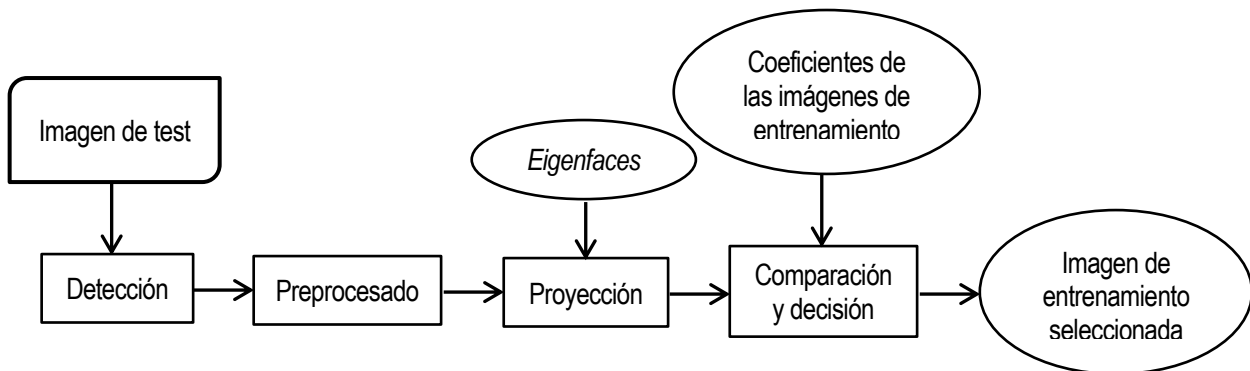


Figura 5-1. Diagrama general del sistema implementado

A continuación se describe en qué consisten cada una de estas dos etapas.

5.2.1 Etapa de entrenamiento

En esta etapa se prepara el sistema para que aprenda a reconocer posteriormente cualquier rostro con los que se haya entrenado. Para ello, es necesario disponer de una serie de imágenes por individuo, tantas como se desee. Es necesario que dichas imágenes sean individuales, de tal forma que solo aparezca el individuo en cuestión. Como bien se muestra en el esquema de la Figura 5-1, el proceso es el siguiente:

- Para cada imagen de cada individuo, se detecta el rostro y ojos, y se preprocesa, de tal forma que se obtenga una imagen en la que únicamente aparezca el rostro ecualizado.
- Una vez se tienen todas las imágenes de entrenamiento preprocesadas, se procede a la extracción de características. Como se explicó en el capítulo 4, el primer paso consiste en aplicar la técnica PCA al conjunto de imágenes preprocesadas para calcular los autovectores que formarán la base ortonormal del nuevo subespacio (*Eigenfaces*).
- Por último se proyectan las imágenes de entrenamiento preprocesadas sobre las *eigenfaces*, de tal forma que se obtienen los coeficientes de las imágenes de entrenamiento en el nuevo subespacio.

5.2.2 Etapa de test

Una vez se han calculado las *eigenfaces* y los coeficientes de las imágenes de entrenamiento en dicho subespacio, solo queda proyectar la imagen de test en cuestión en la nueva base y realizar la comparación. A diferencia de la etapa anterior, en estas imágenes sí que pueden aparecer varios individuos, donde cada uno de ellos pasará a ser detectado y reconocido. El proceso es el que se muestra en el anterior esquema (Figura 5-1):

- Al igual que en la etapa anterior, el primer paso vuelve a ser detectar todos los individuos que aparezcan en la imagen de test y preprocesar sus rostros.
- A partir de las *eigenfaces* ya calculadas, se proyectan cada uno de los rostros en el nuevo subespacio para obtener los coeficientes de las imágenes de estos rostros.
- Teniendo también los coeficientes de las imágenes de entrenamiento, ya se puede para cada rostro de la imagen de test iniciar el reconocimiento. Como se ha comentado anteriormente, en este trabajo se decide utilizar como método de comparación la distancia euclídea, al ofrecer muy buenos resultados siendo una medida muy fácil de implementar. Para ello se calcula dicha distancia entre la proyección del rostro de test en cuestión y cada una de las proyecciones de las imágenes de entrenamiento. La proyección cuya distancia sea menor, indicará de que individuo se trata.

En los sucesivos apartados se detalla cómo se han implementado cada uno de los bloques del diagrama.

5.3 Detección

Es la primera fase del sistema. En esta fase, se realiza la detección tanto del rostro como de los ojos, en ese orden. El orden es importante ya que se detectará primero la región de la cara y, a continuación, se buscará solamente dentro de esta región la localización de los ojos. De esta forma se consigue reducir el tiempo de ejecución del algoritmo, pues la búsqueda de ojos se lleva a cabo dentro de un área más reducida.

Para la detección de rostros y ojos se han utilizado los clasificadores *Haar-like features* bajo el algoritmo de Viola-Jones [11], cuyo funcionamiento se explicó en el capítulo 3. La ventaja de este método es su robustez y bajo coste computacional, lo que permite la detección de rostros en tiempo real.

La implementación se ha hecho con la librería OpenCV [35], que incluye funciones tanto para entrenar clasificadores en cascada como para detectar objetos [37]. Esta librería facilita además una serie de clasificadores ya entrenados, entre los que se incluyen clasificadores entrenados con rostros y ojos, por lo que se han aprovechado dichos clasificadores evitando un previo entrenamiento.

5.3.1 Detección de caras

El procedimiento consiste primero en cargar de la librería el clasificador *Haar* del objeto que se quiere detectar, en nuestro caso el de detección frontal de rostros 'haarcascade_frontalface'. Posteriormente, utilizando la función de detección de la librería es posible variar el tamaño de la ventana mínima de detección e imponer un escalado a la propia imagen para reducir los tiempos de cómputo. El sistema se ha diseñado para detectar todos los rostros que aparezcan en la imagen, como se muestra en la Figura 5-2.

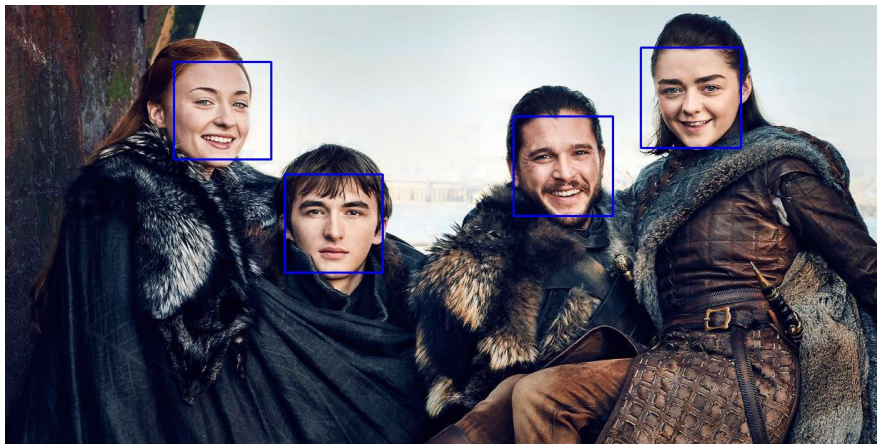


Figura 5-2. Resultado de la detección de caras del sistema implementado

5.3.2 Detección de ojos

El proceso de detección de ojos es muy similar al de detección de rostros. Sin embargo, este punto adquiere más importancia en la fase de preprocesado ya que, conociendo la posición de los ojos, se puede llegar a determinar el ángulo de rotación de la cara para alinearla.

A la hora de detectar se puede hacer distinción entre ojo derecho y ojo izquierdo o detectar ambos como un conjunto. Se ha optado por utilizar un clasificador que sí realice dicha distinción, para obtener un resultado más completo y poder afinar aún más la detección. No obstante en este caso hay que tener más cuidado a la hora de la detección, ya que como se van detectando los ojos de forma independiente puede que se detecte el mismo ojo tanto como izquierdo como derecho, o puede que los detecte de manera inversa. El sistema se ha implementado para que solucione estos problemas.

El procedimiento es similar al de la detección de caras salvo que ahora se cargan de la librería los clasificadores *Haar* para la detección de ojos ‘haarcascade_lefteye’ y ‘haarcascade_righteye’. Al igual que en el caso anterior, el sistema también se ha diseñado para que detecte todos los ojos de todos los rostros que aparezcan en la imagen, como se muestra en la Figura 5-3.

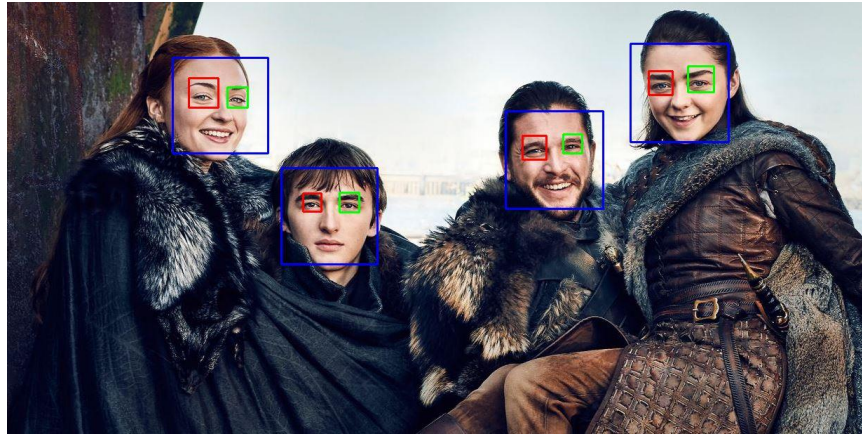


Figura 5-3. Resultado de la detección de caras y ojos del sistema implementado

5.4 Preprocesado

En esta segunda fase, se le aplican una serie de transformaciones a cada rostro detectado anteriormente para que la extracción de características sea lo más efectiva posible. Al preprocesar las imágenes, sólo la región de la cara queda visible, eliminando orejas, cabello y barbilla. Las operaciones que se llevan a cabo en esta fase son la rotación, el escalado, el recorte y la ecualización del histograma. Tras finalizar esta fase se obtiene una imagen por cada rostro detectado, en la que aparece únicamente el rostro ecualizado.

Todos los rostros ecualizados deben tener un mismo tamaño y coordenadas de los ojos para que sea posible la comparación entre ellas. En este trabajo se ha optado por seguir el formato de imagen recogido por el estándar ISO/IEC 19794-5 [38], que define un área similar a una foto de carnet, con unas dimensiones de 168x192 píxeles (ancho x alto) y una distancia entre centros de los ojos de 96 píxeles.

Se mostrarán los resultados de esta fase para una imagen en la que solo aparezca un rostro (Figura 5-4), pues las transformaciones se aplican de la misma manera para cada uno de los rostros que aparezcan en la imagen.

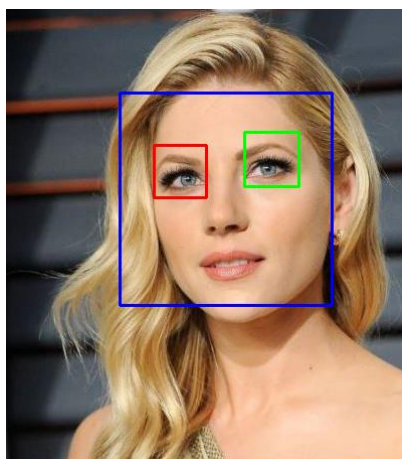


Figura 5-4. Ejemplo de imagen de un rostro para el preprocesado

5.4.1 Rotación

La primera transformación que se lleva a cabo es la de rotación. Utilizando la información de las coordenadas de los centros de los ojos obtenidas en la etapa de detección, es posible calcular mediante un cálculo trigonométrico el ángulo de giro de la cara detectada respecto al eje horizontal y compensarlo mediante una transformación afin de rotación. En la Figura 5-5 podemos ver el resultado de la transformación de rotación del ejemplo, una vez se ha detectado el rostro y la posición de los ojos.



Figura 5-5. Resultado de la rotación

5.4.2 Escalado

El escalado consiste en proporcionar un tamaño determinado al rostro ya rotado y nivelado de forma que todos tengan las mismas proporciones. Para realizar esta operación existen diversas alternativas, como se ha comentado anteriormente en este trabajo se ha optado por seguir el estándar propuesto por la norma ISO/IEC 19794- 5 [38] que aconseja que la distancia entre el centro de los dos ojos sea como máximo de 96 píxeles (Figura 5-6). Tomando esa distancia como referencia para el proceso de escalado, se consigue que todas las imágenes tengan unas proporciones muy similares facilitando, así, las tareas de comparación.



Figura 5-6. Resultado del escalado

Debido a que las dimensiones de las imágenes aquí mostradas no se corresponden con las dimensiones reales de las mismas al tener que ser adaptadas al documento, la diferencia de tamaño que se aprecia por el escalado entre la Figura 5-5 y 5-6 no es la real que se ha obtenido.

5.4.3 Recorte

Después del proceso de escalado, es el momento de recortar la imagen. Se le da a todas las imágenes las mismas dimensiones finales. Del mismo modo que en el punto anterior, se sigue el estándar que marca la norma en [38]. Así, las fotografías finales tienen unas dimensiones de 168 x 192 píxeles (ancho x alto), como se muestra en la Figura 5-7.



Figura 5-7. Resultado del recorte

5.4.4 Ecuación del histograma

Por último, para acabar con el proceso de normalización, se realiza una ecualización del histograma (Figura 5-8) a la imagen recortada. El objetivo de esta transformación es conseguir que el histograma de la imagen convertida a escala de grises sea lo más uniforme posible, es decir, que el número de píxeles para cada nivel de gris sea lo más igualado posible. Con este proceso se consiguen mitigar los cambios de luminosidad que afectan negativamente al algoritmo de extracción de características que se utilizará, PCA.



Figura 5-8. Resultado de la ecualización del histograma

5.5 Extracción de Características

5.5.1 Cálculo de las *Eigenfaces*

Una vez se han preprocesado todas las imágenes de entrenamiento, el siguiente paso es calcular los autovectores que formarán la base ortonormal del nuevo subespacio, las llamadas *Eigenfaces*. Como se vio en el capítulo 4, para extraer las *eigenfaces* se realiza PCA mediante los siguientes pasos:

- Se redimensionan cada una de las imágenes de entrenamiento preprocesadas en un vector y se forma la matriz \mathbf{M}_X cuyas filas serán dichos vectores.
- Se le resta a la matriz \mathbf{M}_X su media.
- Como siempre será mayor la dimensión de las imágenes de entrenamiento que el número total de imágenes, se puede aplicar el método compacto [31] visto anteriormente. Este método consistía en trabajar con la matriz de covarianza de la matriz \mathbf{M}_X , donde mediante sus autovectores y autovectores se conseguían calcular de una forma rápida las deseadas *eigenfaces*. (Los pasos detallados se muestran en el apartado 4.3).
- No obstante, se contemplan también los posibles casos en los que la dimensión de las imágenes sea menor que el número total de las mismas. En estos casos no se puede emplear el método compacto y al tratarse de imágenes relativamente pequeñas se recurre a la descomposición en valores simples (SVD) [30] de la matriz \mathbf{M}_X , también explicada en el capítulo 4, para obtener las *eigenfaces*.

5.5.2 Proyección sobre las *Eigenfaces*

Las *eigenfaces* forman un conjunto ortonormal. Por lo tanto, para hallar la proyección de una imagen sobre ellas basta con realizar el producto escalar de la imagen sobre cada una de las *eigenfaces*. De esta forma se consigue proyectar en el nuevo subespacio los rostros tanto de las imágenes de entrenamiento como de la imagen de test, obteniendo así los coeficientes de dichas imágenes sobre la nueva base.

Además, para evitar que dos vectores con la misma dirección pero distintas longitudes parezcan distintos, se normalizarán antes de ser proyectados.

5.6 Comparación y decisión

Finalizada la extracción de características se llega a la última fase, cuyo objetivo es determinar qué imagen del conjunto de entrenamiento es más parecida a la imagen de test, a partir de sus representaciones mediante las *eigenfaces* (sus proyecciones). Para calcular la distancia se usa la distancia euclídea con el fin de tomar una decisión.

En esta fase se analiza la distancia entre la proyección de la imagen de test donde se quiere hacer el reconocimiento y las correspondientes a las imágenes de entrenamiento. Se busca pues, la proyección de las imágenes de entrenamiento que más cerca esté de la de la imagen de test. Esta proyección se corresponderá con el individuo que más similitud presente con la muestra analizada.

6 EXPERIMENTOS REALIZADOS Y RESULTADOS

El objetivo de esta fase es evaluar el rendimiento que ofrece el sistema implementado. Los experimentos que se llevan a cabo para evaluar diferentes características del funcionamiento del sistema se dividen en dos secciones, experimentos relacionados con la detección y experimentos relacionados con el reconocimiento.

Para llevar a cabo el conjunto de pruebas se utilizarán dos bases de datos estándar de imágenes: ORL Face Database [39] y Normalized Yale Face Database [40]. No obstante, para verificar que el sistema funciona con cualquier tipo de imágenes y no solo con bases de datos ya creadas donde aparecen las imágenes de los rostros preparadas, también se experimentará con una base de datos creada manualmente con imágenes cotidianas buscadas en Google de diferentes personajes conocidos en la actualidad. Lo que demostrará que el sistema implementado funciona con cualquier base de datos con la que trabaje el usuario.

6.1 Bases de datos utilizadas

6.1.1 ORL Face Database

Es la base de datos de AT&T [41]. Contiene 40 individuos con 10 imágenes para cada uno, haciendo un total de 400 imágenes tal y como detalla la Tabla 6-1. Las imágenes fueron tomadas en diferentes momentos, variando la pose, las expresiones faciales (ojos abiertos / cerrados, sonriendo / no sonriendo) y detalles faciales (gafas / no gafas). Todas las imágenes fueron tomadas contra un fondo oscuro homogéneo con los sujetos en una posición frontal, pero con una cierta tolerancia para algún movimiento lateral. En la Figura 6-1 se muestran las imágenes para tres de los sujetos.

Individuos totales	Hombres	Mujeres	Tamaño de las imágenes	Imágenes por individuo	Número total de imágenes
40	40	0	92x112 px	10	400

Tabla 6-1. Datos sobre la base de datos ORL



Figura 6-1. Imágenes de tres sujetos de la base de datos ORL

6.1.2 Normalized Yale Face Database

Es la base de datos de la universidad de Yale. Contiene 165 imágenes de 15 individuos, con 11 imágenes por individuo como se detalla en la Tabla 6-2. Cada individuo aparece en diferentes condiciones de luminosidad y exposición a la luz, y diferentes expresiones faciales (sonriendo, triste, sorprendido, guiñando un ojo, etc) pero siempre en una posición frontal. Un ejemplo de los cambios de iluminación y expresiones sobre un mismo individuo aparece en la Figura 6-2.

Individuos totales	Hombres	Mujeres	Tamaño de las imágenes	Imágenes por individuo	Número total de imágenes
15	14	1	195x231 px	11	165

Tabla 6-2. Datos sobre la base de datos Yale



Figura 6-2. Imágenes de cuatro sujetos de la base de datos Yale

6.1.3 Base de datos propia

Se ha construido una base de datos formada por 8 personajes célebres con 11 imágenes por individuo, como se detalla en la Tabla 3. Son imágenes cotidianas, por lo que no están preparadas como las bases de datos anteriores. Por tanto, esta será la base de datos con la que mejor se pueda comprobar el funcionamiento de la detección, pues no serán imágenes donde aparezca únicamente el rostro del individuo.

Individuos totales	Hombres	Mujeres	Tamaño de las imágenes	Imágenes por individuo	Número total de imágenes
8	4	4	Variable	11	88

Tabla 6-3. Datos sobre la base de datos propia

En la Figura 6-3 se muestran algunas de las imágenes elegidas para cada individuo. Se han ajustado las imágenes a un tamaño cuadrado para adaptarlas al documento y poder mostrarlas de manera ordenada. No obstante, en los experimentos se utilizan dichas imágenes con su tamaño original, el cual es variable.



Figura 6-3. Algunas de las imágenes utilizadas para cada individuo de la base de datos propia

6.2 Evaluación de la detección

La detección es una de las fases más importante de la aplicación, ya que una mala detección conllevaría un error en el resto de las fases. Como se ha comentado anteriormente, la base de datos con la que mejor se puede evaluar el funcionamiento de esta fase es con la base de datos propia, debido a que son imágenes cotidianas, no preparadas para el reconocimiento facial.

El sistema implementa para la detección el algoritmo de Viola-Jones [11], el cual necesita que los individuos aparezcan en las imágenes de frente y con el rostro limpio sin ningún tipo de elemento o accesorios para que la detección sea lo más correcta posible. En la Figura 6-4 se puede comprobar el funcionamiento del algoritmo de la detección sobre algunas de las imágenes de la base de datos.

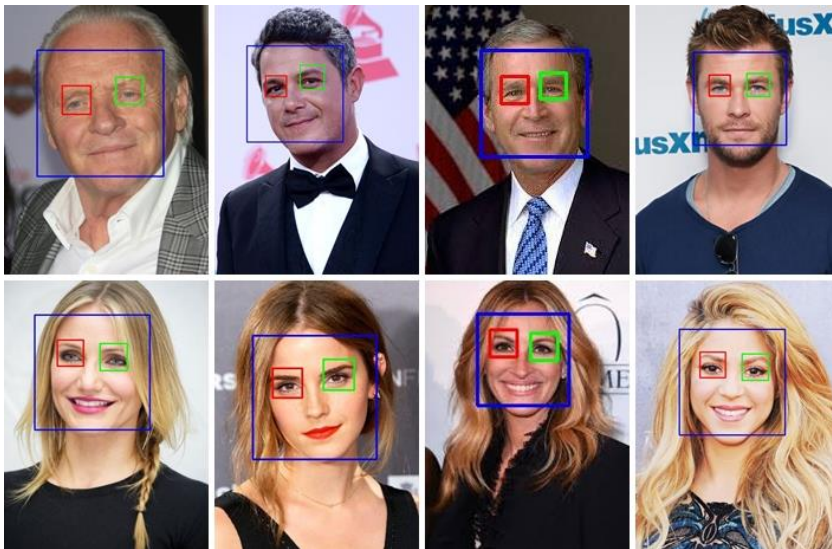


Figura 6-4. Imágenes de individuos con los rostros despejados, aptas para la detección

Existen casos particulares en los que el algoritmo no funciona correctamente. A continuación se muestran algunos ejemplos de imágenes que no servirían, ya que no es posible la detección del rostro y/o ojos.

- **Individuos de perfil:** en este caso no se consigue detectar el rostro y por lo tanto no se inicializa la detección de los ojos, pues es necesaria la detección correcta de ambas partes.



Figura 6-5. Imágenes de individuos de perfil, no aptas para la detección

- **Elementos o accesorios tapando parte del rostro:** en el primer caso se ve como al aparecer el individuo de frente sí se detecta correctamente el rostro, sin embargo debido a las gafas de sol, no es posible detectar los ojos. En la segunda imagen, la mano y el micrófono tapan parte del rostro, produciendo además sombras perjudiciales, por lo que no es posible la detección correcta del rostro. Por último en el tercer caso, el pelo tapa parte de rostro e incluso un ojo al completo, por lo que tampoco funciona correctamente el algoritmo.



Figura 6-6. Imágenes de individuos con el rostro sin despejar, no aptas para la detección

- **Condiciones de iluminación:** la detección, además de estar condicionada por la posición y orientación de la cara del sujeto con respecto a la cámara, también depende fuertemente de las condiciones de iluminación. En la primera imagen se ve como el ojo derecho no es detectado debido a aparecer en una zona muy oscura. En el segundo caso el individuo aparece detrás de una ventana y esto hace que no aparezca nítido, aun así el rostro sí se consigue detectar, pero no los ojos. Por último en el tercer caso la imagen es muy oscura, apareciendo además sombras en el rostro sobre todo en la zona de los ojos, lo que dificulta mucho la detección de ambas partes.

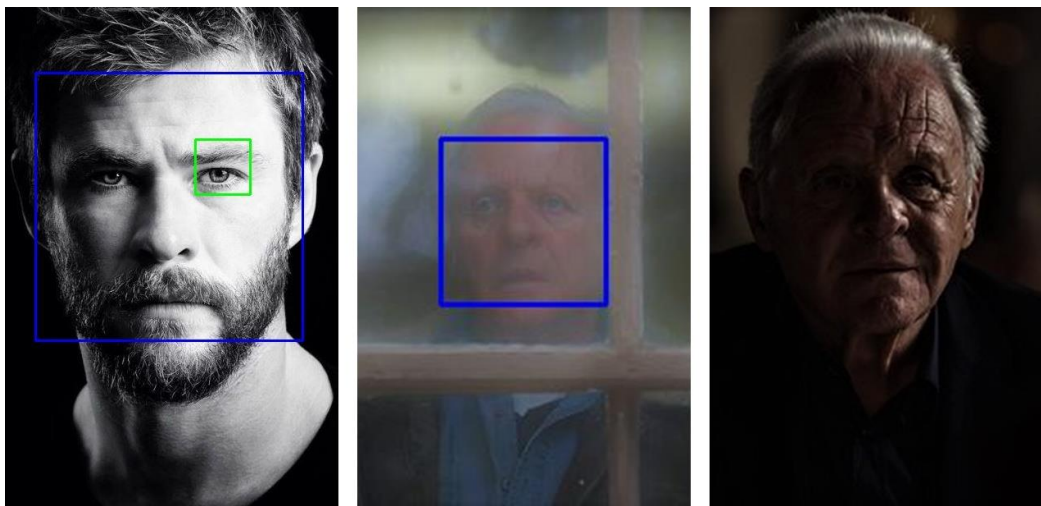


Figura 6-7. Imágenes de individuos con condiciones de iluminación no aptas para la detección

Sería posible procesar las imágenes antes de llevar a cabo la detección para intentar evitar casos como los aquí mencionados y mejorar los resultados, pero esto no es objetivo del trabajo.

Para terminar, aclarar que las otras dos bases de datos, ORL y Yale, ya contienen las imágenes de los individuos con sus rostros debidamente recortados y centrados, por lo que ya están las imágenes preparadas para iniciar la extracción de características y reconocimiento sin necesidad de realizar la previa detección del rostro y de los ojos.

6.3 Evaluación del reconocimiento

6.3.1 Variación del número de imágenes de entrenamiento

Se realizarán pruebas de reconocimiento con cada una de las tres bases de datos. Para ello se han hecho diferentes experimentos variando el número de imágenes de entrenamiento por individuo, obteniendo los resultados en término de acierto y error porcentuales. De entre todas las imágenes de entrenamiento de cada individuo se escogen aleatoriamente tantas imágenes para entrenar como se deseen, probando posteriormente en la etapa de test con las imágenes restantes, ya que si se someten al reconocimiento imágenes con las que se ha entrenado el acierto siempre sería del 100%. Además, para un número de imágenes de entrenamiento fijado, se repite 10 veces el experimento variando aleatoriamente la selección de dichas imágenes de entrenamiento. En las tablas siguientes aparece la media obtenida de los 10 experimentos para cada caso, así como la desviación típica.

Se comenzará con las dos bases de datos estándar ORL y Yale. Recordemos que la base de datos ORL estaba caracterizada por las variaciones de pose de los individuos, ya que a pesar de que todos se encuentran en una posición frontal, tienen cierta tolerancia hacia algún movimiento lateral. La base de datos Yale sí contiene todos sus individuos en posición frontal. Ésta se caracteriza por variar las condiciones de luminosidad y exposición a la luz, además de tener expresiones faciales más variadas que la base de datos ORL.

Como se puede ver en dichos resultados (Tabla 6-4 y 6-5), a medida que se va aumentando el número de imágenes de entrenamiento el porcentaje de acierto es mayor, ya que se va entrenando el sistema con un mayor número de imágenes de cada individuo, lo que facilita su posterior reconocimiento. Ambas bases de datos obtienen buenos resultados, alcanzando el 90% de acierto con 5 imágenes de entrenamiento para la base de datos ORL y con 7 para la de Yale.

Viendo estos resultados se puede concluir que el sistema implementado basado en técnicas PCA ofrece mejores resultados frente a variaciones de pose (ORL) que frente a variaciones en la iluminación (Yale). Además, la base de datos Yale contiene variaciones más fuertes en las expresiones faciales que la de ORL, por lo que también puede verse afectada en los resultados por este motivo.

BASE DE DATOS ORL

Entrenamiento		Test	Porcentaje medio de acierto	Desviación típica
Img / Indiv	Total	Total		
1	40	360	61.42%	3.27
2	80	320	76.31%	1.58
3	120	280	84.29%	2.41
4	160	240	88.79%	1.56
5	200	200	92.75%	2.14
6	240	160	93.88%	1.15
7	280	120	95.33%	0.67
8	320	80	96.25%	1.12
9	360	40	98.00%	1.00

Tabla 6-4. Resultados de la variación de imágenes de entrenamiento en la base de datos ORL

BASE DE DATOS YALE

Entrenamiento		Test	Porcentaje medio de acierto	Desviación típica
Img / Indiv	Total	Total		
1	15	150	61.60%	9.99
2	30	135	78.67%	2.41
3	45	120	82.67%	2.32
4	60	105	85.43%	2.73
5	75	90	88.33%	2.50
6	90	75	88.40%	2.98
7	105	60	91.17%	3.66
8	120	45	91.78%	4.34
9	135	30	92.00%	4.52
10	150	15	97.33%	6.80

Tabla 6-5. Resultados de la variación de imágenes de entrenamiento en la base de datos YALE

La base de datos propia contiene imágenes cotidianas de los sujetos, es decir, sin estar tomadas en unas condiciones preparadas para el reconocimiento facial como las otras dos bases de datos anteriores. Por ello hay que tener en cuenta que las imágenes están sometidas a diferentes condiciones de iluminación, variaciones de pose y de expresiones faciales, además de poder sufrir también otro tipo de variaciones en los rasgos al no estar tomadas en los mismos intervalos de tiempo. Por tanto, como muestra la Tabla 6-6, ahora es necesario un número mayor de imágenes de entrenamiento para conseguir un porcentaje de acierto similar al de las otras bases de datos. No obstante, los resultados siguen siendo satisfactorios.

BASE DE DATOS PROPIA

Entrenamiento		Test	Porcentaje medio de acierto	Desviación típica
Img / Indiv	Total	Total		
1	8	80	51.38%	7.76
2	16	72	59.58%	5.14
3	24	64	65.63%	8.98
4	32	56	72.68%	5.30
5	40	48	78.96%	3.66
6	48	40	80.50%	8.28
7	56	32	85.94%	4.25
8	64	24	90.00%	4.64
9	72	16	95.63%	6.88
10	80	8	98.75%	11.79

Tabla 6-6. Resultados de la variación de imágenes de entrenamiento en la base de datos propia

6.3.2 Variación del número de *eigenfaces*

En el capítulo 4 se vio como gracias a la técnica PCA se conseguía representar un conjunto de N imágenes de caras de entrenamiento en un espacio de dimensión N en lugar de $w \cdot h$, siendo la base ortonormal de este nuevo subespacio (*eigenfaces*) los autovectores de la matriz de covarianza. Sin embargo, también se explicó que la verdadera reducción de dimensionalidad consistía en no utilizar las N *eigenfaces*, sino solo aquellas que tuvieran mayor autovalores.

En todos los experimentos anteriores se ha ido variando el número de imágenes de entrenamiento pero siempre se ha mantenido el número de las *eigenfaces* utilizadas en el total de las mismas (N). En este apartado se va a evaluar cómo afecta el utilizar solo las *eigenfaces* con mayor poder discriminante, es decir, eliminando aquellas con menores autovalores. Para ello se realizarán experimentos con, por ejemplo, 7 imágenes de entrenamiento por individuo y se irá disminuyendo el número de *eigenfaces*, comparando los resultados de acierto y error con los que se obtuvo en los experimentos anteriores al utilizar las N *eigenfaces* totales. Al igual que en los experimentos anteriores, para cada valor de *eigenfaces* elegido, se realizarán 10 experimentos variando aleatoriamente la selección de las 7 imágenes de entrenamiento, obteniendo finalmente la media de los porcentajes y la desviación típica.

En las tablas siguientes se muestran los resultados de estos experimentos para cada una de las tres bases de datos. Como se puede comprobar, disminuir el número de *eigenfaces* no siempre conlleva a disminuir el porcentaje de acierto. Se puede ver en las tablas como hay casos en los que disminuyendo dichas *eigenfaces*, se mantiene o aumenta dicho porcentaje. La gran ventaja de trabajar con menos *eigenfaces* es que supone una notable reducción del coste computacional, al trabajar con dimensiones menores.

La base de datos estándar ORL es la que dispone de un mayor conjunto de imágenes de entrenamiento, con un total de 280. Se puede comprobar en la Tabla 6-7 como disminuyendo notablemente el número de *eigenfaces* a 50, se consigue mantener, e incluso aumentar ligeramente, el porcentaje de acierto. No obstante, reduciendo este número aún más a 20 o 10, el porcentaje solo disminuye un 1% y 2% respectivamente respecto al conseguido con las *eigenfaces* totales. Por tanto, en esta base de datos se puede llegar a mantener el mismo porcentaje de acierto reduciendo considerablemente la dimensionalidad.

BASE DE DATOS ORL

Eigenfaces	Test	Porcentaje medio de acierto	Desviación típica
Utilizadas	Total		
280 (total)	120	95.33%	4.67
200	120	93.83%	6.17
150	120	95.42%	4.58
100	120	94.50%	5.50
50	120	95.42%	4.58
30	120	94.83%	5.17
20	120	94.33%	5.67
10	120	89.33%	10.67
5	120	76.50%	23.50

Tabla 6-7. Resultados de la variación del número de *eigenfaces* en la base de datos ORL

En cuanto a la base de datos Yale (Tabla 6-8) y la base de datos propia (Tabla 6-9), no se consigue mantener tan bien el porcentaje con una gran disminución de las *eigenfaces* como en el caso anterior. No obstante, siempre se puede llevar a cabo una reducción de la dimensionalidad dependiendo del porcentaje de acierto que se disponga a perder. Cualquier disminución de las *eigenfaces*, por muy pequeña que sea, supone un ahorro en cuanto a coste computacional, por lo que siempre dependerá del uso que se le quiera dar a la aplicación si darle más peso al porcentaje de acierto o al coste.

BASE DE DATOS YALE

Eigenfaces	Test	Porcentaje medio de acierto	Desviación típica
Utilizadas	Total		
105 (total)	60	91.17%	3.66
90	60	89.83%	3.37
70	60	86.50%	2.83
50	60	87.83%	4.28
30	60	86.33%	1.63
20	60	85.33%	4.33
15	60	84.17%	4.84
10	60	79.50%	4.48
5	60	71.17%	4.72

Tabla 6-8. Resultados de la variación del número de *eigenfaces* en la base de datos YALE

BASE DE DATOS PROPIA

Eigenfaces	Test	Porcentaje medio de acierto	Desviación típica
Utilizadas	Total		
56 (total)	32	85.94%	4.25
45	32	82.19%	6.10
35	32	79.38%	4.68
25	32	78.44%	6.16
20	32	78.13%	6.25
15	32	76.88%	3.19
10	32	75.94%	6.56
5	32	73.44%	3.20

Tabla 6-9. Resultados de la variación del número de *eigenfaces* en la base de datos propia

6.4 Representación de los resultados

El sistema está preparado para mostrar la imagen de test indicando el rostro y ojos detectados, además del nombre del individuo que se ha reconocido, como muestra la Figura 6-8.

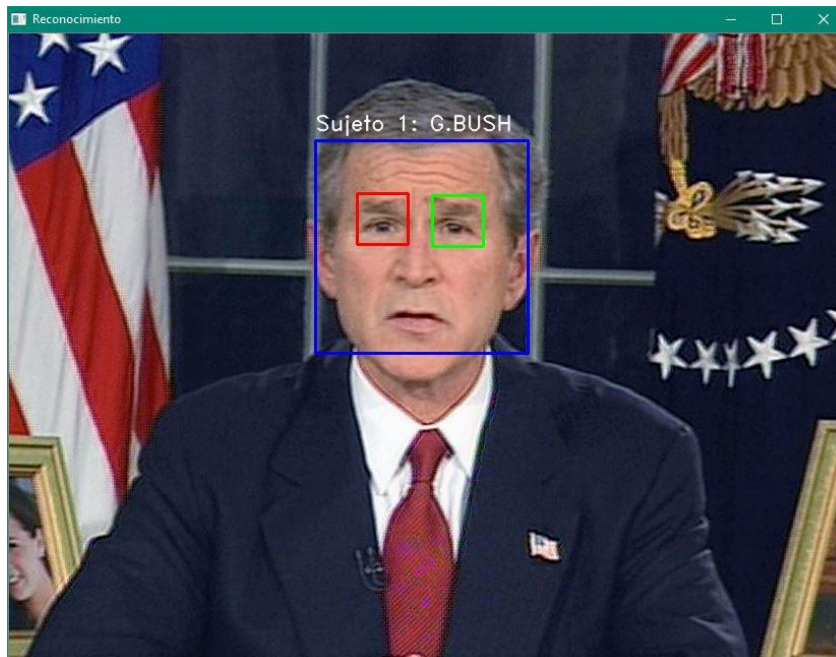


Figura 6-8. Salida que muestra el sistema al finalizar el reconocimiento

Además como ya se comentó, el sistema no solo está preparado para trabajar con imágenes en las que aparezca un solo individuo, sino para que detecte y reconozca todos los rostros que aparezcan en la imagen. Debido a no disponer de imágenes en la que aparezcan juntos varios individuos de las bases de datos, para mostrar el correcto funcionamiento del sistema también en estos casos, se han unido varias de las imágenes individuales de la base de datos propia para formar una sola, obteniendo así una imagen con varios sujetos. El resultado del reconocimiento puede verse en la Figura 6-9.

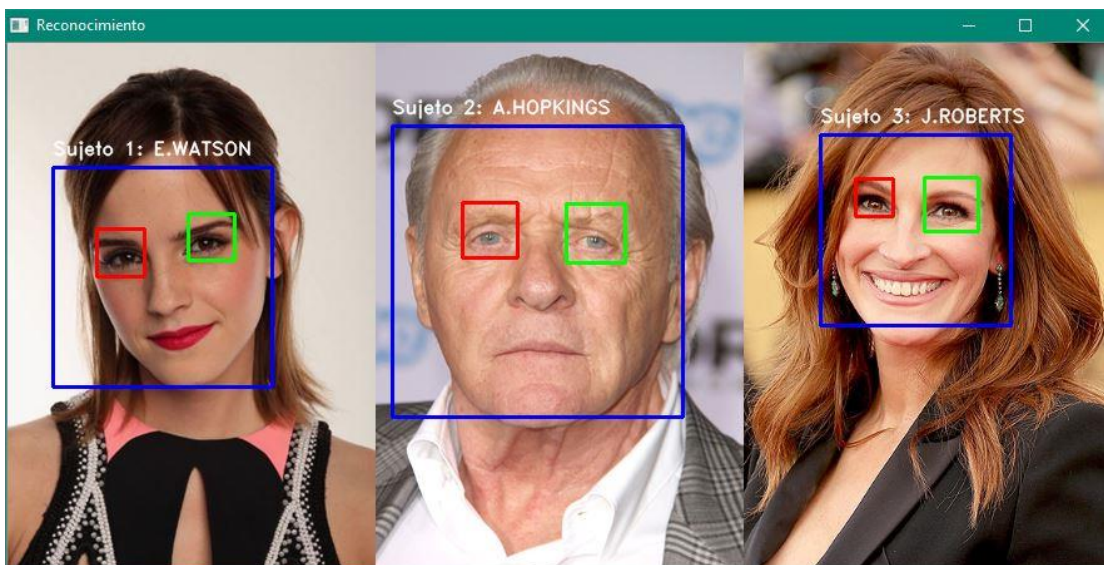


Figura 6-9. Resultado del reconocimiento para una imagen con más de un individuo

Por último, en los casos en los que se detecte el rostro correctamente pero no los ojos, y por tanto, no se pueda proceder al reconocimiento, en lugar del nombre del individuo aparecerá el mensaje *ERROR*, como se muestra en la Figura 6-10.

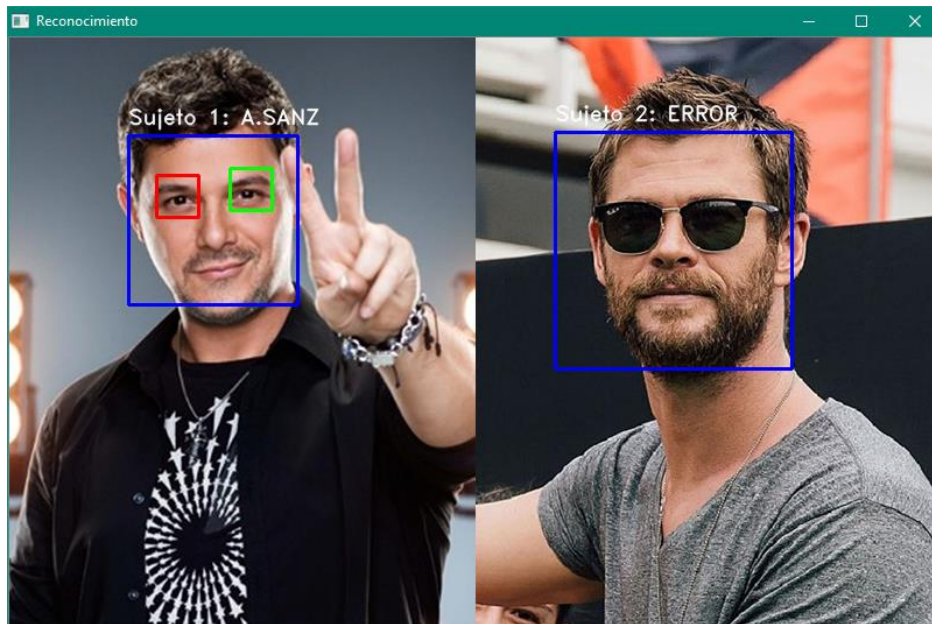


Figura 6-10. Resultado del reconocimiento en caso de error

7 CONCLUSIONES Y LÍNEAS FUTURAS

A lo largo de este trabajo se ha podido ver un estudio de las técnicas de reconocimiento facial más comunes así como una clasificación de las mismas. También se ha visto el funcionamiento de los sistemas de detección y reconocimiento facial y se han descrito los pasos para el diseño y desarrollo de una aplicación de estas características. Finalmente, se han evaluado los resultados del sistema implementado en una fase de pruebas. En este capítulo final se estudia el resultado obtenido para después proponer posibles mejoras y líneas de investigación futuras.

A pesar de que han ido surgiendo diferentes problemas, principalmente a lo largo de la implementación del sistema, se han conseguido solventar a tiempo. De esta forma, se ha conseguido un sistema final funcional que cumple con todos los objetivos que se habían propuesto durante las primeras fases del trabajo.

En dichos objetivos se indicó que el sistema permitiría reconocer rostros tanto en imágenes como en videos en tiempo real. Debido a la imposibilidad de mostrar experimentos de videos en un documento a papel, solo se han expuesto pruebas referentes a imágenes. No obstante, el sistema también está preparado para trabajar con videos, donde se realizaría el mismo proceso de reconocimiento mostrado para las imágenes, fotograma a fotograma. Además, y para introducir el trabajo en el ámbito del tiempo real, también está preparado para trabajar con desde una *webcam*.

7.1 Conclusiones

Una de las conclusiones más importantes que se puede obtener es la gran importancia que se lleva el paso de la detección. A pesar de ser un trabajo sobre reconocimiento facial, como ya se ha comentado en varias ocasiones siempre hay un paso previo que es la detección y que es crucial para el proceso. Para llevarla a cabo se ha utilizado uno de los algoritmos más usados en la actualidad y se ha verificado su correcto funcionamiento. No obstante, si no se trata de imágenes con caras frontales, su rendimiento no está totalmente optimizado y el rendimiento obtenido puede ser inferior.

En cuanto al reconocimiento facial, es necesario someter al sistema a un buen entrenamiento. En el capítulo anterior se ha visto como se obtienen mejores resultados a medida que se van aumentando el número de imágenes de entrenamiento de los individuos. Por tanto un buen sistema de reconocimiento debe disponer de una gran base de datos, incluyendo imágenes de los individuos con distintas poses y condiciones de iluminación para facilitar el posterior reconocimiento lo mejor posible. Además, como ocurría en los experimentos con la base de datos ORL y Yale, si se entrena con imágenes de individuos debidamente preparadas y recortadas, se puede evitar posibles fallos en la detección y conseguir mejores resultados en el reconocimiento.

Otra de las conclusiones que se puede obtener de los experimentos realizados en la parte del reconocimiento, tiene que ver con la reducción de la dimensionalidad. Se ha visto como reduciendo el número de *eigenfaces* se puede seguir manteniendo aproximadamente el mismo porcentaje de acierto que con el total de las mismas. Incluso también hay casos puntuales en los que dicho porcentaje puede mantenerse o incluso aumentar al disminuir las *eigenfaces*. Esto conlleva a un compromiso entre porcentaje de acierto y coste

computacional que dependerá exclusivamente del uso que se le quiera dar a la aplicación. Reducir el número de *eigenfaces* utilizadas posiblemente disminuirá el porcentaje de acierto, pero también supondrá una notable reducción del coste computacional al trabajar con dimensiones menores.

Por último, destacar el buen funcionamiento del método elegido para la extracción de características PCA, ofreciendo un buen compromiso entre complejidad, rapidez de ejecución y resultados, dando muy buenos resultados en bases de datos que ofrecen imágenes homogéneas en cuanto a iluminación, aunque contengan variaciones en la pose de los individuos y siendo algo más sensible a bases de datos con imágenes que ofrecen cambios más notables en las condiciones de iluminación.

7.2 Líneas futuras

Con el objetivo de continuar y mejorar lo estudiado en este proyecto, se proponen posibles líneas futuras de trabajo. No obstante, además de lo aquí mencionado, al tratarse de un sistema completamente modular siempre se pueden cambiar fácilmente las técnicas aquí empleadas en cada una de las fases por otras de las mencionadas en el capítulo 3, ya sea con el propósito de comparar resultados o de buscar alguna mejoría.

Uno de los puntos más buscados y también más costosos en todos los sistemas de reconocimiento facial, consiste en que no se requiera que el individuo aparezca de frente y con el rostro completamente limpio. Se podría intentar mejorar la detección del sistema en este aspecto, pues como se ha comentado anteriormente tiene esta limitación. De esta manera se ampliarían las posibilidades de reconocer a los individuos en otras poses como por ejemplo de perfil o con elementos que tapen parte de la cara, al entrenar el sistema también con imágenes de estas características.

Por último, una de las mejoras más importantes que se podrían añadir al trabajo en un futuro, sería llevar a cabo una interfaz gráfica para trabajar con el sistema en tiempo real, generando una herramienta que serviría de base para desarrollos aplicados al reconocimiento facial en tiempo real y a la videovigilancia. El sistema actual está también preparado para trabajar con la *webcam* de un ordenador, se podría modificar para que trabajase con una cámara instalada en un recinto que, junto con una interfaz funcional, permitiese controlar y evaluar el sistema de forma interactiva.

REFERENCIAS

- [1] W. W. Bledsoe, The model method in facial recognition, Technical report, Panoramic Research Inc, 1966.
- [2] L. Sirovich and M. Kirby, Low-dimensional procedure for the characterization of human, Journal of The Optical Society of America A-optics Image Science and Vision, 4, 1987.
- [3] T. Kohonen, Self-organization and associative memory: 3rd edition, Springer-Verlag New York, Inc, 1989.
- [4] M. A. Turk and A. P. Pentland, Face recognition using eigenfaces. In Computer Vision and Pattern Recognition, Proceedings CVPR '91., IEEE Computer Society Conference, 1991.
- [5] A. K. Jain and S. Z. Li, Handbook of Face Recognition, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
- [6] S. Z. Li, U. Park and A. K. Jain, A discriminative model for age invariant face, IEEE Transactions on Information Forensics and Security, 2011.
- [7] C. Sanderson and B. C. Lovell, Multi-region probabilistic histograms for robust and scalable identity inference, In Proceedings of the Third International Conference on Advances in Biometrics, ICB '09, Berlin, Heidelberg. Springer-Verlag, 2009.
- [8] Y. Taigman, M. Yang and M. Ranzato, DeepFace: Closing the Gap to Human-Level Performance in Face Verification, Conference on Computer Vision and Pattern Recognition, California. Facebook AI Research, 2014.
- [9] F. Schroff, K. Kalenichenko and J. Philbin, FaceNet: A Unified Embedding for Face Recognition and Clustering, Computer Society Conference on Computer Vision and Pattern Recognition, California. Google, 2015.
- [10] N. Zhang, M. Paluri, Y. Taigman, R. Fergus and L. Bourdev, Beyond Frontal Faces: Improving Person Recognition Using Multiple Cues, International Conference on Computer Vision and Pattern Recognition, Berkeley. Facebook AI Research, 2015.
- [11] P. Viola and M. Jones, Rapid object detection using a boosted cascade of simple features, In Computer Vision and Pattern Recognition. Proceedings of the 2001 IEEE Computer Society Conference, 2001.
- [12] S. G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1989.

- [13] Y. Freund and R. Schapire, A short Introduction to Boosting, Journal of Japanese Society for Artificial Intelligence, 1999.
- [14] P. Wilson and J. Fernandez, Facial feature detection using haar classifiers, Journal of Computing Sciences in Colleges, 2006.
- [15] A. Nabatchian, E. Abdel-Raheem and M. Ahmadi, Human face recognition using different moment invariants: A comparative study, In Image and Signal Processing, 2008.
- [16] J. C. Russ, Image Processing Handbook, Fourth Edition, CRC Press, Inc., Boca Raton, FL, USA, 4th edition, 2002.
- [17] A. K. Agrawala and Y. N. Singhb, Evaluation of face recognition methods in unconstrained environments, Procedia Computer Science, International Conference on Intelligent Computing, Communication & Convergence, 2015.
- [18] I. T. Jolliffe, Principal Component Analysis, Springer Verlag, 1986.
- [19] P. N. Belhumeur, J. P. Hespanha and D. Kriegman, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection, Pattern Analysis and Machine Intelligence, IEEE Transactions, 1997.
- [20] M. Z. Hafed and M. D. Levine, Face recognition using the discrete cosine transform, Int. J. Comput. Vision, 2001.
- [21] I. J. Cox, J. Ghosn and P. N. Yianilos, Feature-based face recognition using mixture distance, In Computer Vision and Pattern Recognition, IEEE Computer Society Conference, 1996.
- [22] T. Ahonen, A. Hadid and M. Pietikainen, Face recognition with local binary patterns, In Tomás Pajdla and Jirí Matas, editors, Computer Vision, 2004.
- [23] T. Ahonen, A. Hadid and M. Pietikainen, Face description with local binary patterns: Application to face recognition, IEEE Trans. Pattern Anal. Mach. Intell, 2006.
- [24] L. Wiskott, J. M. Fellous, N. Krüger and C. Von Der Malsburg, Face recognition by elastic bunch graph matching, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997.
- [25] F. Samaria and S. Young, Hmm-based architecture for face identification, Image and Vision Computing, 1994.
- [26] S. Theodoridis and K. Koutroumbas, Pattern Recognition, Fourth Edition, Academic Press, 2008.
- [27] E. Vezzeti and F. Marcolin, Similarity Measures for Face Recognition, Bentham Science Publishers, 2015.
- [28] P. Parveen and B. Thuraisingham, Face recognition using multiple classifiers, 2006.
- [29] C. Cortes and V. Vapnik, Support-vector networks, 1995: Mach. Learn.

- [30] L. Lorente Giménez, Representación de caras mediante Eigenfaces, Ramas de estudiantes del IEEE, Universidad Politécnica de Catalunya, 1998.
- [31] J. E. Solem, Programming Computer Vision with Python, 2012.
- [32] Python, <https://www.python.org/>.
- [33] Python, <https://wiki.python.org/moin/Python2orPython3>.
- [34] Anaconda, Continuum Analytics, <https://www.continuum.io/>.
- [35] OpenCV, <http://opencv.org/>.
- [36] Numpy, www.numpy.org.
- [37] G. Bradski and A. Kaehler, Learning OpenCV, 2008: O'Reilly Media Inc.
- [38] ISO/IEC 19794-5:2011. Information technology - biometric data interchange formats - part 5: Face image data, In International Organization for Standardization, 2011.
- [39] The Database of Faces, <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>.
- [40] The normalized Yale Face Database, <http://vismod.media.mit.edu/vismod/classes/mas622-00/datasets/>.
- [41] AT&T Laboratories Cambridge, <http://www.cl.cam.ac.uk/research/dtg/attarchive/>.

GLOSARIO

AdaBoost: Adaptative Boosting	10
DCT: Discrete Cosine Transform	14
Detección: localizar todos los rostros y ojos que aparecen en una imagen sin identificarlos	9
DFT: Discrete Fourier Transform	14
DWT: Discrete Wavelet Transform	14
EBGM: Elastic Bunch Graph Matching	16
Eigenfaces: método utilizado para reconocimiento facial basado en técnicas de PCA	3
Espacio de las caras: zona reducida del espacio de las imágenes donde se encuentran las imágenes de las caras	19
Espacio de las imágenes: espacio de dimensión $w \cdot h$	19
Etapas de entrenamiento: primera etapa en la que el sistema se prepara con un conjunto de imágenes de entrenamiento	9
Etapas de test: etapa en la que se efectúa el reconocimiento con un conjunto de imágenes de test	9
FDA: Frequency Domain Analysis	14
Fisherfaces: método usado para reconocimiento facial basado en técnicas de LDA y PCA	14
Haar-like features: clasificadores en cascada que permiten la detección de objetos mediante el uso de la función matemática <i>wavelets de Haar</i>	10
HMM: Hidden Markov Models	16
ICA: Independent Component Analysis	13
Imágenes de entrenamiento: imágenes de los individuos con los que se entrenará el sistema	9
Imágenes de test: imágenes a las que se les aplicará el reconocimiento para identificar a los individuos con los que se entrenó previamente	9
ISO/IEC: International Organization Standardization / International Electrotechnical Commission	29
Jets: valores de los nodos de los grafos en el algoritmo EBGM	16
K-NN: K-Nearest Neighbours	18
KPCA: Kernel Principal Component Analysis	13
LBP: Local Binary Patterns	15

LDA: Linear Discriminant Analysis	14
PCA: Principal Component Analysis	3
Reconocimiento: identificar los rostros previamente detectados	9
Sistema de reconocimiento facial: aplicación dirigida por ordenador que identifica automáticamente a una persona analizando su rostro	2
SVD: Singular Value Descomposition	21
SVM: Support Vector Machines	18