

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/237270192>

# Analisis comparativo de metodos basados en subespacios aplicados al reconocimiento de caras

Article · January 2006

CITATIONS

4

READS

1,028

2 authors, including:



[Marcelo Armengot](#)

Complutense University of Madrid

5 PUBLICATIONS 6 CITATIONS

SEE PROFILE

# Análisis comparativo de métodos basados en subespacios aplicados al reconocimiento de caras



---

**Director:** Francesc J. Ferri  
ferri@uv.es

**Autor:** Marcelo J. Armengot Iborra  
marcelo.armengot@uv.es

Septiembre – 2006



*Nunca olvido una cara pero con la suya voy a hacer una excepción.*

Groucho Marx



## Síntesis preliminar

En el primer capítulo se introduce el problema del reconocimiento de caras humanas. Se presenta el contexto y se hace un breve recorrido en las técnicas más importantes recogidas en la literatura reciente. No sólo de reconocimiento, también se revisa el problema de la detección y ajuste de una cara dentro de la foto.

El segundo capítulo revisa conceptos teóricos que serán necesarios en el resto del documento y que conviene tener presentes.

El tercer capítulo presenta los tres métodos que han sido estudiados con más profundidad, introduciéndolos teóricamente para dejarlos preparados para su implementación.

En el cuarto capítulo se presentan una serie de experimentos realizados con dichos métodos y bases de datos estándar de imágenes extrayéndose conclusiones del presente trabajo e indicando posibles líneas para trabajos futuros.



# Índice general

Síntesis	v
Índice general	vii
Índice de figuras	xi
<b>1. Presentación</b>	<b>1</b>
1.1. Introducción . . . . .	1
1.2. Historia . . . . .	2
1.3. Quién es quién . . . . .	2
1.4. Detección de caras . . . . .	4
1.4.1. Técnicas basadas en rasgos . . . . .	4
Análisis de bajo nivel . . . . .	4
Análisis de rasgos . . . . .	5
Análisis de formas activas . . . . .	6
1.4.2. Técnicas basadas en la imagen . . . . .	7
1.5. Reconocimiento de caras . . . . .	7
1.5.1. Fotografía . . . . .	8
Métodos basados en la imagen . . . . .	8
Métodos basados en modelos . . . . .	9
1.5.2. Video . . . . .	12
Técnicas básicas . . . . .	13
<b>2. Introducción teórica</b>	<b>15</b>
2.1. Álgebra . . . . .	15
2.1.1. Espacios vectoriales . . . . .	15
Definición . . . . .	15



Base de un subespacio vectorial . . . . .	16
Subespacios suplementarios . . . . .	16
Espacio vectorial euclídeo . . . . .	16
Subespacio ortogonal . . . . .	17
2.1.2. Homomorfismos . . . . .	17
Definición . . . . .	17
Imagen de un homomorfismo . . . . .	17
Núcleo de un homomorfismo . . . . .	17
Matriz asociada a un homomorfismo . . . . .	18
Inversa de una matriz . . . . .	18
2.2. Estadística unidimensional . . . . .	18
2.2.1. Medidas de centralización . . . . .	18
2.2.2. Medidas de dispersión . . . . .	19
2.3. Estadísticos para datos multivariantes . . . . .	19
2.3.1. Medidas básicas de estadística multivariante . . . . .	19
2.3.2. Punto de vista geométrico . . . . .	21
2.3.3. Vectores y valores propios . . . . .	21
Asociados a un endomorfismo . . . . .	21
De una matriz . . . . .	22
Generalizados . . . . .	22
2.3.4. Gram-Schmidt . . . . .	23
Algoritmo . . . . .	23
2.4. PCA . . . . .	23
2.4.1. Introducción . . . . .	23
2.4.2. Planteamiento . . . . .	24
2.4.3. Generalización . . . . .	25
2.5. LDA de Fisher . . . . .	27
<b>3. Métodos</b>	<b>31</b>
3.1. Preliminar . . . . .	31
3.2. Eigenfaces . . . . .	31
3.2.1. Contexto . . . . .	31
3.2.2. Cálculo . . . . .	32
3.2.3. Algoritmo . . . . .	33

3.3. Fisherfaces . . . . .	33
3.3.1. Justificación . . . . .	33
3.3.2. Formulación . . . . .	34
3.3.3. Algoritmo . . . . .	35
3.4. Vectores comunes discriminantes . . . . .	35
3.4.1. Vector común . . . . .	35
3.4.2. Idea general . . . . .	37
Ejemplo en 3d . . . . .	38
3.4.3. Relación con PCA . . . . .	39
3.4.4. Cálculo de los vectores comunes discriminantes . . . . .	41
3.4.5. Sumario . . . . .	43
<b>4. Experimentos y resultados</b>	<b>45</b>
4.1. Base de datos de caras de AT&T . . . . .	45
4.2. Leaving one out . . . . .	45
4.2.1. Preparación . . . . .	45
4.2.2. Resultados . . . . .	46
4.2.3. Dimensión en el espacio de llegada . . . . .	47
Eigenface . . . . .	47
Fisherface y DCV . . . . .	49
4.3. Ruido . . . . .	49
4.3.1. Degradación de la imagen . . . . .	49
4.3.2. Resultados . . . . .	50
Sal y pimienta . . . . .	50
Gaussiana . . . . .	51
4.4. Conclusiones . . . . .	51
<b>Bibliografía</b>	<b>55</b>



# Índice de figuras

1.1. Detección . . . . .	6
1.2. Taxonomía básica de reconocimiento . . . . .	8
1.3. Elastic bunch-graph . . . . .	9
1.4. Esquema 3d . . . . .	10
1.5. Imágenes de rango . . . . .	12
2.1. Significado de la varianza . . . . .	22
2.2. Karl Pearson (1856-1936) . . . . .	24
2.3. PCA - descripción . . . . .	25
2.4. PCA - ejemplo . . . . .	26
2.5. PCA - ejemplo . . . . .	28
3.1. LDA, ejemplo en dos dimensiones. . . . .	34
3.2. Vector común . . . . .	37
3.3. DCV 3d . . . . .	38
3.4. Vector común 3d . . . . .	40
4.1. AT&T . . . . .	46
4.2. Leaving one out . . . . .	47
4.3. Reducción de dimensión en <i>Eigenface</i> . . . . .	48
4.4. Sal y pimienta . . . . .	49
4.5. Gausiano . . . . .	50
4.6. Resultados sal y pimienta . . . . .	50
4.7. Resultados de ruido gaussiano. . . . .	51
4.8. Fisherface vs. DCV (i) . . . . .	52
4.9. Fisherface vs. DCV (ii) . . . . .	53



# Capítulo 1

## Presentación del problema: Reconocimiento de caras

### 1.1. Introducción

Detectar y reconocer rostros humanos en fotografías y secuencias de video es un problema cada vez más en auge en el campo de la visión por computador. Algunas de sus soluciones entran en el campo de la minería de datos o en el reconocimiento de patrones, lo cual lo hace doblemente atractivo. Existen distintas aplicaciones prácticas:

- Vigilancia. Buscar a un individuo perseguido por la justicia. Localizarle en ámbitos públicos, aduanas, aeropuertos. Vigilancia doméstica, quien entra puede ser conocido o desconocido: detección de intrusos.
- Videoconferencia. Tiene que ver con localizar la imagen del individuo en una secuencia de *webcam* para poder hacer un seguimiento. También saber si está o no está, y quién es.
- Interfaces hombre-máquina. Seguridad de *log in* en una computadora.
- Detección de expresiones faciales (se considera un subcampo del reconocimiento de caras en general): interfaces inteligentes, detectores de cansancio para conductores, aplicaciones médicas.
- Control de acceso. Tarjetas de identificación, detección de fraudes (usurpar identidad), etc. Ejemplo: detectar *carne* falso.
- Base de datos multimedia. El incremento de imágenes en el *world wide web* sugiere cada vez más la existencia de aplicaciones CBIR de recuperación<sup>1</sup> de imágenes. En [8] se cita el caso de Informedia, un proyecto de recuperación de programas de televisión y documentales. En sistemas con bibliotecas digitales de terabytes de video y audio la clasificación de las imágenes cumple un papel fundamental.

---

<sup>1</sup>Habitualmente «CBIR systems», de Content-Based Image Retrieval, sistemas de recuperación de imágenes basados en contenidos.

El planteamiento de partida es tener un nivel de conocimiento previo de imágenes (etiquetadas) almacenadas en forma de base de datos, a partir del cual se quiere entrenar algún tipo de sistema de aprendizaje que permita resolver el problema concreto. Para valorar y comparar los algoritmos (de acuerdo con [2]) se consideran dos tipos según el escenario que se dé al aplicar el algoritmo y los objetivos que se tengan. Identificación de caras: *¿quién soy?* (test de uno a muchos), verificación o autenticación de caras *¿soy quien digo ser?* (test de uno a uno). Aunque también se trabaja con test de lista (*¿me estás buscando?*), el cual se considera un test de *universo abierto* ya que la cara buscada puede estar en la base de datos o puede no estar.

## 1.2. Historia

El primer método formal de clasificación de caras fue propuesto en 1888 por Francis Galton. Se recogían perfiles faciales como curvas, encontrando su norma, después se clasificaban perfiles por sus desviaciones con respecto a la norma. El resultado era un vector que podía ser comparado con otros vectores de la base de datos.

En los años 1960 se introdujeron sistemas semiautomáticos que hacían marcas en las fotografías para localizar los rasgos principales: ojos, orejas, nariz y boca. Las distancias y radios se calculaban a través de las marcas para construir un sistema de referencia y poder comparar los datos. A principios de los 70, Goldstein, Harmon y Lesk ([6]) crearon un sistema con 21 marcadores que incluían color del pelo y grosor de labios. Sus pruebas eran también difíciles de automatizar porque muchas de estas medidas se tomaban a mano.

Pocos años después Fisher y Elschlagerb introdujeron un sistema más automático que utilizaba plantillas para medir los rasgos de diferentes partes de la cara, con esas medidas se construía un mapa global. Tras una continuada investigación resultó que estas medidas no contenían suficientes datos únicos como para representar una cara de adulto.

Otra aproximación intenta clasificar la cara humana usando una combinación de la gama de gestos como un juego de marcadores a identificar. En general se pone en práctica usando reconocimiento de modelo 2d y principios de redes neuronales. La mayoría de veces esta técnica requiere un enorme número de caras a entrenar para alcanzar una exactitud respetable.

El primer método completamente automatizado comparaba las caras en un modelo genérico de rasgos esperados y creaba una serie de patrones para una imagen. Esta estrategia, era principalmente estadística, se basaba en histogramas y en el valor de escala de gris. Por otro lado, a partir de 1988 los trabajos –primero [16]– de Kirby y Sirovich (Brown University) y –después [17]– de Turk y Pentland (MIT) basados en el método *Eigenface* abrieron un camino al que ha ido contribuyendo mucha gente.

Desde 1990, al interés por el reconocimiento de caras automático se han sumado un creciente número de grupos de investigación. Apoyándose también en las mejoras técnicas que han supuesto el desarrollo de las redes neuronales, análisis de *wavelet*, infografía y visión por computador.

## 1.3. Quién es quién

Aunque las aplicaciones pueden ser todas las imaginables y están en función del desarrollo mismo de las técnicas y la infalibilidad que éstas alcancen, la primera (vigilancia) y todos sus derivados constituyen la razón de ser del auge que en los últimos años ha tenido este campo. Sirva

a modo de síntesis de la situación actual esta colección de extractos de [www.frvt.org](http://www.frvt.org).

El Departamento de Defensa (DoD) de los EEUU a través de la CTDPO (Oficina para el Programa de Desarrollo Tecnológico Antidroga) financia el plan **FERET**. Cuyo objetivo es «asistir al personal de las fuerzas de la ley, seguridad e inteligencia» en sus asuntos. El FERET se dedica a financiar investigación, construir la base de datos FERET y llevar adelante sus evaluaciones. Las evaluaciones que ha ido introduciendo a la comunidad dedicada a la detección de caras han «ayudado a su avance desde la infancia hasta la etapa de prototipos».

Compañía	Sede
A4Vision	USA
Animetrics Inc.	USA
Cognitec Systems GmbH	Alemania
Cross Match Technologies, Inc.	USA
Cybula Ltd.	Inglaterra
DreamMirh Co., Ltd.	Corea del Sur
Geometrix Inc.	USA
ID One, Inc.	USA
Identix Inc.	USA
JAD Communication & Security	USA
Neurotechnologija.	Lituania <sup>2</sup>
Neven Vision Inc.	USA
OmniPerception Ltd.	Inglaterra
Sensible Vision, Inc.	USA
TAB Systems	Slovenia
Takumi Vision Technologies, Inc.	Japón
The Covenant Consortium (TCC)	USA
Viisage Technology	USA
x-pin.com GmbH	Alemania

Cuadro 1.1: Principales compañías participantes del test FRVT para evaluar técnicas de reconocimiento de caras.

El **FRVT** es el «Face Recognition Vendor Test», el test que sirve como referencia para comparar técnicas y algoritmos. Como su propio nombre indica es el test que manejan las compañías comerciales entre cuyas actividades y servicios se incluyen aplicaciones de reconocimiento de caras, el cuadro 1.1 muestra algunas de las principales<sup>3</sup>. Sin embargo, tres de las cinco evaluaciones usadas por FRVT actualmente han sido introducidas por el plan FERET. Hacia el año 2000 «la tecnología de reconocimiento de caras ha madurado desde sistemas de prototipo hacia sistemas comerciales». Ya desde hacía años, pero particularmente desde entonces, tanto el FERET como el FRVT parecen haber sido determinantes a la hora de marcar los lindes del camino.

El **NIST** es el «National Institute of Standards and Technology» de EEUU. Entre sus diversas actividades dirige el FRVT 2006 (y ha dirigido el FRVT con anterioridad), que es el plan de evaluación que FRVT ha puesto en marcha actualmente. El FRVT 2006 no está diseñado actualmente para ser un certificado<sup>4</sup> «*NIST Patriot Act*» pero sus resultados serán utilizados en los diseños futuros de certificados «*NIST Patriot Act*». Varias agencias del gobierno de los EEUU (entre ellas el FBI) participan actualmente en la cofinanciación del FRVT 2006.

<sup>2</sup>Es una compañía lituana aunque fue fundada en 1990 cuando Lituania pertenecía a la antigua Unión Soviética.

<sup>3</sup>Extraído de <http://www.face-rec.org> el 23 de septiembre de 2006.

<sup>4</sup>Se podría traducir el original NIST Patriot Act certificado como «certificado patriótico del NIST».



## 1.4. Detección de caras

Como un paso previo al reconocimiento de caras, conviene tener algo de conciencia sobre las técnicas que suelen emplearse para su detección, es decir su localización dentro de una imagen, ambos casos ponen en práctica algunos métodos y conceptos comunes. Además, la mayoría de las aplicaciones del reconocimiento de caras tienen sentido si esta fase está «resuelta» adecuadamente.

### 1.4.1. Técnicas basadas en rasgos

En este tipo de técnicas el conocimiento previo se hace completamente explícito y se sigue una metodología clásica de detección en la que los rasgos de más bajo nivel se extraen de un análisis a priori basado en el conocimiento. En diferentes niveles del sistema se explotan propiedades aparentes de la cara tal como el color de la piel y la geometría facial. En estas técnicas la detección de la cara se resuelve manipulando medidas de distancia, ángulos y áreas de los rasgos visuales en la escena. La cuestión determinante en este tipo de métodos es decidir qué rasgos de la cara o de la imagen interesan para su estudio.

#### Análisis de bajo nivel

**Bordes:** Uno de los rasgos más primitivos que tiene cualquier figura es su contorno. Los trabajos que utilizaban esta idea extraían bordes de la cara tanto externos como internos. Luego eran sometidos a análisis de forma y posición. También se puede utilizar estos métodos para detectar si el individuo lleva gafas. Una vez extraídos los bordes pueden usarse diferentes operadores. Existen técnicas más recientes que utilizan estas estrategias, en las que se alcanzan tasas de acierto del 76 % con dos falsos positivos por imagen.

**Niveles de gris:** Los rasgos faciales como las cejas, pupilas y los labios aparecen generalmente más oscurecidos que las regiones de su alrededor. Esta propiedad puede ser explotada para diferenciar partes. Algunos algoritmos de extracción de rasgos faciales buscan mínimos locales dentro de regiones faciales segmentadas. En dichos algoritmos a las imágenes se les realiza previamente el contraste y se les aplica rutinas morfológicas de escalas de grises para mejorar la calidad de las regiones oscuras y facilitar su detección. La posición relativa de los ojos y su detección también puede ser descubierta con este tipo de métodos.

**Color:** Si el blanco y negro permite una representación básica de la imagen para extraer propiedades, el color es una herramienta mucho más poderosa, entre otras cosas porque al triplicarse la dimensionalidad se tiene más información. Dos formas que en una imagen de blanco y negro aparecen iguales pueden ser diferentes en una imagen con color. El color de la piel humana (en sus diferentes variantes) ha permitido desarrollar algunas técnicas que detectan la raza, en individuos varía mucho menos. Como una de las maneras más habituales de trabajar con fotografía digital es la representación RGB<sup>5</sup>, en este tipo de técnicas se suele trabajar con un RGB normalizado que vendría dado por las ecuaciones:

---

<sup>5</sup>Red-Green-Blue.

$$\begin{aligned}
 r &= \frac{R}{R + G + B} \\
 g &= \frac{G}{R + G + B} \\
 b &= \frac{B}{R + G + B}
 \end{aligned}
 \tag{1.1}$$

A partir de las cuales siempre ocurrirá  $r + g + b = 1$ . Pudiéndose averiguar cualquiera de los tres componentes conociendo los otros, la componente azul por ejemplo será:  $b = 1 - r - g$ . Al analizar el color de la piel, un histograma basado en  $r$  y  $g$  muestra el color de la cara como una pequeña región. Al comparar luego un píxel con dicha región se puede averiguar si dicho píxel pertenece o no a la cara.

Relacionado con la detección de razas, se utiliza también la representación YIQ en la que el componente I representa un rango de color desde el naranja hasta el cyan, lo que permite realzar la imagen para el caso de individuos asiáticos.

En este tipo de trabajos y aprovechando la representación  $rgb$  (normalizada) se pueden encontrar también técnicas que detecten labios, ojos, cejas, etc para poder usar esquemas de segmentación.

**Video:** Si se dispone de una secuencia de video, la localización de objetos en la imagen es más factible. Una de las mejores formas es mediante diferencia de fotogramas. Existen técnicas que miden variaciones verticales y horizontales para encontrar los ojos. Detectar contornos de una escena en movimiento es más sencillo. Se utilizan filtros espaciotemporales de gaussiana para encontrar los bordes de la cara y el cuerpo. También existen técnicas que trabajan con la velocidad a la que se mueven ciertas regiones de la escena. En estas técnicas se completa la localización de la cara con un algoritmo que construye una elipse ajustándola.

**Medidas generalizadas:** Como la forma de la cara es simétrica se utilizan medidas de simetría de la imagen. Medidas del gradiente en una imagen frontal. Existen trabajos en los que el entrenamiento se hace con imágenes en distintas orientaciones y fondos, en las que se consigue detectar la boca y los ojos. En otros trabajos se parte de la concavidad y convexidad de las regiones (pómulos, ojos) para utilizar la derivada de la medida del gradiente.

## Análisis de rasgos

El problema de los métodos anteriores es que sus medidas son demasiado ambiguas. Detectar la cara sobre la base del color permite que se «cuelen» como falsas caras cualesquiera objetos del fondo. Utilizar el conocimiento que ya se tiene de la geometría facial puede servir para caracterizar y verificar muchos rasgos a priori confusos. Existen básicamente dos estrategias:

**Búsqueda de rasgos:** Estas técnicas buscan rasgos prominentes que permiten localizar rasgos menos prominentes partiendo de hipótesis geométricas. Por ejemplo una pequeña área sobre un área alargada puede corresponderse al escenario «*cabeza sobre los hombros*» y un par de regiones oscuras encontradas en el área facial incrementa la probabilidad de que aquello sea realmente una cara. Los rasgos más usados son los ojos, contorno de la cabeza y el cuerpo (bajo la cabeza).

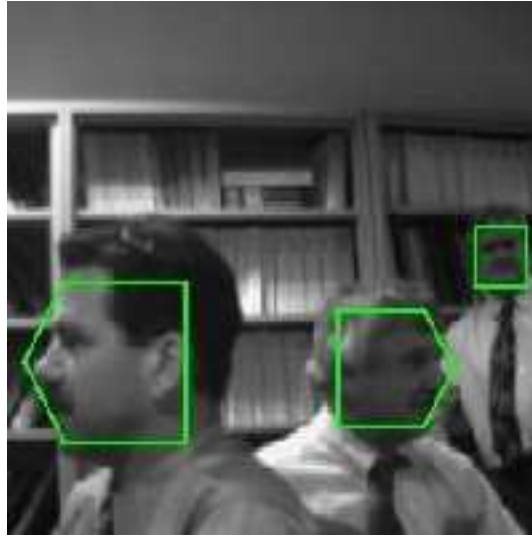


Figura 1.1: La detección de caras dentro de una imagen es un paso previo para su reconocimiento.

**Análisis de constelación:** La mayoría de los algoritmos que se basan en búsqueda de rasgos dependen fuertemente de su heurística. Aplicados a casos más generales (diferentes posturas, primer plano sobre fondos complejos) pueden volverse demasiado rígidos. Existen otros métodos posteriores a aquellos que agrupan rasgos faciales en constelaciones<sup>6</sup> «*facial-like*» y utilizan métodos más robustos de modelado como análisis estadístico.

### Análisis de formas activas

Se basan en representar la imagen en rasgos de alto nivel para posteriormente interactuar con rasgos locales de la imagen (ojos, brillo) y gradualmente deformarla hasta adaptarla a la forma de los rasgos.

**Serpientes:** Se utilizan las «*serpientes*» o contornos activos comunmente para localizar el contorno de la cabeza. Se inicializa la serpiente ante una cara inminente. Se va cerrando a los contornos de la imagen asumiendo la forma de la cabeza.

**Plantillas deformables:** Localizar una cara partiendo de contornos genéricos puede no ser tan sencillo. Existen técnicas que aprovechan los cambios de contraste. Algunas de ellas aprovechan la idea de las «*serpientes*» y utilizan una plantilla parametrizada.

**Modelos de distribuciones de puntos:** Se trata de una descripción compacta parametrizada de las formas de la cara basada en estadísticas. La arquitectura y ajuste de este método es diferente a los anteriores. El contorno del PDM se discretiza en un conjunto etiquetado de puntos. Las variaciones de dichos puntos son primero entrenadas con diferentes posturas y tamaños. Durante el proceso de deformación sólo se permite variar a la *forma* en una de las posibilidades del conjunto de entrenamiento.

---

<sup>6</sup> *Facial-like constellations, constelaciones parecidas a una cara.*

### 1.4.2. Técnicas basadas en la imagen

En estas técnicas, por el contrario, el objeto de estudio es la imagen misma. El conocimiento previo se incorpora implícitamente en esquemas de entrenamiento. Se trabaja directamente con una representación de la imagen a la que se le aplican algoritmos de entrenamiento y análisis.

#### Métodos basados en subespacios:

Se basan en los trabajos de Sirovich y Kirby y los desarrollos posteriores de Turk y Pentland. Consideran las imágenes de caras humanas como un subespacio lineal de un espacio mayor (de todas las imágenes). Al representarlas así pueden utilizarse muchos métodos para tratar los datos: análisis estadístico multivariante, redes neuronales, etc.

#### Redes neuronales:

Las redes neuronales se han convertido en una técnica popular para el reconocimiento de patrones. Ciertas técnicas basadas en redes neuronales han utilizado ventanas de  $20 \times 20$  píxels que reciben como entrada una imagen preprocesada. Se entrenan las redes neuronales con imágenes con caras e imágenes sin caras. Se resuelven problemas de solapamiento de regiones ya que estas técnicas son usadas para verificar si una foto es de una cara o no<sup>7</sup>. También se han utilizado este tipo de técnicas para detección de rasgos.

#### Métodos estadísticos:

Hacen referencia a métodos basados en teoría de la información. También se incluyen aquí técnicas que utilizan máquinas de vectores soporte y reglas de decisión de Bayes. La figura 1.1 muestra un ejemplo de las técnicas estadísticas empleadas por Henry Schneiderman y Takeo Kanade que les ha permitido en 2004 ofrecer varias soluciones comerciales de su tecnología, que utiliza modelado estadístico para capturar la variación que crea la apariencia facial.

## 1.5. Reconocimiento de caras

Al igual que ocurre con la detección, el reconocimiento de caras es un problema que puede resolverse (o no) de muchas maneras. Los trabajos que se han hecho al respecto han propuesto todo tipo de métodos, algoritmos y estrategias, así que para tener una visión de conjunto en esta sección se ha tomado sobretodo los informes [1] de W. Zhao y [2] de X. Lu, que vienen con la recomendación de [www.face-rec.org](http://www.face-rec.org).

La figura 1.2 puede servir de orientación en el universo del reconocimiento de caras. Lo primero que separa el tipo de técnicas es si se utiliza como fuente el video o la foto. Las propiedades del video permiten ciertas consideraciones relacionadas con el movimiento que en la foto no existen. Dentro de las técnicas basadas en fotografía, ocurre algo parecido a lo que se ha descrito sobre la detección de caras, se puede considerar la fotografía como base de trabajo y como objeto de trabajo, o por el contrario puede intentar construirse partiendo de ella distintos modelos que representen la cara y puedan responder ante métodos de comparación y búsqueda. Separadas estas cuestiones previas puede profundizarse más en sus propias variantes. Éstas harán referencia al tipo de estrategias, teoría y algoritmos que se utilicen en cada caso.

---

<sup>7</sup>No es lo mismo saber si una foto es de una cara que saber si en esa foto hay alguna cara.

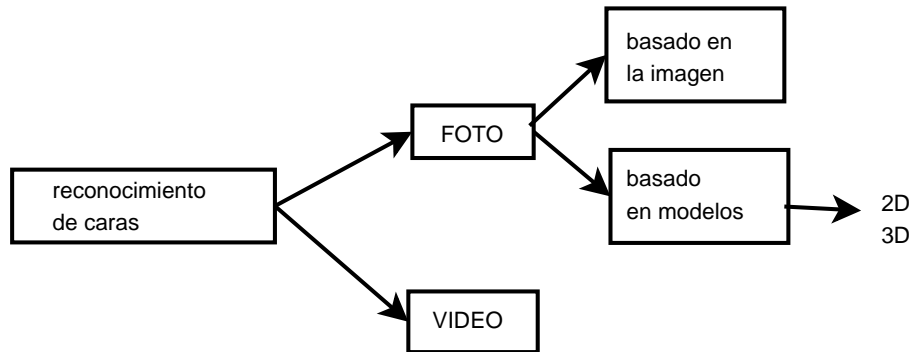


Figura 1.2: Clasificación de métodos para reconocimiento de caras.

### 1.5.1. Fotografía

#### Métodos basados en la imagen

Existen dos alternativas fundamentalmente a la hora de plantearse métodos de reconocimiento basados en la imagen. O bien diseñar un clasificador suficientemente potente (com por ejemplo algunos modelos de redes neuronales) sobre el espacio de todas las caras. O bien aplicar métodos basados en subespacios u otras técnicas de reducción de dimensionalidad para poder aplicar otros métodos de clasificación en mas bajas dimensionalidades. La segunda posibilidad será tratada en el capítulo 3.

#### ■ Redes neuronales

Una red típica para reconocimiento de imagen requiere  $N = m \times n$  neuronas de entrada, una por cada píxel en una imagen de  $m \times n$ . El mapa correspondiente se traza en un número  $p$  de neuronas en capas ocultas. Éstas van por turnos trazando el mapa de las neuronas de salida, para que al menos una dispare la correspondiente cara en la base de datos. Ocurre que  $p$  puede ser mucho más pequeño que  $N$ . La capa oculta se considera un vector de características que expresa los rasgos faciales de forma concentrada.

El entrenamiento de redes neuronales con un pequeño conjunto de imágenes permite registrar un almacenamiento ajustado incluso cuando las imágenes son muy ruidosas o alguna parte está oculta. En [9] se supone una red neuronal perceptrón multicapa para construir un dispositivo de control de acceso basado en reconocimiento facial. Se entrena la red neuronal con personas autorizadas y con personas no autorizadas estudiándose tanto las falsas aceptaciones como los falsos rechazos. El objetivo de este tipo de aplicaciones es funcionar en tiempo real.

Más recientemente Prasanna, Sudha y Kamakoti han descrito en [10] una red neuronal que utiliza el método PCA, a la que llaman PCNN<sup>8</sup>. Su sistema tolera variaciones locales en la cara como cambios en la expresión y dirección de la luz. En este caso, los autores han diseñado un hardware cuya implementación en ASIC<sup>9</sup> es capaz de procesar 11000 entradas por segundo en el proceso de aprendizaje y 19000 en la fase de recuperación. Lo cual es –según

<sup>8</sup>NN de neural network, red neuronal.

<sup>9</sup>Application-specific integrated circuit, circuito integrado de propósito específico.

sus autores— 10 veces superior y 5 veces más rápido que un sistema software (funcionando en un PC).

#### ■ Otras estrategias

Dentro de los métodos que funcionan con una imagen 2d de la cara existen algunas variantes. En [1] se reseñan varios trabajos interesantes. Existen alternativas que trabajan con fotos tomadas en *infrarrojos* cuyo interés es escapar a las variaciones de la luz. Los cambios en la iluminación pueden degradar significativamente el reconocimiento. De hecho, muchos de los métodos basados en fotografías normales utilizan diferentes técnicas para escapar de este problema conocido. Las imágenes en infrarrojos pueden servir de alternativa, aunque tienen otros problemas derivados de su resolución.

Otros trabajos destinados a aplicaciones más concretas de investigación policial han trabajado con bocetos. Basándose en la descripción de un testimonio un dibujante hace un boceto de la cara. La base de datos de fotos reales de caras puede contener miles de fotos. Construir un sistema capaz de reconocer automáticamente las fotos partiendo del boceto puede tener un enorme valor práctico.

### Métodos basados en modelos

#### ■ 2D

##### Grafos elásticos de racimo<sup>10</sup>

Todas las caras humanas comparten una estructura topológica similar. Para representar la cara como un grafo hay que seleccionar los puntos testimoniales o *fiduciaros* (ojos, extremo de la nariz, algunos puntos del contorno).

El reconocimiento de caras se basa en grafos etiquetados. Un grafo etiquetado es un conjunto de nodos conectados por arcos, los nodos están etiquetados con muelles<sup>11</sup>, los arcos están etiquetados con distancias. La geometría de un objeto queda codificada por los arcos mientras que la distribución de grises se codifica por los nodos.

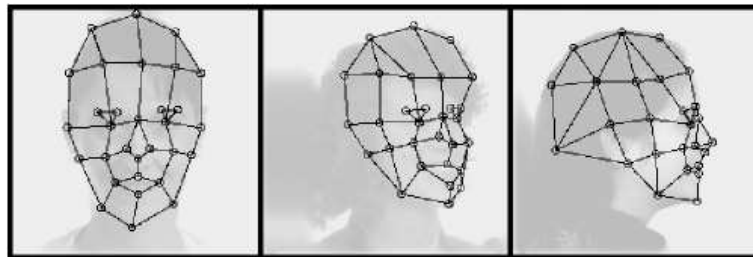


Figura 1.3: Reconocimiento basado en *elastic bunch-graph* (gráficos elásticos de racimo).

Para identificar una nueva cara, el grafo de la cara se posiciona en la imagen usando una correspondencia elástica. El objetivo de la correspondencia elástica es encontrar en la imagen consultada los puntos fiduciaros que permiten maximizar la función de similitud del grafo. Si el grafo se inicializa adecuadamente este proceso es automático.

<sup>10</sup>En el original *bunch graph*, grafo de manojo.

<sup>11</sup>En el original *jets*: «Representación concentrada y robusta de la distribución local de valores de gris».

Un grafo de este tipo aplicado a una cara puede concebirse como una colección de modelos individuales de caras combinados en una estructura de pila. Para posicionar la *rejilla* en una nueva cara, se intenta maximizar la similitud entre el grafo de la base de datos y el grafo de la imagen. La función similitud se define como la media de las mejores posibles correspondencias entre la nueva imagen y cualesquiera de las almacenadas previamente.

### Modelo de Apariencia Activa

Un AAM es un modelo estadístico integrado que combina un modelo de la variación de la forma con un modelo de variaciones de la apariencia en un fotograma normalizado en escala de grises. Corresponder a una imagen supone encontrar los parámetros del modelo que minimizan la diferencia entre dicha imagen y la síntesis del modelo, proyectado en la imagen. La dificultad reside en que el número de parámetros puede ser potencialmente grande.

Para construir el AAM hace falta un conjunto de entrenamiento con imágenes etiquetadas con los puntos significativos marcados en cada ejemplo en posiciones clave para los rasgos característicos. Todos los vectores de caras se normalizan en un sistema de coordenadas común. Se aplica PCA para construir el modelo de la forma. En la fase de reconocimiento se hace el proceso inverso.

### ■ 3d

El reconocimiento de caras humanas basado en correspondencias con superficies 3d puede resolver muchas de las limitaciones actuales del reconocimiento 2d. La forma 3d es invariante a los cambios de posición e intensidad de luz aunque no ante los cambios de expresión facial. Por otra parte, recoger y almacenar plantillas de cada sujeto en una base de datos (asociadas a varias expresiones), no es trivial<sup>12</sup>.

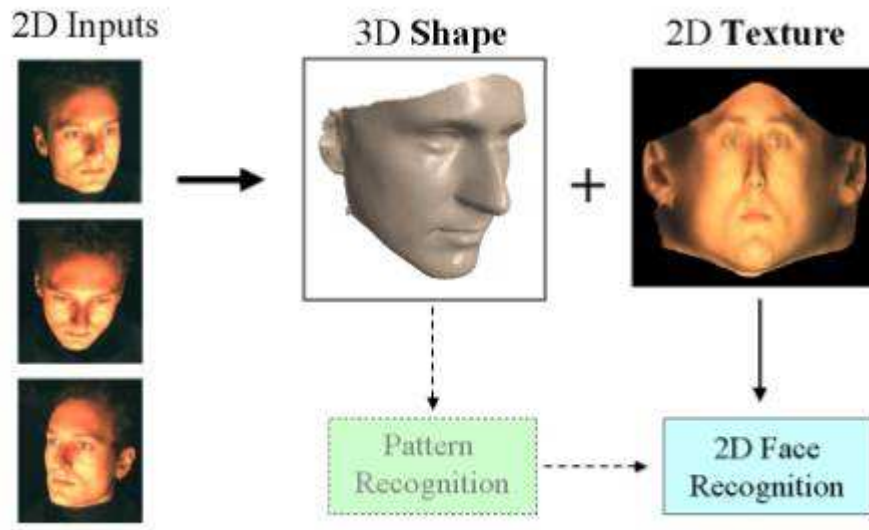


Figura 1.4: Esquema habitual 3d deformable, tomado de Mitsubishi Electric Research Labs.

<sup>12</sup>Existen varias técnicas, entre ellas el uso de dispositivos láser de *scanner*.

**Modelo deformable 3d:**

La primera y única técnica reseñada en [2] es relativamente reciente y se corresponde con un esquema bastante generalizado. El planteamiento es reconocer una imagen nueva (en fotografía) partiendo de una base de datos con formas 3d. El método de cara 3d deformable<sup>13</sup> codifica la forma y la textura en un modelo de parámetros y con un algoritmo se recogen dichos parámetros en un caso de foto simple. Para la identificación se usan los parámetros de textura y forma del modelo separados de los parámetros de la imagen, como la luz y la posición. Para manejar las variaciones se suelen usar modelos generativos de imagen. Para el análisis de imagen, la estrategia general de todas las técnicas es ajustar el modelo generativo a una imagen nueva, que se tiene que parametrizar en los términos del modelo. Para hacer la identificación independiente de las condiciones de la imagen el objetivo es separar los parámetros intrínsecos del modelo de los parámetros extrínsecos de la imagen de la cara. Esa separación se registra explícitamente utilizando un proceso de formación de imagen utilizando tecnología gráfica 3d.

**Correspondencia 2.5d:**

Además de los *scanners* láser, existen otros recursos aplicables a este contexto. Es el caso de las cámaras de rango<sup>14</sup>: capaces de tomar **imágenes de rango**. Podemos entender una imagen de rango como una imagen digital donde cada píxel contiene un valor equivalente a la distancia de dicho objeto con la cámara. Estas cámaras pueden tomar ese tipo de fotos por medio de un objeto de referencia que permite el calibrado. Las imágenes de la izquierda en la figura 1.5 están tomadas con una cámara de rango, normalizadas y representadas como una imagen normal en la que el negro se asigna a puntos cercanos y el blanco a puntos lejanos. Las imágenes de la derecha son imágenes de intensidad (normales) con el foco de luz colocado cerca de la cámara (por eso parecen negativos) pero en entornos no controlados la luz puede estar distribuida de muy diferentes maneras. Las fotos tomadas con este tipo de cámaras también se llaman fotos en 2.5d (dos dimensiones y media).

El método de correspondencia 2.5d presentado en [3] genera superficies 3d en una representación  $(x, y, z)$  partiendo de varias imágenes 2.5d tomadas para cada individuo. La técnica de reconocimiento consiste en dos componentes, una correspondencia de superficie y una correspondencia basada en la apariencia. La parte de correspondencia con la superficie se apoya en el algoritmo del punto más cercano<sup>15</sup> (ICP). El candidato de la base de datos usado para verificar correspondencia de la apariencia es generado dinámicamente basándose en la salida del componente de superficie, que reduce la complejidad de la fase de correspondencia basada en la apariencia. Los modelos tridimensionales de la base de datos se utilizan para sintetizar muestras de apariencia con las variaciones de posición y luz y se utilizan dichas imágenes sintetizadas para hacer el análisis con la foto de prueba.

**Análisis de curvatura:**

En [4] se describe otra posibilidad que consiste en buscar los rasgos salientes (ojos y nariz) y someterlos a un análisis de curvatura de superficies. Cada triplete se considera un candidato para encontrar nariz y ojos. Este método también parte de disponer las imágenes de la base de datos en *scanner* de láser y la foto nueva en imagen de rango. El método funciona para imágenes con y sin cara, que se procesan y se comparan utilizando *Eigenface* para comparar con los caras tridimensionales de la base de datos.

<sup>13</sup>En su primera aparición en el original «3d morphable model», posteriormente se puede encontrar «3d deformable model» en otros trabajos.

<sup>14</sup>Range cameras

<sup>15</sup>Iterative Closest Point



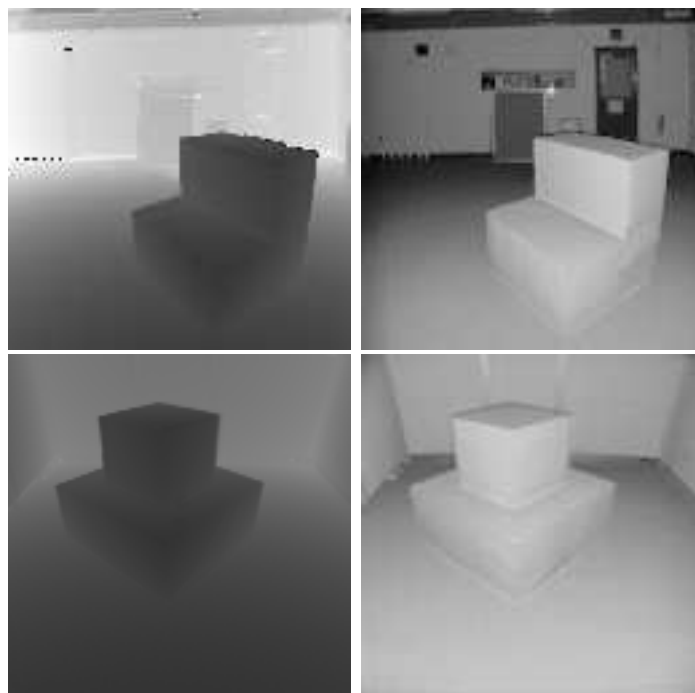


Figura 1.5: A la izquierda imágenes de rango, a la derecha fotos normales.

### 1.5.2. Video

Una de las aplicaciones más importantes del reconocimiento de cara es en sistemas de video-vigilancia. Después de muchos años de estancamiento (hasta finales de los noventa), durante la última década la investigación en el reconocimiento de acción y comportamiento a partir de video está activándose: Detectores de cansancio para conductores, sistemas de control doméstico, etc. Sin embargo, reconocer sujetos para video-vigilancia conlleva algunas dificultades:

1. Baja calidad de video. Cámaras situadas en sitios con luz natural y la no «colaboración» de los sujetos, oclusión parcial, etc.
2. La imagen de la cara una vez tomado el video queda muy pequeña. Por las mismas condiciones de adquisición de los datos la cara puede llegar a ser diminuta y no se alcanzan los tamaños asumidos en la mayoría de sistemas de reconocimiento de caras.
3. Características propias humanas. Una de las principales razones de la viabilidad de una descripción genérica de comportamiento humano es que la variación intrínseca de *objetos* humanos y en particular faciales es mucho más pequeña que la variación con respecto a los objetos contextuales. Por ejemplo, es mucho más fácil detectar y localizar caras que reconocer una cara específica.

### Técnicas básicas de procesamiento de caras basado en video

La literatura al respecto parece ponerse de acuerdo en cuatro áreas acotadas para tratar este problema:

1. **Segmentación de objetos en video:** Los intentos más sencillos como basarse en cualquier movimiento de los píxeles puede no funcionar en condiciones adversas: por ejemplo si hay múltiples objetos en movimiento y existe oclusión. Un buen grueso de la literatura se dedica al análisis de segmentación y detección de objetos en movimiento en video obtenido de plataformas estáticas o móviles. Hay métodos basados en el análisis de diferentes imágenes, discontinuidades en campos de flujo utilizando segmentación, procesos de línea o modelos basados en cadenas de Markov. Algunas de estas técnicas se han extendido a situaciones con movimiento (de la cámara o de los objetos). Combinar segmentación y detección en video puede ser una buena aproximación a la detección de caras individuales. El color de la piel puede ser utilizado para fortalecer ciertos algoritmos de detección de caras.
2. **Estructura del movimiento:** El problema de la estructura del movimiento es estimar la profundidad 3d de los puntos en una secuencia de imágenes. A menos que la cámara pueda moverse a lo largo de una base conocida las técnicas típicas *stereo* quedan excluidas. Se puede calcular algún tipo de campo de flujo y usarlo para estimar la profundidad de los puntos visibles, pero esto es computacionalmente costoso. En la aproximación discreta se calculan sólo una serie de puntos, bordes, esquinas, líneas o contornos extraídos de la secuencia de video. Tanto en la aproximación diferencial como en la discreta los parámetros que caracterizan el movimiento aparecen conjuntamente con los parámetros de la profundidad. La información de profundidad es útil para construir modelos 3d para objetos y usar esos modelos para reconocer objetos presentes u ocluidos. En visión monocular dicha información es relativa a un factor, si se dispone de visión binocular (o multi-cámara) se pueden triangular valores absolutos. Existen (y se han descrito) infinidad de técnicas para todo este tipo de casuística.
3. **Modelos 3d para caras:** Son conocidos los modelos 3d para reconocimiento de caras en aplicaciones de reconstrucción facial forense, información parcial y envejecimiento infográfico. En la sección 1.5.1 se han reseñado algunas alternativas más recientes para reconocimiento. Aplicar alguna de ellas a video supone aplicar técnicas de tiempo real de modelado 3d y rastreo de caras. Ciertas técnicas utilizan un modelo 3d genérico de cabeza para alinear a la frontal de la vista en una secuencia de video.
4. **Análisis de movimiento no rígido:** Se han aplicado a problemas de reconocimiento algunas de las técnicas de detección de movimiento no rígido. Este tipo de técnicas tienen que ver con la observación de objetos no rígidos en movimiento y su detección. Sus aplicaciones van desde trabajos relacionados con el movimiento de glaciares hasta la detección de objetos articulados.



## Capítulo 2

# Introducción teórica

### 2.1. Álgebra

Se revisa en esta sección algunos elementos de álgebra que antes o después se han tenido que repasar. En su mayoría se ha tomado esta sección de [14], a veces literalmente y a veces sintéticamente. Para simplificar, se ha dado por sentado en todas las definiciones que el cuerpo sobre el que se describen todas las estructuras algebraicas es  $\mathbb{R}$ .

#### 2.1.1. Espacios vectoriales

##### Definición

Teniendo en  $\mathbb{R}^n$  el conjunto de los vectores de  $n$  dimensiones se entiende que la combinación  $(\mathbb{R}^n, +, \cdot)$  es un espacio vectorial al existir dos tipos de operaciones (suma y producto por un *escalar*) que verifican una serie de propiedades. A saber:

- **Suma:**

- Ley de composición interna.
- Asociativa
- Elemento neutro
- Elemento opuesto
- Conmutativa

- **Producto por un escalar:**

- Ley de composición externa
- $\forall \alpha \in \mathbb{R}, \forall x, y \in \mathbb{R}^n \implies \alpha(x + y) = \alpha x + \alpha y$
- $\forall \alpha, \beta \in \mathbb{R}, \forall x \in \mathbb{R}^n \implies (\alpha + \beta)x = \alpha x + \beta x$
- $\forall \alpha, \beta \in \mathbb{R}, \forall x \in \mathbb{R}^n \implies \alpha(\beta x) = (\alpha\beta)x$
- $\forall x \in \mathbb{R}^n \implies 1 \cdot x = x$

Una parte,  $U$ , de un espacio vectorial  $V$  es un subespacio vectorial cuando cumple:

$$\forall u, v \in U, \forall \lambda, \mu \implies \lambda u + \mu v \in U \quad (2.1)$$

### Base de un subespacio vectorial

El conjunto de vectores  $u_1, u_2 \dots u_n$  es una base del espacio vectorial  $U$  definido en  $\mathbb{R}$  si es un sistema generador de  $U$ . Es decir cumple:

$$\forall x \in U \implies x = \sum_{i=1}^n \alpha_i u_i \text{ con } \alpha_i \in \mathbb{R} \quad (2.2)$$

Y si es un sistema libre, es decir que no es un sistema ligado, ya que de cualquier combinación lineal de ellos igualada a cero se deduce que todos los coeficientes han de ser cero.

### Subespacios suplementarios

Se dice que  $U_1$  y  $U_2$  son subespacios suplementarios de  $V$  si y sólo si cumplen:

$$\begin{aligned} U_1 + U_2 &= V \\ U_1 \cap U_2 &= 0 \end{aligned} \quad (2.3)$$

También puede escribirse como:

$$V = U_1 \oplus U_2 \quad (2.4)$$

### Espacio vectorial euclídeo

Se dice que a un espacio vectorial se le ha dotado de estructura de espacio vectorial euclídeo cuando se considere en él un producto escalar. Un espacio vectorial euclídeo es un par  $(V, \langle, \rangle)$  donde  $V$  es un espacio vectorial y  $\langle, \rangle : V \times V \rightarrow \mathbb{R}$  es un producto escalar, definido en  $V$ .

Un **producto escalar** sobre  $V$  es aquella aplicación que a cada par de vectores  $x, y \in V$  asocia un número real, llamado producto escalar de  $x$  por  $y$ , habitualmente representado por  $\langle x, y \rangle$ . Sus propiedades son:

- |  |  |
|--|--|
| 1. $\langle x, y \rangle = \langle y, x \rangle$                                 | $\forall x, y \in V$                                 |
| 2. $\langle x, y + z \rangle \equiv \langle x, y \rangle + \langle x, z \rangle$ | $\forall x, y, z \in V$                              |
| 3. $\langle \lambda x, y \rangle = \lambda \langle x, y \rangle$                 | $\forall x, y \in V, \forall \lambda \in \mathbb{R}$ |
| 4. $\langle x, x \rangle \geq 0$   | $\forall x \in V$                                    |
| 5. $\langle x, x \rangle = 0 \iff x = 0$   |  |

### Subespacio ortogonal

Dado un subconjunto  $C$  de vectores de un espacio vectorial euclídeo, se llama subespacio ortogonal de  $C$  y se representa por  $C^\perp$  al conjunto formado por todos los vectores que son ortogonales a  $C$ . Si  $U$  es un subespacio del espacio vectorial euclídeo  $V$ , su subespacio ortogonal  $U^\perp$  es tal que la intersección  $U \cap U^\perp$  se anula. En el caso en el que  $C$  sea un espacio de Hilbert,  $C^\perp$  recibe el nombre de **complemento ortogonal**. Los espacios de Hilbert son una generalización de los espacios euclídeos sin restricciones de dimensiones finitas. Tienen nociones de distancia y ángulo y permiten algunas abstracciones geométricas.

### 2.1.2. Homomorfismos

#### Definición

Conviene diferenciar el concepto de **aplicación** y el de función. En una función no todos los elementos del conjunto partida tienen imagen en el conjunto de llegada. Una aplicación de un conjunto en otro es cualquier regla mediante la cual a cada elemento del conjunto de partida se le asocia un único elemento del conjunto de llegada.

Sean  $U$  y  $V$  dos espacios vectoriales definidos sobre  $\mathbb{R}$  y sea  $f : U \rightarrow V$  una aplicación lineal u *homomorfismo* entre los espacios vectoriales  $U$  y  $V$  si cumple:

- a)  $\forall x, y \in U \quad f(x + y) = f(x) + f(y)$  *aditividad*
- b)  $\forall \lambda \in \mathbb{R} \forall x \in U \quad f(\lambda x) = \lambda f(x)$  *homogeneidad*

Finalmente:

Si  $U \equiv V$  y  $f : U \rightarrow V$  es lineal se dice que  $f$  es un *endomorfismo*.

Si  $f : U \rightarrow V$  es lineal y biyectiva se dice que es un *isomorfismo*.

Si  $U \equiv V$  y  $f : U \rightarrow V$  es un isomorfismo se dice que es un *automorfismo*.

#### Imagen de un homomorfismo

Dada una aplicación lineal  $f : V \rightarrow W$  entre espacios vectoriales, al espacio  $f(V)$  se le llama imagen de  $f$ . Es decir, la imagen de  $f$  no es sólo el conjunto  $f(V)$  sino que se le considera con estructura de espacio vectorial, subespacio de  $W$ . La imagen de  $f$  es denotada por  $Im(f)$ . Una aplicación lineal  $f : V \rightarrow W$  será sobreyectiva si su imagen es todo el espacio  $W$  de llegada.

#### Núcleo de un homomorfismo

Si  $f : V \rightarrow W$  es un homomorfismo se define el núcleo de  $f$  o  $Ker(f)$  como:

$$Ker(f) = \{x \in V : f(x) = 0\} \quad (2.5)$$

Es decir, todos los elementos  $x$  de  $V$  cuya imagen es el vector nulo de  $W$ .

### Matriz asociada a un homomorfismo

Dada una aplicación lineal  $f$  del espacio vectorial  $U_n$  (de dimensión  $n$ ) en el espacio vectorial  $V_m$ . Sean  $B = (u_1, u_2 \dots u_n)$  y  $B' = (v_1, v_2 \dots v_m)$  las respectivas bases. Si es  $A_{m \times n}$  la matriz asociada a  $f$  respecto de las bases anteriores, se tendrá que si expresamos  $x$  e  $y$  en forma de matrices columna  $X$  e  $Y$  cuyos elementos son las coordenadas en  $B$  y  $B'$ , evidentemente:

$$Y = AX \quad (2.6)$$

Muchas de las definiciones sobre aplicación lineal tienen un equivalente para su matriz asociada. No se han incluido en esta sección todas –evidentemente– pero sirva de ejemplo que el núcleo de una matriz serán aquellos vectores  $x$  donde el producto  $Ax$  sea nulo. También se le llama el **espacio nulo**.

### Inversa de una matriz

La matriz inversa de la matriz  $A$  es otra matriz  $A^{-1}$  que verifica  $AA^{-1} = A^{-1}A = I$  siendo  $I$  la matriz *identidad* compuesta por unos en la diagonal y ceros en el resto de posiciones. La inversa de  $A$  se calcula según:

$$A^{-1} = \frac{1}{|A|} (Adj(A))^T \quad (2.7)$$

Donde  $Adj(A)$  es la matriz *adjunta* de  $A$  y  $|A|$  es su determinante. De todo ello se deduce que sólo pueden tener matrices inversas aquellas matrices cuadradas cuyo determinante sea distinto de cero.

## 2.2. Estadística unidimensional

Sea una variable que puede tomar una serie de valores. Considérese  $x$  a un vector que contenga distintos valores para una serie de observaciones de dicha variable. Ténganse en cuenta las siguientes definiciones para su correcto trato, empezando desde lo más básico:

### 2.2.1. Medidas de centralización

Son las medidas cuyo valor es representativo de un conjunto de datos, al centro del cual tiende a situarse.

- **Mediana:** Sean  $x_i$  los elementos del vector  $x$  ordenados crecientemente. La mediana es el valor que deja a cada lado la mitad de los valores de la muestra. Siendo  $n$  el número de datos de la muestra contenida en el vector  $x$ , la mediana será el elemento  $x_{\frac{n+1}{2}}$  si  $n$  es impar. En caso contrario vendrá definida como  $\frac{x_{\frac{n}{2}} + x_{\frac{n+1}{2}}}{2}$ .
- **Moda:** Valor más frecuente que tomen los  $x_i$ .
- **Media aritmética:** O promedio de los  $x_i$  habitualmente entendido como:  $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ .

### 2.2.2. Medidas de dispersión

Expresan el grado en el que los valores numéricos tienden a extenderse entorno a un valor medio.

- **Recorrido:** Diferencie entre el mayor y el menor de los  $x_i$  valores.
- **Varianza:** Una forma de medir la dispersión entorno a la media es calcular la media de las diferencias, pero como los valores por encima y por debajo se compensan es mejor calcular el cuadrado de las diferencias.

$$s^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (2.8)$$

Cuya raíz cuadrada  $s$  es lo que se llama desviación típica o estándar. La desviación estándar es una medida del grado de dispersión de los datos del valor promedio. Dicho de otra manera, la desviación estándar es simplemente el «promedio» o variación esperada con respecto de la media aritmética.

## 2.3. Estadísticos para datos multivariantes

Considérese ahora un estudio de distintas variables para distintos individuos. Una matriz de datos multivariantes típica tendrá la forma:

$$X = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1c} \\ x_{21} & x_{22} & \dots & x_{2c} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nc} \end{pmatrix} \quad (2.9)$$

Es una matriz de  $c$  vectores columna, donde cada vector representa a un individuo con los  $n$  valores observados. Según la literatura puede encontrarse los datos organizados en una matriz de vectores fila o columna. En este caso, hay que recordar:

$$\begin{aligned} c &= \text{número de individuos, de objetos de estudio} \\ n &= \text{número de datos observados para cada individuo} \end{aligned} \quad (2.10)$$

Y en cualquier caso puede entenderse que las filas representan las variables mientras que las columnas representan los individuos.

### 2.3.1. Medidas básicas de estadística multivariante

Se considera la generalización –en esta sección– de las fórmulas y ecuaciones vistas antes, atendiendo a la matriz 2.9. Obsérvese también –sin confundir– que se manejan vectores columna y vectores fila usando el subíndice  $i$  para denominar a unos y a otros o a sus elementos, indistintamente. Así, puede ser correcto –según el contexto– hablar de la variable  $x_i$  (lo cual hace referencia a un vector fila de mismo tipo de datos pero con valores para diferentes individuos) o hablar del individuo  $x_i$  (que sería un vector columna con todos los valores de las variables tomados para un mismo individuo).



- **Media:** Análogo al caso univariante, ahora se tiene el vector media  $\mu$  con la forma:

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix} \quad (2.11)$$

Donde los  $\mu_i$  resultan de calcular la media al vector fila  $x_i = (x_{i1} \ x_{i2} \ \dots \ x_{ic})$  como en el caso univariante. Puede entenderse que  $\mu$  representa los valores del «individuo medio».

- **Varianza:** De la misma manera, se representa la varianza como un vector  $\sigma^2$  descrito por:

$$\sigma^2 = \begin{pmatrix} \sigma_1^2 \\ \sigma_2^2 \\ \vdots \\ \sigma_n^2 \end{pmatrix} \quad (2.12)$$

Donde  $\sigma_i^2$  será calculado según:

$$\sigma_i^2 = \frac{1}{c-1} \sum_{k=1}^c (x_{ik} - \mu_i)^2 \quad (2.13)$$

- **Covarianza:** La covarianza de dos variables  $x_i$  y  $x_j$  se define por:

$$\sigma_{ij} = \frac{1}{c-1} \sum_{k=1}^c (x_{ik} - \mu_i)(x_{jk} - \mu_j) \quad (2.14)$$

Nótese que si  $i = j$  tendríamos la varianza para la variable  $x_i$ . La covarianza de  $x_i$  y  $x_j$  se denota usualmente como  $\sigma_{ij}$ . Por lo tanto, la varianza de la variable  $x_i$  se denota amenudo como  $\sigma_{ii}$  en lugar de  $\sigma_i^2$ . Con  $n$  variables  $x_1, x_2, \dots, x_n$  existen  $n$  varianzas y  $n(n-1)/2$  covarianzas.

- **Matriz de covarianza** también llamada matriz de varianza-covarianza o simplemente matriz de dispersión<sup>1</sup>: Con  $n$  variables podemos formar una matriz  $\Sigma$  simétrica  $n \times n$  tal que:

$$\Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma_{nn} \end{pmatrix} \quad (2.15)$$

Suele encontrarse que la matriz  $\Sigma$  es estimada<sup>2</sup> por medio de la matriz  $S$  que viene dada por:

$$S = \frac{1}{c-1} \sum_{i=1}^c (x_i - \mu)(x_i - \mu)^T \quad (2.16)$$

<sup>1</sup>En inglés es la «*scatter matrix*», como puede encontrarse en parte de la bibliografía.

<sup>2</sup>Obsérvese que por concordancia con las ecuaciones de varianza, y por la estructura misma de esta presentación, se ha conservado el denominador  $c-1$ . Según [11], así formulado debe considerarse a  $S$  como el estimador imparcial «*unbiased*» (o no sesgado) de  $\Sigma$ . Cuando apareciera  $c$  en el denominador,  $S$  sería el estimador de máxima verosimilitud.

Donde  $\mu$  es el vector definido por la ecuación 2.11. El vector  $x_i$  es –en este caso– un vector columna que representa a un individuo.

$$x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \end{pmatrix} \quad (2.17)$$

Llegados a este punto, es importante darse cuenta de la diferencia entre la ecuación 2.16 y las ecuaciones 2.13 y 2.14. Mientras que en el interior de aquellos sumatorios se calculaba un valor, ahora en el interior del sumatorio se libra un producto cuya resultante es una matriz<sup>3</sup> atendiendo al producto de un vector columna y su (traspuesto) vector fila.

Por cierto, si se considera una matriz  $X$  a la que ya se le ha restado las medias, la matriz de dispersión es equivalente a  $\frac{1}{c-1}XX^T$ .

- **Coefficiente de correlación lineal:** Es una medida  $\rho$  que expresa el grado de asociación lineal entre dos variables  $x_i$  y  $x_j$  y su valor estará comprendido entre  $-1$  y  $+1$ . Expresando los valores altos (cerca de uno) y positivos un grado muy alto de relación lineal y los negativos lo contrario.

$$\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}} \quad (2.18)$$

- **Matriz de correlación:** Con  $n$  variables existen  $n(n-1)/2$  distintas correlaciones que pueden ser dispuestas en una matriz  $n \times n$  llamada  $R$  cuya diagonal está formada por unos. Esta matriz puede ser expresada en términos de la matriz de covarianza  $\Sigma$  como sigue:

$$R = D^{-1/2}\Sigma D^{-1/2} \quad (2.19)$$

Donde  $D^{-1/2} = \text{diagonal}(1/\sqrt{\sigma_{ii}})$ .

### 2.3.2. Punto de vista geométrico

Suponiendo dos nubes de puntos distribuidas geométricamente como se muestra en la figura 2.1, puede ocurrir que la varianza sea la misma para uno y otro caso, sin embargo la covarianza en el círculo es cero pero no la de la elipse.

### 2.3.3. Vectores y valores propios

#### Vectores y valores propios de un endomorfismo

Dado un endomorfismo cualquiera en un espacio vectorial  $V$ , o sea, una aplicación  $f : V \rightarrow V$  que a cada vector  $x \in V$  le atribuye un vector  $f(x)$  también en  $V$  se dan las siguientes definiciones:

---

<sup>3</sup>En álgebra de matrices tiene sentido el producto de la matriz  $A_{m \times p}$  y la matriz  $B_{p \times n}$  cuya resultante  $A \cdot B$  es una matriz  $m \times n$ . En el caso de la ecuación 2.16 se multiplica una matriz de  $n \times 1$  (o sea un vector columna) por su traspuesta, un vector fila o matriz  $1 \times n$ , por lo que el resultado es una («auténtica») matriz  $n \times n$

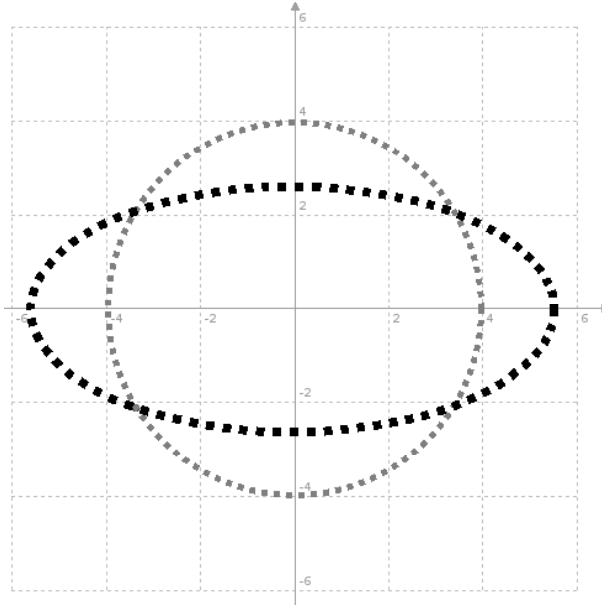


Figura 2.1: Ilustración del significado geométrico de la varianza

- a) Un valor propio o autovalor de  $f$  es todo escalar  $\lambda \in \mathbb{R}$  tal que existe al menos un vector no nulo  $x$  de  $V$  para el que se verifica  $f(x) = \lambda x$ .
- b) Un vector propio o autovector, correspondiente al autovalor  $\lambda$  de  $f$ , es todo vector no nulo de  $V$  tal que  $f(x) = \lambda x$ .

En otras palabras, los **autovectores** de un endomorfismo son vectores que o no se ven alterados por el endomorfismo o se ven multiplicados por un escalar ( $\lambda$ ) que no varía su dirección. Siendo el **autovalor** el factor de escala por el que ha sido multiplicado.

### Vectores y valores propios de una matriz

Las definiciones anteriores pueden extenderse a una matriz si consideramos ésta asociada a una aplicación lineal. Siendo así, dada una matriz cuadrada  $A$ , de tamaño  $n$ , es decir  $A \in \mathbb{R}^{n \times n}$  se dice que un escalar  $\lambda$  es un valor propio o un autovalor si existe alguna matriz no nula  $X \in \mathbb{R}^{n \times 1}$  tal que  $AX = \lambda X$ . Se dice que una matriz columna  $X \neq 0$  del espacio  $\mathbb{R}^{n \times 1}$  es un autovector o vector propio de  $A$  correspondiente al autovalor  $\lambda$ , si se verifica la ecuación:

$$AX = \lambda X \quad (2.20)$$

De esta definición se entiende que si la matriz  $A$  es la matriz asociada a un endomorfismo  $f$  los valores y vectores propios de  $A$  lo serán también del endomorfismo  $f$ .

### El problema del autovalor generalizado

Se llama así al problema de, dadas dos matrices  $A$  y  $B$  encontrar las parejas  $\alpha_k$  y  $\beta_k$  de escalares, y los vectores  $x_k$  que cumplan:

$$\begin{aligned}\beta_k Ax_k &= \alpha_k Bx_k \\ k &= 1, 2 \dots n\end{aligned}\tag{2.21}$$

Los escalares  $\lambda_k = \alpha_k / \beta_k$  son los autovalores generalizados y  $x_k$  los autovectores generalizados. Si la matriz  $B$  tiene inversa los vectores  $x_k$  serán los vectores propios de  $B^{-1}A$ . Pudiéndose reescribir la ecuación 2.21 como un problema estándar de autovalores, resolviendo:

$$B^{-1}Ax_k = \lambda_k x_k\tag{2.22}$$

### 2.3.4. Método de Gram-Schmidt

El método de Gram-Schmidt es un método para encontrar una base ortonormal<sup>4</sup> un conjunto de vectores en un espacio. Partiendo de los vectores  $S = \{v_1 \dots v_n\}$  se encuentran los vectores ortogonales  $S' = \{u_1 \dots u_n\}$  que generan el mismo subespacio que los primeros. Finalmente se normalizará el conjunto de vectores  $S'$  (dividiendo por su tamaño).

#### Algoritmo

Se define el operador proyección de  $v$  en la dirección de  $u$  como sigue:

$$proj_u(v) = \frac{\langle v, u \rangle}{\langle u, u \rangle} u\tag{2.23}$$

Donde  $\langle v, u \rangle = v^T u$  es el producto escalar estándar en  $\mathbb{R}^n$ . El algoritmo de Gram-Schmidt seguiría la iteración:

$$\begin{aligned}u_1 &= v_1 & e_1 &= \frac{u_1}{\|u_1\|} \\ u_2 &= v_2 - proj_{u_1} v_2 & e_2 &= \frac{u_2}{\|u_2\|} \\ u_3 &= v_3 - proj_{u_1} v_3 - proj_{u_2} v_3 & e_3 &= \frac{u_3}{\|u_3\|} \\ \vdots & & \vdots & \\ u_n &= v_k - \sum_{i=1}^{k-1} proj_{u_i} v_k & e_n &= \frac{u_3}{\|u_3\|}\end{aligned}\tag{2.24}$$

## 2.4. Análisis de componentes principales

### 2.4.1. Introducción

Un problema típico en el análisis de datos multivariantes es la reducción de la dimensionalidad de los mismos. El objetivo es poder describir con precisión los valores por medio de un subconjunto

<sup>4</sup>Una base ortonormal es aquella que además de ser ortogonal, la norma de cada elemento que la compone es igual a 1.

de los mismos. El análisis de componentes principales tiene este objetivo: dadas  $c$  muestras de un conjunto de  $n$  variables se analiza si es posible representar adecuadamente esa información por un número menor de variables construidas como combinaciones lineales de las originales.



Figura 2.2: Karl Pearson (1857–1936) fue Catedrático de Matemáticas y Eugenésia en la Universidad de Londres.

Esta técnica recibe varios nombres debido a la evolución por la que fue siendo concebida. En 1901 el científico británico Karl Pearson (1857-1936), inventor del contraste que lleva su nombre y uno de los fundadores de la Estadística en el siglo XIX, introdujo el punto de partida en sus trabajos de ajustes ortogonales por mínimos cuadrados. En 1933, Harold Hotelling hizo un primer desarrollo teórico y le dio el nombre de análisis de componentes principales, a él se debe que se conozca también como la transformación de Hotelling.

Posteriormente los trabajos –independientes entre sí– de Kari Karhunen (en 1946) y de Michel Loève en 1955, le dieron uno de sus nombres más conocidos, la transformación de Karhunen-Loève, «transformada  $KL$ » o  $TDKL$ , transformada discreta de Karhunen-Loève.

### 2.4.2. Planteamiento

Supóngase que se dispone de los valores de  $n$ -variables en  $c$  elementos a estudio, dispuestos en una matriz  $X$  de dimensiones  $n \times c$ , donde las columnas contienen los elementos y las filas las variables. Se ha de haber normalizado la matriz  $X$  restando a cada variable su media, de manera que las variables de la matriz  $X$  tienen media cero y su matriz de covarianzas viene dada por  $(1/n)XX^T$ .

El objetivo es encontrar un espacio de dimensión más pequeña que represente a los datos. La figura 2.3 muestra un ejemplo para  $\mathbb{R}^2$  en el que se han dibujado una serie de puntos siguiendo una distribución más o menos lineal. Se considera un subespacio de dimensión 1, una recta. Se desea que las proyecciones de los puntos sobre esta recta mantengan sus posiciones relativas. Puede calcularse esta recta usando cálculo geométrico y minimizando la suma de las distancias de los puntos a la «recta de proyección», lo cual equivale a maximizar la suma al cuadrado de los valores de las proyecciones. Al ser variables de media cero, ello equivale a maximizar la suma de las varianzas. Intuitivamente supone encontrar la recta que más cerca pase de todos los puntos y mejor mantenga

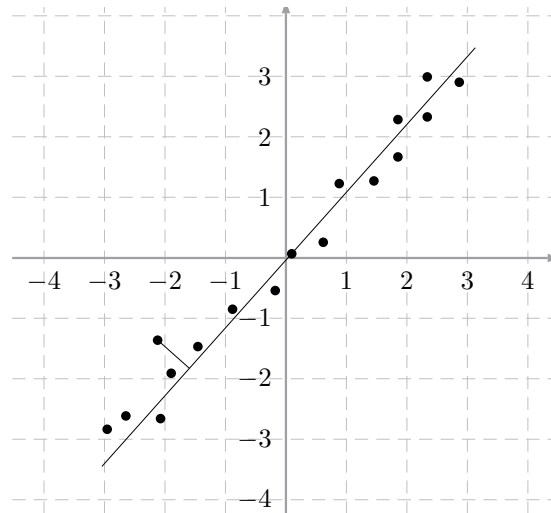


Figura 2.3: Enfoque descriptivo del análisis de componentes principales.

(al proyectarse esos punto en sí misma) la variabilidad original.

El *primer componente principal* se define como la combinación lineal de las variables originales que tienen varianza máxima. Con el planteamiento geométrico anterior y teniendo presentes las varianzas iniciales, el primer desarrollo matemático de la técnica de los componentes principales, plantea cómo ir calculando uno por uno dichos componentes. No hay necesidad de abordar dicha técnica en esta sección, pero sí que conviene recordar para tener bien presente, la generalización de esta transformación y repasar la técnica tal como habitualmente se utiliza.

### 2.4.3. Generalización

Esta sección no tiene el objetivo de explicar por qué funciona esta técnica sino proveer de una visión de conjunto de la técnica, a modo de algoritmo. Servirán como datos de ejemplo una serie de 10 puntos que sirve además para seguir gráficamente el proceso. En concordancia con la sección 2.3 se diría que en este ejemplo hay  $c = 10$  muestras y  $n = 2$  dimensiones (2 variables).

x	2.5	0.5	2.2	1.9	3.1	2.3	2	1	1.5	1.1
y	2.4	0.7	2.9	2.2	3	2.7	1.6	1.1	1.6	0.9

1. **Disponer los datos** en forma de matriz. Representando las columnas a las muestras y las filas a las variables.

$$X = \begin{pmatrix} 2,5 & 0,5 & 2,2 & 1,9 & 3,1 & 2,3 & 2,0 & 1,0 & 1,5 & 1,1 \\ 2,4 & 0,7 & 2,9 & 2,2 & 3,0 & 2,7 & 1,6 & 1,1 & 1,6 & 0,9 \end{pmatrix} \quad (2.25)$$

2. **Restar la media:** Restar la media para cada dimensión. Para todos los valores de  $x$  se calcula  $\bar{x}$  y se le resta a todos los valores de  $x$ . Con  $y$  lo mismo. En el caso del ejemplo

$(\bar{x}, \bar{y}) = (1,81, 1,91)$ , de manera que la matriz  $X$  *normalizada* quedaría:

$$X = \begin{pmatrix} 0,69 & -1,31 & 0,39 & 0,09 & 1,29 & 0,49 & 0,19 & -0,81 & -0,31 & -0,71 \\ 0,49 & -1,21 & 0,99 & 0,29 & 1,09 & 0,79 & -0,31 & -0,81 & -0,31 & -1,01 \end{pmatrix} \quad (2.26)$$

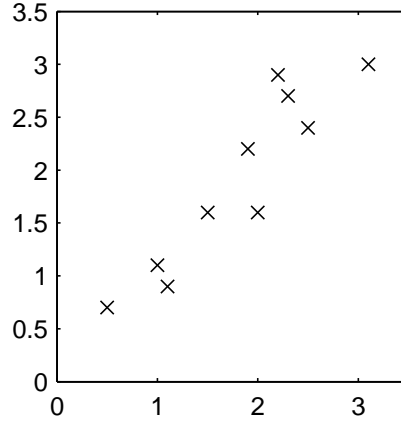


Figura 2.4: Gráfica con los puntos del ejemplo para un cálculo de PCA.

3. **Calcular la matriz de covarianza:** Tal como se ha visto en la sección 2.3 la matriz de covarianza es de dimensión  $n \times n$ , aplicado al ejemplo:  $2 \times 2$ .

$$S = \begin{pmatrix} 0,6166 & 0,6154 \\ 0,6154 & 0,7166 \end{pmatrix} \quad (2.27)$$

4. **Calcular los valores y vectores propios de la matriz de covarianza:** En el caso del ejemplo, los valores propios serían  $(0,0491, 1,2840)$  y los vectores propios normalizados vendrían dados por la matriz:

$$\text{vectores propios} = \begin{pmatrix} -0,7352 & 0,6779 \\ 0,6779 & 0,7352 \end{pmatrix} \quad (2.28)$$

Observando la figura 2.4 se entiende el significado de estos resultados. Los vectores propios están asociados a los valores. El primer valor propio del ejemplo es 0,0491 y el vector propio correspondiente es  $(-0,7352, 0,6779)$ , el segundo valor es 1,2840 con el vector  $(0,6779, 0,7352)$ . El peso de los valores marca la importancia y la dirección del vector marca la máxima variación de los datos. Ordenados los valores propios de mayor o menor nos queda que en la dirección del segundo vector  $(0,6779, 0,7352)$  es en la que se halla la máxima variación de los datos, y la siguiente viene determinada por el segundo vector  $(-0,7352, 0,6779)$ .

5. **Elegir los componentes y formar una base con sus vectores:** De acuerdo con la importancia marcada por los valores propios, existen distintas estrategias para elegir un número de ellos, que será la reducción de dimensionalidad.

- Ordenar los  $\lambda_i$  autovalores y comenzar eligiendo por el primero hasta que la suma de los restantes valgan igual o menos que  $\lambda_i$ .

- Los valores propios de la matriz de dispersión son una expresión de la varianza de ésta. Y asociadamente con los vectores, expresan las direcciones principales de variación. Desde este punto de vista, entendiendo que la suma total de los autovalores es una expresión de la varianza de los datos, puede utilizarse un porcentaje (el 80 o 90 por cien) de corte que permita deshechar aquellas direcciones de dispersión que no aportan nada o casi nada de información.
- Desechar aquellos componentes asociados a valores propios inferiores a una cota. Suele fijarse como la varianza media. Estas últimas dos técnicas son arbitrarias y en la literatura se aconseja manejar adecuadamente.

En el caso del ejemplo hay dos opciones, elegir los dos vectores o quedarse con (0,6779, 0,7352) cuyo valor propio es mayor 1,2840. En tal caso se reduciría de dos dimensiones a una.

#### 6. Procesar los datos:

Llamando  $P$  a la matriz cuyas columnas son los vectores propios de  $S$  adecuadamente ordenados. Calcular los componentes principales equivale a aplicar una transformación ortogonal a las variables originales ( $X$ ) para obtener unas nuevas variables  $Z$  incorreladas entre sí. Esta operación puede interpretarse como elegir unos nuevos ejes coordenados que coincidan con los «*ejés naturales*» de los datos. La transformación viene dada por:

$$Z = P^T X \quad (2.29)$$

En el caso del ejemplo:

$$P = \begin{pmatrix} 0,6779 & -0,7352 \\ 0,7352 & 0,6779 \end{pmatrix} \quad (2.30)$$

Y los nuevos valores  $Z$  pueden reescribirse como muestra la tabla 6. La figura 2.5 muestra los datos después de la transformación.

x	y
0.8280	-0.1751
-1.7776	0.1429
0.9922	0.3844
0.2742	0.1304
1.6758	-0.2095
0.9129	0.1753
-0.0991	-0.3498
-1.1446	0.0464
-0.4380	0.0178
-1.2238	-0.1627

Si se hubiera elegido sólo el primer vector propio (asociado al mayor autovalor) para formar la matriz  $P$  el resultado del producto matricial de la ecuación 2.29 hubiera dado como resultado un matriz de  $1 \times n$ , que representaría los datos en una sola dimensión.

## 2.5. Análisis Discriminante Lineal de Fisher

PCA es la técnica estadística más famosa para reducir la dimensionalidad. Busca las direcciones en los datos que tienen mayor varianza y proyecta los datos en ellas. De esa manera, se obtienen



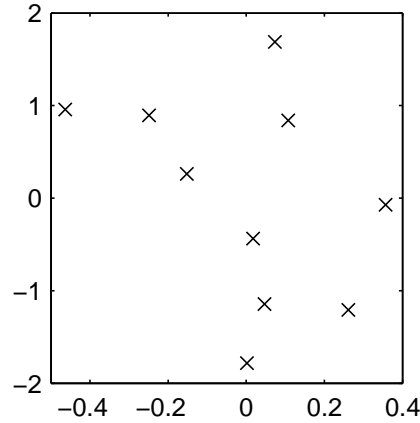


Figura 2.5: Los datos después de la transformación de PCA aparecen con mayor variación entre sí.

representaciones de los datos de menor dimensión eliminando direcciones «ruidosas». Se trata de una técnica no supervisada y como tal no incluye información etiquetada de los datos. Por ejemplo si se disponen dos distribuciones de puntos a lo largo de dos rectas paralelas y próximas entre sí, ignorando las etiquetas, la varianza del conjunto de datos total es la dirección de las rectas. Para clasificar ambas distribuciones se trataría de una pésima proyección porque todas las etiquetas están mezcladas y la información útil se destruye.

Una aproximación para resolver dicho caso la propuso en 1936 el biólogo y estadístico británico Sir Ronald Aylmer Fisher (1890-1962). Considerando una matriz  $w$  de proyección análoga al papel que cumplía la matriz de proyección  $P$  en la ecuación 2.29 para PCA, el objetivo es encontrar una  $w$  que maximice:

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \quad (2.31)$$

Donde  $S_B$  es la matriz de covarianza (o matriz de dispersión) interclase<sup>5</sup> y  $S_W$  es la matriz de dispersión intraclase<sup>6</sup>. Es decir,  $S_B$  habla de la dispersión entre sí que tienen las clases de distribuciones de puntos mientras que  $S_W$  habla de la dispersión interna que tienen las clases de puntos. Suponiendo los datos de  $c$  clases, donde en cada clase tenemos  $n$  muestras,  $S_B$  y  $S_W$  se definirían:

$$S_B = \sum_c n (\mu_c - \bar{x}) (\mu_c - \bar{x})^T \quad (2.32)$$

$$S_W = \sum_c \sum_{i \in c} (x_i - \mu_c) (x_i - \mu_c)^T \quad (2.33)$$

Donde  $\mu_c$  es la media de la clase  $c$  y  $\bar{x}$  es la media total, cumpliéndose:

$$\mu_c = \frac{1}{n} \sum_{i \in c} x_i \quad (2.34)$$

$$\bar{x} = \frac{1}{n} \sum_i x_i = \frac{1}{n} \sum_c n \mu_c \quad (2.35)$$

<sup>5</sup>Between class scatter matrix.

<sup>6</sup>Within class scatter matrix

Si se considera  $S_T$  la matriz de dispersión total,

$$S_T = \sum_i (x_i - \bar{x}) (x_i - \bar{x})^T \quad (2.36)$$

se cumple  $S_T = S_W + S_B$

De acuerdo con [11] el problema de maximizar  $J(w)$  conduce a la ecuación de vectores propios generalizados, considerando los autovalores en la diagonal de una matriz  $\lambda$ , dicha ecuación se describiría como:

$$S_B w = \lambda S_W w \quad (2.37)$$

Lo cual concuerda con las ecuaciones 2.21 y 2.22.



## Capítulo 3

# Métodos de clasificación basados en subespacios

### 3.1. Preliminar

Habitualmente se considera una imagen digital como una matriz donde cada valor representa el color del píxel correspondiente. Si en lugar de representarla así, se colocan las filas de la matriz una detrás de otra, lo que se tiene es un vector. De esta manera, una imagen de  $n \times m$  píxeles, podrá ser manejada como un vector de  $d = n \times m$  elementos. Asumida esta transformación, resulta que todas las imágenes de  $m \times n$  tienen un espacio vectorial asociado en  $\mathbb{R}^d$ . La aplicación tanto del álgebra lineal como de análisis estadístico a dichos datos, ha permitido construir toda una línea de desarrollo científico a lo largo de la cual se han ido sumando –con distintos resultados– varios métodos, algoritmos y estrategias diversas. En este capítulo se pretende revisar, de entre los que se atribuyen a este campo, los métodos principales que utilizan proyecciones en subespacios.

### 3.2. *Eigenfaces*

#### 3.2.1. Contexto

La idea de aplicar análisis de componentes principales (en adelante PCA) para representar imágenes de caras en una dimensión baja fue introducida por primera vez por Lawrence Sirovich y Robert M. Kirby en 1987 ([16]). Comenzando con un conjunto original de imágenes de caras calculaban el mejor sistema de coordenadas para comprimir la imagen, donde cada coordenada es una imagen que ellos llamaban *eigenpictures*<sup>1</sup>, «*imágenes propias*». Esta idea fue tomada poco después por Matthew Turk y Alex Pentland en [17] quienes desarrollaron el método de reconocimiento de Sirovich y Kirby y lo reformularon con la denominación *Eigenfaces*, «*caras propias*». En cierta literatura se habla de PCA y *Eigenfaces* indistintamente, lo cual no es del todo correcto: El método PCA es un método general de análisis de datos, por otra parte *Eigenfaces* es un método de reconocimiento que utiliza PCA con alguna variación.

---

<sup>1</sup>El término *eigen* lo introdujo por primera vez en este contexto David Hilbert en 1904, significa «característico» o «individual», en alemán. Los *eigen vectors* son los vectores propios.

### 3.2.2. Calcular *Eigenfaces*

Sea una imagen de una cara dada por una matriz<sup>2</sup>  $N \times N$ . Puede considerarse como un vector de dimensión  $N^2$ , de manera que una imagen típica (pequeña) de 256 por 256 será un vector de dimensión 65536 o de forma equivalente, un punto en un espacio de 65536 dimensiones. Un conjunto de imágenes trazará una nube de puntos en este gran espacio.

La idea de PCA es encontrar la base que mejor expresan la distribución de las imágenes de las caras dentro del espacio completo, al que Kirby y Sirovich llaman espacio de las imágenes. Estos vectores describen la base del subespacio de las imágenes de caras, el «espacio de las caras». Cada vector de dimensión  $d = N^2$  describe una imagen  $N \times N$ , y es una combinación lineal de los vectores base del subespacio. Como estos vectores son los vectores propios de la matriz de covarianza correspondiente al espacio original de las imágenes, y como son parecidas a una cara, se les llama *Eigenfaces*.

Sea el conjunto de imágenes de caras  $x_1, x_2, x_3 \dots x_m$  (considerando vectores columna de dimensión  $d$  podemos contruir la matrix  $X$  de dimensiones  $d \times m$ ). La media del conjunto (o la cara media) se define  $\mu = \frac{1}{m} \sum_{i=1}^m x_i$ . Cada cara difiere de la media por el vector  $x_i - \mu$ . Este conjunto de vectores grandes es sometido a PCA lo cual buscará un conjunto de  $m$  vectores ortonormales  $u_k$  que describe la distribución de los datos.

Los vectores  $u_k$  y los escalares  $\lambda_k$  son los vectores y valores propios respectivamente de la matriz de covarianza:

$$\begin{aligned} S &= \frac{1}{m} \sum_{i=1}^m (x_i - \mu)(x_i - \mu)^T \\ &= AA^T \end{aligned} \quad (3.1)$$

Donde  $A$  sería la matriz  $X$  normalizada (a cada  $x_i$  columna se le ha restado la media). La matriz  $S$  tiene una dimensión de  $d \times d$  ( $= N^2 \times N^2$ ), lo que hace de extraer sus vectores y valores propios una tarea computacionalmente prohibitiva para imágenes de un tamaño normal.

Si el número de puntos en el espacio de imágenes es menor que la dimensión del espacio  $m < N^2$ , habrá como mucho  $m - 1$  vectores propios significativos. Puede resolverse el problema tomando apropiadamente combinaciones lineales de las imágenes. Si en lugar de calcular  $AA^T$  consideramos  $v_i$  los vectores propios de  $A^T A$  (y  $\alpha_i$  sus valores propios) resultará que:

$$A^T A v_i = \alpha_i v_i \quad (3.2)$$

Premultiplicando a ambos lados por  $A$ , se tiene:

$$AA^T A v_i = \alpha_i A v_i \quad (3.3)$$

Donde  $A v_i$  son los vectores propios de  $S = AA^T$

Partiendo de este análisis, se construye la matriz  $A^T A$  de dimensión  $m \times m$  y se encuentran los  $m$  vectores propios.

---

<sup>2</sup>Se habla aquí de una matriz cuadrada por guardar concordancia con el original (en [17]) pero no tiene por qué serlo.

### 3.2.3. Algoritmo

Según [17], un algoritmo típico de reconocimiento de caras utilizando las *Eigenfaces* seguiría:

1. Tomar un conjunto inicial de imágenes (el conjunto de entrenamiento). Este conjunto debería incluir un número de imágenes para cada persona, con variaciones en la expresión y en la iluminación.
2. Calcular la matriz  $m \times m$  normalizada y encontrar sus vectores y valores propios. Elegir los  $m'$  vectores propios con los valores propios asociados más altos.
3. Combinar el conjunto de entrenamiento con los vectores propios para calcular las  $m'$  *Eigenfaces*.
4. Para cada clase (para cada individuo) elegir al menos una (o promediando más de una) de sus imágenes y calcular el vector de clase (proyectándolo en el espacio de caras)  $\Omega_k$ .
5. Para cada nueva imagen  $y$  calcular su vector  $\Omega = u_k(y - \mu)$  y calcular la distancia a cada vector de clase.

## 3.3. *Fisherfaces*

### 3.3.1. Justificación

La misma persona con la misma expresión facial y vista desde el mismo punto de vista puede aparecer dramáticamente diferente cuando las fuentes de luz iluminan la cara desde diferentes direcciones. Originalmente (en [18]), el método de *Fisherfaces* explota dos observaciones:

1. Todas las imágenes de una superficie de *Lambert*<sup>3</sup> tomadas desde un punto de vista fijo pero con variaciones en la iluminación, se proyectan en un subespacio lineal 3d del espacio de la imagen (de dimensionalidad alta).
2. En la práctica por razones de sombra, reflejos y expresiones faciales, la anterior observación no se mantiene exactamente. En la práctica, ciertas regiones de la cara pueden tener variabilidad de imagen a imagen que frecuentemente deriva significativamente del subespacio lineal y consecuentemente merma la fiabilidad del reconocimiento.

El método de *Eigenface* se basa también en la proyección lineal del espacio de las imágenes en un espacio de características de dimensión menor, ya que uno de los objetivos es encontrar una proyección lineal de las imágenes de las caras en un espacio menor de características insensible a las variaciones de la luz y la expresión facial. Sin embargo, al utilizar PCA para reducir la dimensión (ya que PCA busca maximizar la dispersión) se mantienen variaciones indeseadas debidas a cambios en la luz y en la expresión facial. De acuerdo con la literatura, en [18] se insiste en que «*las variaciones entre las imágenes de la misma cara debidas a la dirección del punto de vista y la luz son siempre más grandes que las variaciones en la identidad de la cara*». Mientras que las proyecciones por PCA son óptimas para reconstruir los datos desde bases de menor dimensión, no son óptimas para la discriminación.

---

<sup>3</sup>Según la Ley de Lambert: Un cuerpo que radia obedece a la ley de Lambert si su luminancia espectral energética es la misma para un elemento cualquiera de su superficie, y no depende de la dirección de emisión.

### 3.3.2. Formulación

Como el conjunto de aprendizaje está etiquetado, tiene sentido utilizar dicha información para construir un método de reducción de dimensionalidad del espacio de características más confiable. En este sentido el análisis discriminante de Fisher intenta proyectar los datos de manera que su nueva dispersión sea óptima para clasificación. Mientras PCA busca los vectores que mejor describen los datos, LDA busca los vectores que proporcionan mejor discriminación entre clases después de la proyección.

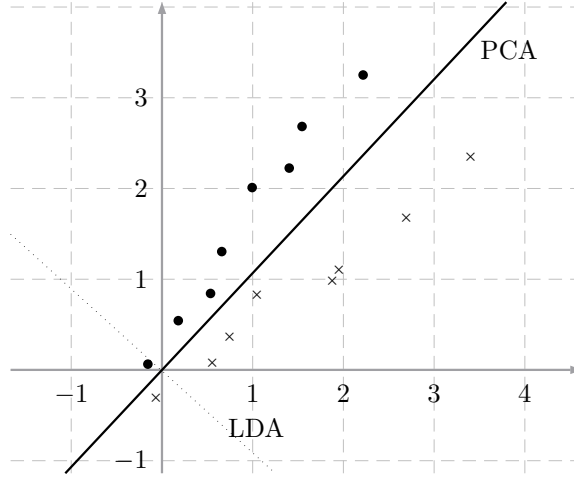


Figura 3.1: LDA, ejemplo en dos dimensiones.

La figura 3.1 muestra un ejemplo habitual en dos dimensiones para ilustrar los beneficios de LDA. En la recta que representa la dirección principal de los datos según PCA, los puntos se proyectarían mezclados y sería imposible discriminarlos. Mientras PCA describe la dispersión total, LDA describe mejor la dispersión entre clases.

En el problema de reconocimiento de caras, de acuerdo con [18], existe la dificultad de que la matriz de dispersión intraclase siempre es *singular*, es decir que no tiene inversa. Esto ocurre porque en general, el número de imágenes en el conjunto de aprendizaje siempre es menor que el número de píxeles en cada imagen.

Para salvar este obstáculo, originalmente, el método de las *Fisherfaces* propone una variación al LDA tradicional: Aunque existen otras estrategias, se puede *evitar* el problema proyectando el conjunto de imágenes en un espacio de dimensión más baja donde la resultante  $S_W$  es no singular. Se utiliza PCA para reducir la dimensión del espacio de características y después se aplica el LDA estándar. Puede formularse estrictamente que la proyección óptima  $W_{opt}$  de acuerdo con:

$$W_{opt}^T = W_{lda}^T W_{pca}^T \quad (3.4)$$

Donde  $W_{pca}$  se calcula partiendo de la matriz de dispersión total  $S_T$  y  $W_{lda}$  se obtiene buscando una matriz  $w$  que maximice  $J_{lda}(w)$ :

$$J_{lda}(w) = \frac{w^T w_{pca}^T S_B w_{pca} w}{w^T w_{pca}^T S_W w_{pca} w} \quad (3.5)$$

### 3.3.3. Algoritmo

Sintéticamente:

- Construir la matriz de proyección con los vectores propios generalizados a  $S_B$  y  $S_W$  correspondientes a los valores propios no nulos.
- Para cada clase contruir un vector de características proyectando con la matriz anterior. Se puede elegir uno por cada clase o promediarlos.
- Previamente, puede ser necesaria una reducción de dimensionalidad para poder tratar con  $S_B$  y  $S_W$ , dicha reducción deberá hacerse a los vectores de características y a los de prueba para la clasificación.
- La clasificación se resuelve por la distancia a los vectores de características.

## 3.4. Vectores comunes discriminantes

Esta técnica ha sido presentada en [22] por Cevikalp, Neamtu, Wilkes y Barkana por primera vez, aunque la idea de vector común había sido propuesta con anterioridad ([20] y [21]). Aunque se trata de técnicas diferentes están relacionadas, y conviene repasar el concepto del vector común para comprender mejor la técnica de los vectores comunes discriminantes.

La técnica del vector común extrae de un conjunto de entrenamiento el vector común, es decir, aquél vector con respecto al cual varían los demás (por sus diferencias y en ellas mismas) vectores de la muestra. Una vez calculado dicho vector, el clasificador busca en los vectores de prueba aquéllos que más próximos están al vector común de una cierta clase.

La técnica de los vectores comunes discriminantes se pregunta si existe una proyección a un subespacio tal que todos los elementos de una misma clase proyectados en dicho subespacio tengan una única proyección. Es decir, en dicho subespacio estén representados por un único vector al que Cevikalp, Neamtu, Wilkes y Barkana han llamado el vector común discriminante.

### 3.4.1. Vector común

En primer lugar, tanto en [20] como [21] se establece que los vectores diferencia de cada clase en el conjunto de entrenamiento constituyen un único subespacio llamado *subespacio diferencia*. Se eliminan todas las diferencias dentro de una cierta clase del conjunto de entrenamiento. Sólo un vector contiene las características invariantes para cada clase, el *vector común*.

Cada vector común es único para cada muestra del conjunto de entrenamiento y representa las propiedades comunes de la clase. Por otro lado, en el conjunto de pruebas, cuando todas las diferencias en el subespacio diferencia han sido eliminadas, los vectores que quedan no son únicos. Cevikalp, Neamtu, Wilkes y Barkana los llaman los *vectores remanentes*. Según [21] los vectores remanentes están habitualmente más próximos a los vectores comunes de su propia clase que a los de otras clases.

Sea un espacio vectorial  $n$ -dimensional en  $\mathbb{R}^n$  y  $\langle x, y \rangle = x_1y_1 + x_2y_2 + \dots + x_ny_n$  el producto escalar de los vectores  $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$  e  $y = (y_1, y_2, \dots, y_n) \in \mathbb{R}^n$ . Y sea  $\|x\|$  la norma euclídea del vector  $x$ . Sean los vectores linealmente independientes  $n$ -dimensionales  $a_1, a_2, \dots, a_m \in \mathbb{R}^n$  con  $m <$



$n$ , considerando todos los  $a_i$  pertenecientes a una misma clase de un conjunto de entrenamiento, se puede describir como un sumatorio de un *vector común*  $x$  que representa las propiedades comunes de la clase y un vector diferencia  $a_{i,dif}$  que representa las propiedades individuales del  $i$ -ésimo.

$$\begin{aligned} a_1 &= x + a_{1,dif} \\ a_2 &= x + a_{2,dif} \\ &\vdots \\ a_m &= x + a_{m,dif} \end{aligned} \tag{3.6}$$

El vector común representa las propiedades comunes o características invariantes de la clase. En la relación 3.6 existen  $m$  ecuaciones pero existen  $m + 1$  vectores desconocidos  $x, a_{1,dif}, a_{2,dif} \dots a_{m,dif}$ . Cada vector común  $x$  puede calcularse como  $x = a_i - a_{i,dif}$ . Los vectores  $a_{i,dif}$  pertenecen a un subespacio que en [20] y [21] es llamado *subespacio diferencia*. Si se llama  $B$  a dicho subespacio diferencia, puede escribirse un sistema de vectores que sea base de  $B$ :

$$\begin{aligned} b_1 &= a_2 - a_1 \\ b_2 &= a_3 - a_1 \\ &\vdots \\ b_{m-1} &= a_m - a_1 \end{aligned} \tag{3.7}$$

Se sugiere en [20] utilizar dicho conjunto para encontrar, según el método de Gram-Schmidt, una base  $z_1, z_2 \dots z_{m-1}$  ortonormal del llamado subespacio diferencia.

Si se llama  $a_{com}$  al vector común, cada uno de los vectores puede escribirse:

$$a_i = a_{i,dif} + a_{com} + \epsilon_i \tag{3.8}$$

Donde  $\epsilon_i$  es un término de error. Si todos los vectores error  $\epsilon_i (i = 1, 2, \dots, m)$  se asumen nulos, existen  $m$  ecuaciones con  $m + 1$  vectores desconocidos. Existen infinitas soluciones. Para obtener una única solución para el vector común, se puede asumir que los vectores  $a_{i,dif}$  son las proyecciones de los vectores de características en el subespacio diferencia  $B$ .

$$a_{i,dif} = \langle a_i, z_1 \rangle z_1 + \langle a_i, z_2 \rangle z_2 + \dots \langle a_i, z_n \rangle z_n \tag{3.9}$$

La figura 3.2 muestra un ejemplo de este método para dos dimensiones. Una de las clases ( $C^1$ ) tiene los vectores  $a_1^1$  y  $a_2^1$  y la otra clase ( $C^2$ ) tiene los vectores  $a_1^2$  y  $a_2^2$ . Los vectores comunes de ambas clases se muestran como  $a_{com}^1$  y  $a_{com}^2$ .

Los vectores propios correspondientes a los valores nulos de la matriz de dispersión de cada clase de cada clase tienen la misma dirección que el vector común. Si se proyectan los datos de entrenamiento en ese subespacio nulo tendrán distancia cero con el vector común. Y los datos de prueba tendrán distancia casi cero. Lo cual permite construir el clasificador.

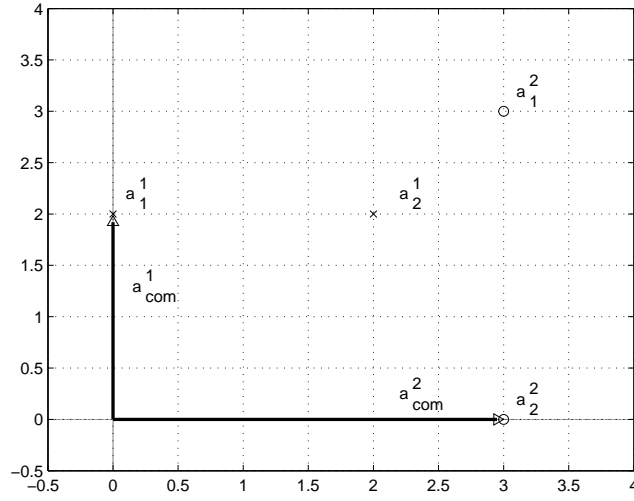


Figura 3.2: Ejemplo de clasificación con vector común en 2D. Los puntos de la clase  $C^1$  están marcados con un aspa, los de la clase  $C^2$  con un círculo.

### 3.4.2. Idea del método de los Vectores Comunes Discriminantes

En el método del vector común se calcula un único vector común para cada clase partiendo de la matriz de dispersión de dicha clase. En [22] Cevikalp, Neamtu, Wilkes y Barkana sustentan teóricamente que los vectores comunes de cada clase están en el espacio nulo de la matriz de dispersión intraclase  $S_W$  común a todas las muestras.

Pensando en datos de  $C$  clases con  $N$  muestras por clase, el método de los *vectores comunes discriminantes* o *DCV* sigue los siguientes objetivos por pasos (en esta sección la notación aparece más flexible porque lo principal que se pretende describir es la idea intuitiva):

1. Disponer los datos en una matriz  $A$  (según se verá en la ecuación 3.25). Es decir, agrupando los vectores de la misma clase. Se cumple además que la matriz de dispersión intraclase  $S_W$  equivale a  $AA^T$ .
2. El objetivo es encontrar los vectores de proyección del espacio nulo de  $S_W$ .
3. El análisis de componentes principales de  $S_W$  permite encontrar los vectores de proyección  $\alpha_k$  del espacio de rango de  $S_W$ . Son los correspondientes a los autovalores no nulos.
4. Considérese  $V$  dicho espacio de rango y  $V^\perp$  el espacio nulo. Recuérdese que su intersección es nula y por lo tanto un vector  $x$  de  $\mathbb{R}^d$  puede descomponerse en la suma de  $y \in V$  y  $z \in V^\perp$ .
5. Por lo tanto, los vectores de proyección del espacio nulo se pueden calcular restando  $z = x - y$ . Los vectores  $y$  se calculan proyectando los vectores  $x$  con los  $\alpha_k$ . Los vectores comunes son combinación lineal de los  $z$ .
6. Una vez obtenida la proyección sobre el espacio nulo de  $S_W$ , encontrar la proyección de máxima dispersión en dicho espacio permite la clasificación.

**Ejemplo en 3d**

Supongamos cuatro puntos en el espacio tal como muestra la figura 3.3. Se han marcado los puntos  $a_i$  de la misma clase con una estrella y los puntos  $b_i$  de la misma clase con un círculo, para intentar ahora, utilizar sobre ellos el método del vector común discriminante, *DCV*. En este ejemplo se calcularán los vectores propios de forma directa en cada caso, como la dimensión es pequeña puede hacerse directamente.

$$a_1 = \begin{pmatrix} -1 \\ -1 \\ 0 \end{pmatrix} a_2 = \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} b_1 = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix} b_2 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \quad (3.10)$$

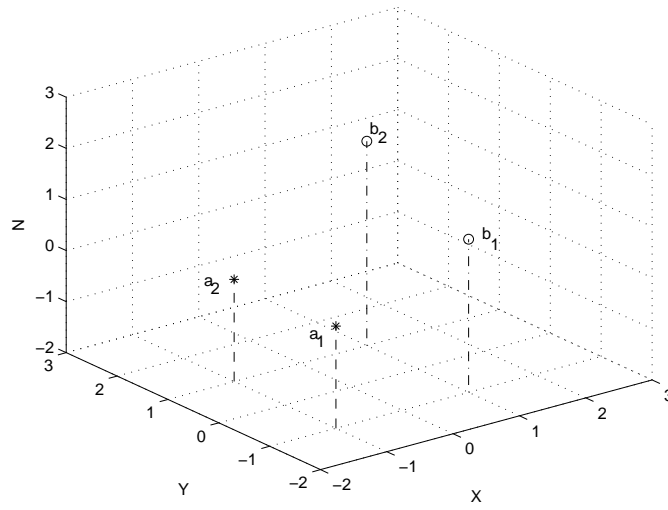


Figura 3.3: Ejemplo para el método DCV en 3d.

Se organizan los datos tal como describe el método *DCV*. Agrupando las muestras que pertenecen a la misma clase.

$$\begin{pmatrix} -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix} \quad (3.11)$$

Se resta la media por clases para construir la matriz  $A$ . De forma que  $S_W = A \times A^T$ .

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 1 & -1 & 1 \\ 0 & 0 & -1/2 & 1/2 \end{pmatrix} \quad (3.12)$$

Se calculan los vectores propios de  $S_W$  y se normalizan,  $S_W$  resulta tener un único valor propio nulo que es el primero. Colocamos en  $Q$  por lo tanto el segundo y el tercer vector propio y al

calcular  $P = Q \times Q^T$  resulta que:

$$P = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3.13)$$

Lo cual significa que el subespacio de rango de  $S_W$  está en el plano  $(Y, Z)$ , o lo que es lo mismo, las diferencias dentro de las clases están en dicho plano. Su complemento ortonormal (la línea recta  $X$ ) es el subespacio nulo, es decir, allí donde no hay diferencias dentro de las clases. Se pueden calcular los vectores comunes bien proyectando los datos en la recta o bien restándoles su proyección con  $P$ , es equivalente.

$$\begin{aligned} a_{com1} &= a_1 - P \times a_1 = (-1, 0, 0) \\ a_{com2} &= a_2 - P \times a_2 = (-1, 0, 0) \\ b_{com1} &= b_1 - P \times a_1 = (1, 0, 0) \\ b_{com2} &= b_2 - P \times b_2 = (1, 0, 0) \end{aligned} \quad (3.14)$$

El método no propone calcularlos todos (aunque aquí se haya hecho) sino hacerlo una sola vez con una muestra de entrenamiento por cada clase.

$$A_{com} = \begin{pmatrix} -1 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \quad (3.15)$$

En este ejemplo resulta trivial el análisis de componentes principales de dicha matriz para proyectar los datos de entrenamiento en un subespacio de máxima dispersión. En cualquier caso,  $S_{com} = A_{com} \times A_{com}^T$  cuyo análisis por componentes principales da un único autovalor no nulo cuyo autovector es  $(100)$ , que es el que se muestra en la figura 3.4 (el otro es el opuesto). En efecto, la matriz de proyección  $w$  de  $DCV$  para este caso estaría compuesta de un único vector. Al utilizarla sobre  $a_1$  y  $a_2$  lleva a dichos puntos sobre el *vector discriminante*  $(-100)$  mientras que al proyectar los puntos de la otra clase el resultado es  $(100)$ . Para clasificar cualquier otro punto de prueba bastaría calcular  $w' \times prueba$  y calcular la distancia a cada uno de los vectores discriminantes antedichos.

### 3.4.3. Relación con PCA

Como las covarianzas caracterizan las variaciones de los vectores sobre su media, mientras  $m \leq n$ , los valores propios no nulos de la matriz de covarianza  $S$  corresponden a los vectores que forman una base ortonormal del subespacio de diferencia  $B$ . El complemento ortogonal  $B^\perp$  en este caso es generado por todos los vectores propios correspondientes a los valores propios nulos. Como el vector común es ortogonal a cualquier vector del subespacio diferencia, el vector común debe ser una combinación lineal de los vectores propios correspondientes a los valores propios nulos. Por otro lado, el vector común puede ser también extraído restando todas las proyecciones de un vector de características sobre los vectores propios correspondientes a los valores propios no nulos.

Los valores propios de la matriz de covarianza  $S$  son no negativos y pueden ser ordenados de mayor a menor:  $\lambda_1, \lambda_2 \dots \lambda_n$ . Sean  $u_1, u_2 \dots u_n$  los correspondientes vectores propios. Sea  $Ker(S)$  el núcleo de la matriz de covarianza y  $B^\perp$  el complemento ortogonal del subespacio diferencia  $B$ .

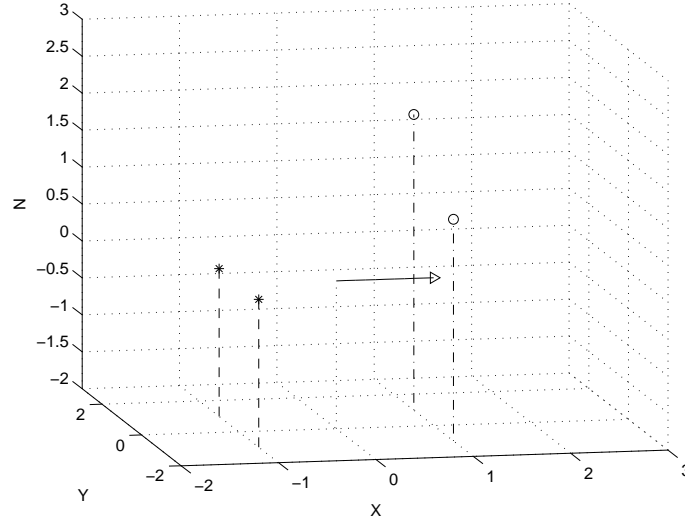


Figura 3.4: El vector común a la clase de los puntos marcados con un círculo.

Se sugieren en [21] las siguientes ecuaciones:

$$\begin{aligned} \text{Ker}(S) &= \{x \in R^{n \times 1} : Sx = 0\} \\ B^\perp &= \{x \in R^{n \times 1} : \langle x, b \rangle = 0 \quad \forall b \in B\} \end{aligned} \quad (3.16)$$

El núcleo de  $S$  es el espacio de todos los vectores propios correspondientes a los valores propios nulos de  $S$ . Aún más, como el espacio de  $B$  tiene de dimensión  $(m - 1)$  el espacio de  $B^\perp$  tiene dimensión  $(n - m + 1)$ . Además, también [21] se plantea una demostración teórica para los siguientes teoremas:

$$\begin{aligned} B^\perp &\subset \text{Ker}(S) \\ \text{Ker}(S) &\subset B^\perp \end{aligned} \quad (3.17)$$

Concluyendo que:

$$\text{Ker}(S) = B^\perp \quad (3.18)$$

De esta igualdad y recordando que  $B^\perp$  tiene dimensión  $(n - m + 1)$  se deduce que los últimos  $(n - m + 1)$  vectores propios del conjunto  $\{u_1, u_2 \dots u_n\}$  corresponden a los valores propios nulos. Por tanto  $\{u_m, u_{m+1} \dots u_n\}$  es un sistema generador del complemento ortogonal de  $B$ . Manteniendo la descomposición para un vector de características  $a_i$ :

$$\begin{aligned} a_i &= a_i^* + a_i^\perp \\ a_i^* &\in B \\ a_i^\perp &\in B^\perp \end{aligned} \quad (3.19)$$

Y la siguiente para  $a_{com}$ :

$$a_{com} = a_i - a_i^* \quad (3.20)$$

### 3.4.4. Cálculo de los vectores comunes discriminantes

Sea un conjunto de entrenamiento compuesto de  $C$  clases donde cada clase contiene  $N$  muestras. Sea  $x_m^i$  un vector columna  $d$ -dimensional que denota la muestra  $m$ -ésima de la clase  $i$ -ésima. Existe un total de  $M = NC$  muestras en el conjunto de entrenamiento. Supóngase  $d > M - C$ . Arreglo a esta nomenclatura y a lo escrito en la sección 2.5 escriben en [22] Cevikalp, Neamtu, Wilkes y Barkana utilizan las definiciones de las matrices de dispersión total, intraclase e interclase:

$$\begin{aligned} S_T &= \sum_{i=1}^C \sum_{m=1}^N (x_m^i - \mu) (x_m^i - \mu)^T = S_W + S_B \\ S_W &= \sum_{i=1}^C \sum_{m=1}^N (x_m^i - \mu_i) (x_m^i - \mu_i)^T \\ S_B &= \sum_{i=1}^C N (\mu_i - \mu) (\mu_i - \mu)^T \end{aligned} \quad (3.21)$$

Como la matriz de dispersión total, intraclase e interclase respectivamente. Siendo  $\mu$  la media de todas las muestras y  $\mu_i$  la media de las muestras de la clase  $i$ -ésima.

Para encontrar los vectores de la proyección  $w$  óptima en el espacio nulo de  $S_W$  se proyectan las muestras en el espacio nulo de  $S_W$  para lo cual es necesario obtener los vectores de proyección utilizando los vectores propios. Así, se deben calcular primero los vectores que generan el espacio nulo de  $S_W$ . Esta tarea es computacionalmente intratable ya que la dimensión del espacio nulo puede ser muy grande. Una manera más eficiente es utilizar el complemento ortogonal del *espacio nulo* de  $S_W$ , que típicamente es un espacio de dimensión más baja.

Si  $\mathbb{R}^d$  es el espacio de muestras original,  $V$  es el *espacio de rango* de  $S_W$  y  $V^\perp$  es el espacio nulo de  $S_W$ , con las ecuaciones:

$$\begin{aligned} V &= S \{ \alpha_k \mid S_W \alpha_k \neq 0 \quad k = 1 \dots r \} \\ V^\perp &= S \{ \alpha_k \mid S_W \alpha_k = 0 \quad k = r + 1 \dots d \} \end{aligned} \quad (3.22)$$

donde  $r < d$  es el rango de  $S_W$ ,  $\{\alpha_1 \dots \alpha_d\}$  es un conjunto ortonormal y  $\{\alpha_1 \dots \alpha_r\}$  es el conjunto de vectores propios ortonormales correspondientes a los valores propios no nulos de  $S_W$ . Se consideran, pues, las matrices  $Q = \{\alpha_1 \dots \alpha_r\}$  y  $\bar{Q} = \{\alpha_{r+1} \dots \alpha_d\}$ . Como  $\mathbb{R}^d = V \oplus V^\perp$  cualquier vector  $x_m^i \in \mathbb{R}^d$  tiene una única descomposición:

$$x_m^i = y_m^i + z_m^i \quad (3.23)$$

donde  $y_m^i = Px_m^i = QQ^T x_m^i \in V$ ,  $z_m^i = \bar{P}x_m^i = \bar{Q}\bar{Q}^T x_m^i \in V^\perp$  y  $P$  y  $\bar{P}$  son los operadores de proyección ortogonal sobre  $V$  y  $V^\perp$ , respectivamente. Las ecuaciones 3.19 y 3.23 son en realidad la misma expresión. El *vector común* –en concordancia con la definición dada en la sección 3.4.1– es  $z_m^i$ , es el primer objetivo de cálculo, que se resuelve según:

$$z_m^i = x_m^i - y_m^i = x_m^i - Px_m^i \quad (3.24)$$

Lo cual necesita conocer una base de  $V$  que puede ser calculada con los vectores propios de la matriz  $S_W$ . En particular los vectores propios normalizados  $\alpha_k$  correspondientes a los valores propios no

nulos de  $S_W$  serán una base ortonormal de  $V$ . Los vectores propios pueden ser obtenidos calculando los vectores propios de la matriz  $A^T A$  más pequeña ( $M \times M$ ), donde se define  $S_W = AA^T$  y en  $A$  se colocan agrupadamente los vectores de la misma clase, con lo que  $A$  tendrá la forma:

$$A = [x_1^1 - \mu_1 \dots x_N^1 - \mu_1 \quad x_1^2 - \mu_2 \dots x_N^C - \mu_C] \quad (3.25)$$

Sean  $\lambda_k$  y  $v_k$  el  $k$ -ésimo valor propio y su correspondiente vector propio de  $A^T A$ , donde  $k \leq M - C$ . Entonces,  $\alpha_k = Av_k$  será el vector propio correspondiente al  $k$ -ésimo valor propio de  $S_W$ . La proyección buscada sobre  $V^\perp$  se archiva utilizando 3.24. Así, se obtiene el mismo único vector para todas las muestras de la misma clase:

$$\begin{aligned} x_{com}^i &= x_m^i - QQ^T x_m^i = \bar{Q} \bar{Q}^T x_m^i \\ m &= 1 \dots N \\ i &= 1 \dots C \end{aligned} \quad (3.26)$$

Es decir, el vector resultante de 3.26 es independiente del índice  $m$ . En [22] Cevikalp, Neamtu, Wilkes y Barkana proponen una demostración matemática para justificar 3.26. Después de obtener los vectores comunes  $x_{com}^i$  la técnica de Cevikalp, Neamtu, Wilkes y Barkana propone optimizar los vectores de proyección maximizando la dispersión total de los vectores comunes, buscando la  $W$  que maximice

$$J(W_{opt}) = |W^T S_{com} W| \quad (3.27)$$

donde  $W$  es una matriz cuyas columnas son los vectores  $w_k$  y  $S_{com}$  es la matriz de dispersión de los vectores comunes. Todo ello lleva –de nuevo– a aplicar PCA, esta vez a  $S_{com}$ . Como  $S_{com}$  puede ser grande (es una matriz  $d \times d$ ) se aplica la modificación de *Eigenfaces* utilizando la matriz  $A_{com}^T A_{com}$  de tamaño  $C \times C$ , para encontrar los valores propios y los correspondientes vectores propios de  $S_{com} = A_{com} A_{com}^T$ , siendo  $A_{com}$  una matriz  $d \times C$  de la forma:

$$A_{com} = [x_{com}^1 - \mu_{com} \dots x_{com}^C - \mu_{com}] \quad (3.28)$$

Habrán  $C - 1$  vectores de proyección (ya que el rango de  $S_{com}$  es  $C - 1$ ) si todos los vectores son linealmente independientes. Si dos vectores comunes son idénticos las dos clases representadas por dicho vector no pueden ser distinguidas. Como los vectores de proyección óptima  $w_k$  pertenecen al espacio nulo de  $S_W$ , cuando los vectores de muestra  $x_m^i$  de la clase  $i$ -ésima son proyectados en el espacio generado por los vectores  $w_k$ , el vector de características  $\Omega_i = [\langle x_m^1, w_1 \rangle \dots \langle x_m^i, w_{C-1} \rangle]$  de los coeficientes de proyección  $\langle x_m^i, w_k \rangle$  serán independientes del índice de la muestra  $m$ . Teniéndose

$$\Omega_i = W^T x_m^i \quad m = 1 \dots N, \quad i = 1 \dots C \quad (3.29)$$

Los autores de esta técnica llaman en [22] *vectores discriminantes comunes* a los vectores de características  $\Omega_i$  y son los utilizados para la clasificación de imágenes. El hecho de que  $\Omega_i$  no dependa del índice  $m$  garantiza la exactitud a la hora de reconocer las muestras del conjunto de entrenamiento. Para hacer el reconocimiento Cevikalp, Neamtu, Wilkes y Barkana proponen utilizar como matriz de proyección  $W$  y comparar con los *vectores comunes discriminantes* utilizando la distancia euclídea. De manera que para una imagen de prueba  $x_{test}$  obtienen su respectiva  $\Omega_{test}$ :

$$\Omega_{test} = W^T x_{test} \quad (3.30)$$

Comparándose  $\Omega_{test}$  con los vectores discriminantes comunes  $\Omega_i$ , siendo el  $i$ -ésimo más próximo el que sirve para hacer la identificación. Al comparar  $\Omega_{test}$  sólo con un único vector por cada clase se presentan ventajas según [22] para tareas de reconocimiento en *tiempo real*. Otros métodos hacen esa comparación con todos los vectores de muestra del conjunto de entrenamiento.

### 3.4.5. Sumario

Se resumen los siguientes pasos para concebir esta técnica como un algoritmo:

1. Calcular los valores propios no nulos (y sus vectores propios) de  $S_W$  utilizando la matriz  $A^T A$ . Agruparlos en la matriz  $Q$ . Construir con ella la matriz de proyección  $P = QQ^T$ .
2. Elegir cualquier vector de la muestra y proyectarlo en el espacio nulo de  $S_W$  para obtener los vectores comunes. Expresión 3.26. Construir con ellos su matriz de dispersión  $S_{com}$ .
3. Calcular componentes principales con los vectores propios  $w_k$  de  $S_{com}$  correspondientes a los valores propios no nulos utilizando la matriz  $A_{com}^T A_{com}$ .
4. Existen como mucho  $C - 1$  vectores propios que corresponden a los valores propios no nulos. Utilizar dichos vectores para construir la matriz de proyección  $W$  que servirá para obtener los vectores de características.
5. Elegir (de nuevo) un vector de características por cada clase y proyectarlo con  $W$ , el resultado será un vector de clasificación, es decir, la proximidad con él nos dirá la pertenencia o no a la clase.
6. Para proyectar en el espacio de los vectores de características finales habrá que componer la primera transformación  $P$  con la segunda  $w$  en una sola matriz de proyección  $W$  que se utilizará para proyectar los vectores de prueba en el espacio de características.





## Capítulo 4

# Experimentos y resultados

### 4.1. Base de datos de caras de AT&T

Existen más de una veintena de bases de datos estándar con imágenes de caras para probar cualquier algoritmo. Sólo hablando de las recomendadas por <http://www.face-rec.org>. El acceso a ellas (no a todas) es más o menos público y podemos encontrar referencias en [22] y otros trabajos al uso de bases de datos de imágenes de caras como la de Yale o la de AR (de entre las fácilmente conseguibles, las más grandes).

Los resultados que se muestran en este capítulo se han obtenido escribiendo los métodos sobre ©Matlab y haciendo uso de la base de datos de AT&T. Se ha elegido ésta por tener un número de caras suficiente y por tener las caras centradas en la foto. En otras bases de datos esto no es así, por lo que se requiere un preprocesado de la imagen donde se «encuentre» y se seleccione la parte donde existe la cara. Como se ha visto durante la sección 1.4 existen muchos trabajos que afrontan este problema y se puede asumir para afrontar el reconocimiento, pero debe ser estudiado por separado.

En la base de datos de AT&T existen 400 imágenes tomadas en laboratorio entre 1992 y 1994. Son imágenes de 40 individuos diferentes, a los cuales se les tomó 10 fotos en cada sesión con variaciones de luz, expresión facial, punto de vista y (en algunos casos) presencia o no de gafas. En la figura 4.1 pueden verse las fotos tomadas a 3 de esos 40 individuos. Para los experimentos cuyos resultados se muestran a continuación se han considerado, este tipo de imágenes, ya suficientemente encuadradas. La dimensión de las fotos es de 80 de alto por 70 de ancho. De manera que los vectores-imagen tendrán una dimensión de 5600.

### 4.2. Leaving one out

#### 4.2.1. Preparación

Cada método genera una matriz de proyección simplificando el problema de reconocer imágenes en  $\mathbb{R}^D$  para una dimensión  $D$  muy grande, llevándolo a  $\mathbb{R}^d$  con  $d \ll D$ . Una vez proyectadas las imágenes de caras al espacio  $\mathbb{R}^d$  se la clasificación puede resolverse de varias maneras.



Figura 4.1: Extracto de la base de datos de imágenes de caras de AT&amp;T

Las implementaciones de *Eigenface* y *Discriminative Common Vector* (o DCV) se han resuelto de manera que la fase de entrenamiento organizara en una matriz  $\Omega$  los vectores de características de cada clase. Un vector por cada clase, cuya proximidad para la imagen de prueba ha de entenderse como la identificación a esa clase.

Por otro lado, la implementación de *Fisherface* se limita a proyectar todas las imágenes de entrenamiento al espacio de llegada  $\mathbb{R}^d$ . Para la clasificación se ha utilizado un promediado de distancias. Es decir se ha clasificado en la clase  $i$  aquella imagen que más cerca (en promedio) estaba de las  $N - 1$  imágenes de entrenamiento. Siendo  $N$  el número de imágenes por cada clase, en el caso de la base de datos de AT&T equivale a  $N = 10$ .

#### 4.2.2. Resultados

Considerando  $C$  el número de clases, en este caso el número de individuos diferentes, han sido lanzados los tres métodos *Eigenface*, *Fisherface* y *Discriminative Common Vector* con diferentes valores para  $C$ .

Para cada valor de  $C$  se realizó un experimento diferente por cada método. Al conjunto de entrenamiento se le extrajo una de las caras por cada clase que luego fue utilizada para probarlo, gastando cada método para reconocer las  $C$  caras extraídas (una por clase) durante la fase de entrenamiento. Primero se extrajo la primera cara de cada clase, luego la segunda y así sucesivamente hasta haber probado el entrenamiento y la prueba sin cada una de las diez fotos que tiene cada individuo. Los resultados (tasa de aciertos) promediados se presentan en la tabla 4.1 y son los que se han utilizado para generar la figura 4.2.

$C=$	5	10	15	20	25	30	35	40
<b>Eigenface</b>	0.9800	0.9600	0.9667	0.8950	0.9000	0.8833	0.8371	0.7675
<b>Fisherface</b>	0.9600	0.9800	0.9867	0.9650	0.9880	0.9900	0.9829	0.9825
<b>DCV</b>	1.0000	0.9900	0.9933	1.0000	0.9920	0.9867	0.9943	0.9875

Cuadro 4.1: Leaving one out para distintos valores de  $C$

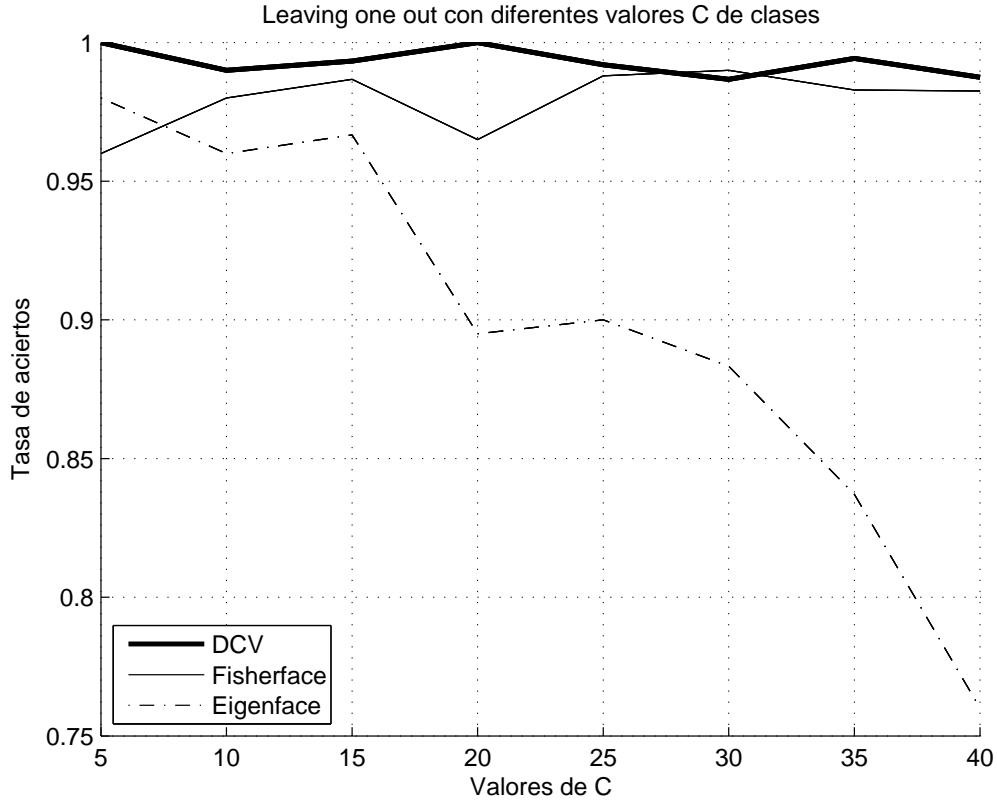


Figura 4.2: Evolución del comportamiento de los tres métodos arreglo al número de caras de entrenamiento

### 4.2.3. Dimensión en el espacio de llegada

#### Eigenface

Como se ha explicado en anteriormente (página 26) el análisis de los componentes principales es un algoritmo que reduce la dimensión de los datos. Sin embargo, esta reducción puede hacerse de varias maneras, incluso puede fijarse tomando un número de vectores propios que parezca apropiado. Para no abandonar esta elección a la arbitrariedad se han hecho varias pruebas cuyos resultados se presentan en la tabla 4.2. Se han probado varias cotas de varianza en relación a la suma acumulada de los autovalores. Cuanto más porcentaje de varianza total tomamos, más autovalores y por lo tanto más autovectores. La matriz de proyección es mayor y el subespacio de llegada de dicha proyección tiene mayor dimensión. Conforme mayor sea la dimensión mejor se reconocerá, pero a partir de un cierto valor, la magnitud de los autovalores expresará la poca información que aportan los vectores asociados, y por lo tanto la tasa de reconocimiento no se verá aumentada.

Estos resultados (que también se pueden ver en forma de gráfica en la figura 4.3) han sido también promediados ya que por cada clase se hacen 10 experimentos diferentes quitando cada

<b>Porcentaje varianza</b>	0.65	0.70	0.75	0.80	0.85	0.90	0.95
<b>Dimensión</b>	11	16	22.5	33.3	50.5	81.4	144.6
<b>Tasa de aciertos</b>	0.7325	0.7500	0.7625	0.7675			

Cuadro 4.2: Dimensión del espacio de llegada para Eigenface.

una de las 10 fotos de cada uno de los cuarenta individuos. Esto explica que haya valores para la dimensión con decimales, se debe a que incluso con el mismo porcentaje según los experimentos, la dimensión de llegada ha podido variar.

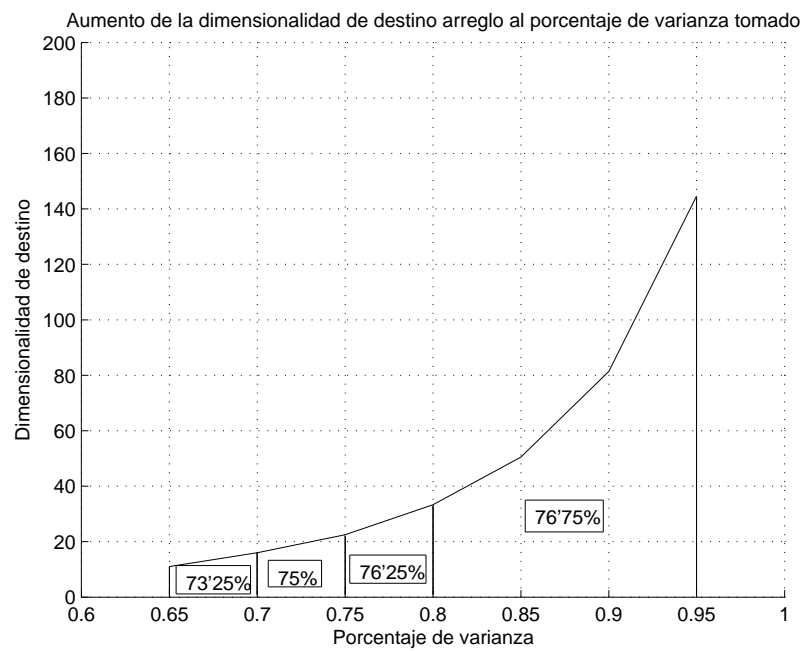


Figura 4.3: Cambio de dimensión y tasa de aciertos según porcentaje de varianza.

Los resultados mostrados en el cuadro 4.1 han sido calculados con un porcentaje del 95 % de la varianza. Sin embargo en el cuadro 4.3 pueden compararse con los obtenidos con el 80 % valor a partir del cual se «estabiliza» la tasa de éxitos con  $C = 40$ .

<b>C=</b>	5	10	15	20	25	30	35	40
<b>0.95</b>	0.9800	0.9600	0.9667	0.8950	0.9000	0.8833	0.8371	0.7675
<b>0.80</b>	0.9800	0.9600	0.9667	0.8950	0.9000	0.8800	0.8257	0.7675

Cuadro 4.3: Variación de éxitos conforme a la dimensión del espacio de llegada.

### Fisherface y DCV

De acuerdo con [19] la dimensión del espacio de llegada para la proyección según Fisherface es de  $C - 1$ . Siendo  $C$  el número de clases. Previamente se reduce la dimensión con PCA a un espacio de  $M - C$  siendo  $N$  el número total de imágenes  $M = N \times C$  equivaliendo  $N$  al número de imágenes por individuo. En este caso, con los experimentos sobre la base de datos de AT&T resulta  $M - C = 400 - 40 = 360$ , en un espacio de  $\mathbb{R}^{360}$  se aplica entrenamiento según Fisher resultando una proyección que lleva de dicho espacio a uno de  $C - 1 = 40 - 1 = 39$  dimensiones.

De acuerdo con [22], la dimensión de llegada en DCV también se ha fijado a  $C - 1$ .

## 4.3. Ruido

### 4.3.1. Degradación de la imagen

Se presentan también los resultados de *leaving one out* con imágenes ruidosas. Aplicando ruido a la imagen de prueba. Se ha fijado el porcentaje de varianza en el método de *Eigenface* a 95 % otra vez. Se tienen dos tipos de distorsiones.

- Sal y pimienta: Variando el porcentaje de imagen que se distorsiona en el 5 %, 10 %, 15 % y 20 % como se muestra en la ilustración 4.4 (en la que también puede verse a la izquierda la imagen original).
- Gaussiana con media cero. Con distintas varianzas 0.005, 0.010, 0.015, 0.020 y 0.025, puede verse en la figura 4.5 en la columna superior.



Figura 4.4: Ruido de sal y pimienta al 5 %, 10 %, 15 % y 20 % respectivamente.



Figura 4.5: Ruido de *gaussiana* con media 0 y varianza 0.005, 0.010, 0.015, 0.020 y 0.025.

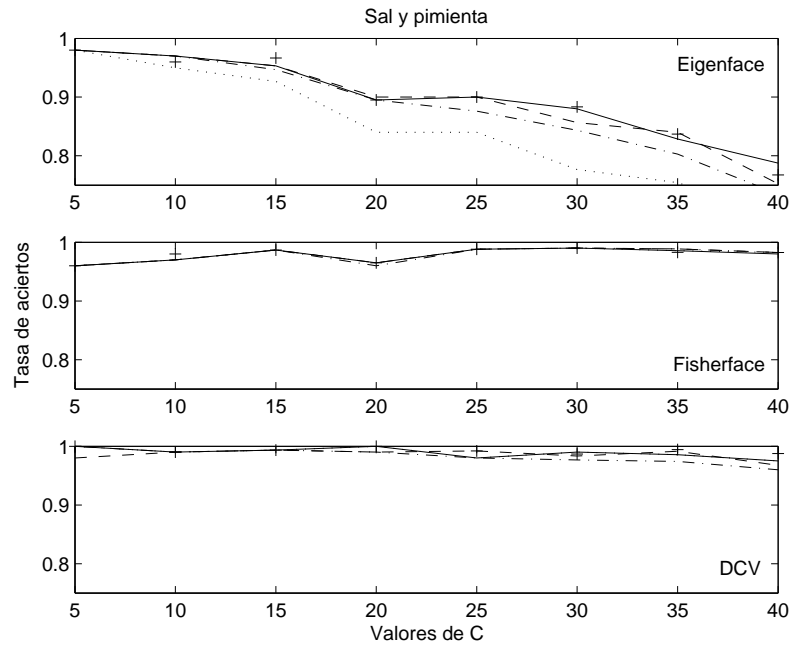


Figura 4.6: Resistencia al ruido de *sal y pimienta* de los tres métodos (de arriba a abajo) *Eigenface*, *Fisherface* y *DCV*. Las cruces marcan el comportamiento del método con imágenes sin ruido. Las líneas son el resultado según la densidad del ruido 5%, 10%, 15% y 20%. Se ha expresado en la misma degradación del trazo (desde continuo hasta punteado) la progresión.

### 4.3.2. Resultados

#### Sal y pimienta

En la figura 4.6 aparecen de arriba abajo tres gráficas, la superior corresponde a *Eigenface*, la del medio a *Fisherface* y la de abajo a *DCV*. Al igual que en la figura 4.2 el eje horizontal representa la variación para distintos números de clases. El eje vertical representa la tasa de aciertos (siendo 1 acierto total, cien por cien de aciertos).

Dichas gráficas muestran la resistencia de los métodos al ruido de *sal y pimienta* como se ha descrito anteriormente. Se ha marcado con cruces el comportamiento de los métodos para imágenes

sin ruido. Y se ha expresado en el mismo trazo de la gráfica el nivel de degradación al que fueron sometidas las imágenes de prueba. Así, el trazo continuo representa imágenes con poco ruido (un 5 % de *sal y pimienta*) y conforme el trazo es más discontinuo se está representando el comportamiento del método para imágenes de prueba con más ruido (un 10 %, un 15 % y un 20 % de ruido de *sal y pimienta*).

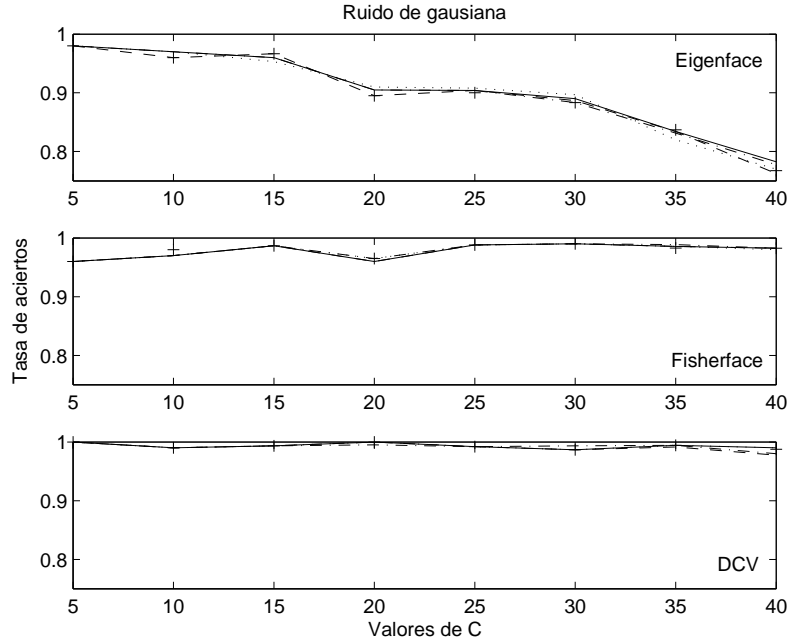


Figura 4.7: Resistencia a ruido aditivo con distribución *gaussiana*.

#### Ruido con distribución *gaussiana*

La figura 4.7 muestra (con la misma estructura y presentación que la figura 4.6) los resultados de los distintos métodos para el ruido con distribución *gaussiana* de media 0. La varianza se ha variado tal como se ha descrito anteriormente (o sea, 0.005, 0.010, 0.015, 0.020 y 0.025).

## 4.4. Conclusiones

Ha sido desarrollado, implementado, probado y comparado un clasificador global para un problema de alta dimensionalidad. Utilizando muestras de bases de datos estándar (reconocidamente) para la resolución del problema de identificación de imágenes de rostros humanos. El objetivo es caracterizar su eficacia en problemas simulados y su posible aplicación práctica.

Dentro de los métodos de reconocimiento de caras se ha trabajado en el campo del análisis estadístico multivariante y los subespacios vectoriales, tomando uno de los métodos más recientes, y siguiendo en [22] a sus autores para cotejarlo con otros anteriores.



Se han hecho pruebas de clasificación con 10 muestras de cada clase, con 40 clases. Entrenando al clasificador con 9 por clase y probando la restante. Repitiendo para cada clase el experimento 10 veces, una con cada muestra. Los resultados de estas pruebas de *leaving one out* muestran mejores tasas de acierto para el método de los *vectores comunes discriminantes* (DCV). La implementación de los métodos anteriores ha sido depurada y ajustada buscando superar a *DCV*.

Con la misma fase de entrenamiento para el clasificador, se ha aplicado ruido a las muestras de prueba en un segundo experimento. Se ha aplicado ruido de *sal y pimienta* y ruido con distribución *gaussiana* a las imágenes. Los mejores resultados de este experimento los muestra el método de *Fisherfaces*.

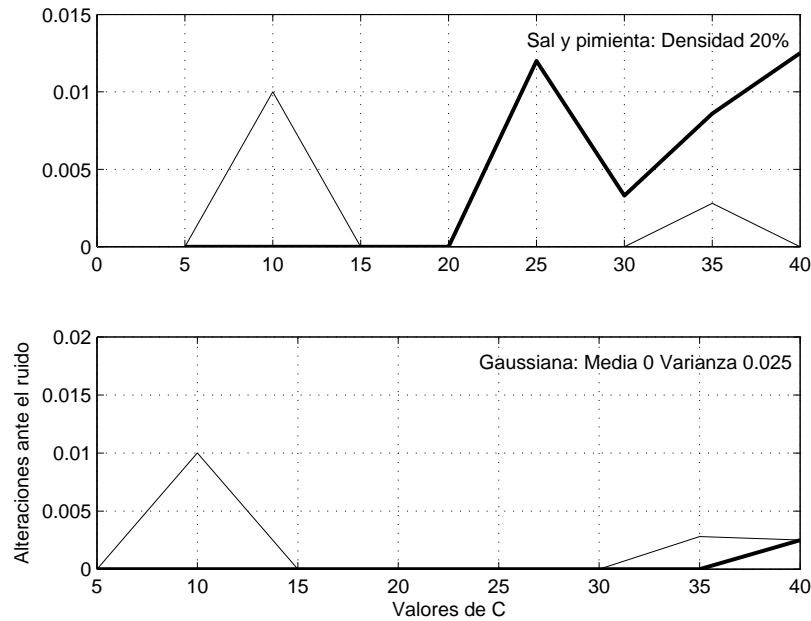


Figura 4.8: Alteración ante el ruido en el método de Fisherface y en el de DCV. El eje horizontal varía con el número de muestras. Se ha fijado los resultados para ruido de *sal y pimienta* con un 20 % de densidad y para ruido con distribución *gaussiana* con 0 de media 0 y 0.025 de varianza. **Los resultados del método de DCV se muestran en trazo grueso.**

El ruido de *sal y pimienta* ha alterado especialmente los resultados del método de *Eigenfaces* y también (aunque mucho menos) los de *DCV*. El aditivo de *gaussiana* ha resultado ser más benigno para este tipo de experimentos. Aunque el error de *DCV* para ruido de *gaussiana* es comparable o incluso menor al del método de las *Fisherfaces*, lo cierto es que el método de *Fisherfaces* ha resultado más robusto para este tipo de situaciones.

Queda pues para trabajos futuros estudiar la razón de estas respuestas con más detalle. Pues si bien es lógico (y de acuerdo con la teoría) que *Eigenface* se muestre peor en general y ante este tipo de condiciones, la mejoría de *DCV* sobre *Fisherfaces* en el primer *leaving one out* prometía mejores comportamientos ante condiciones ruidosas.

No se ha afrontado en ningún momento la implementación desde el punto de vista de la efi-

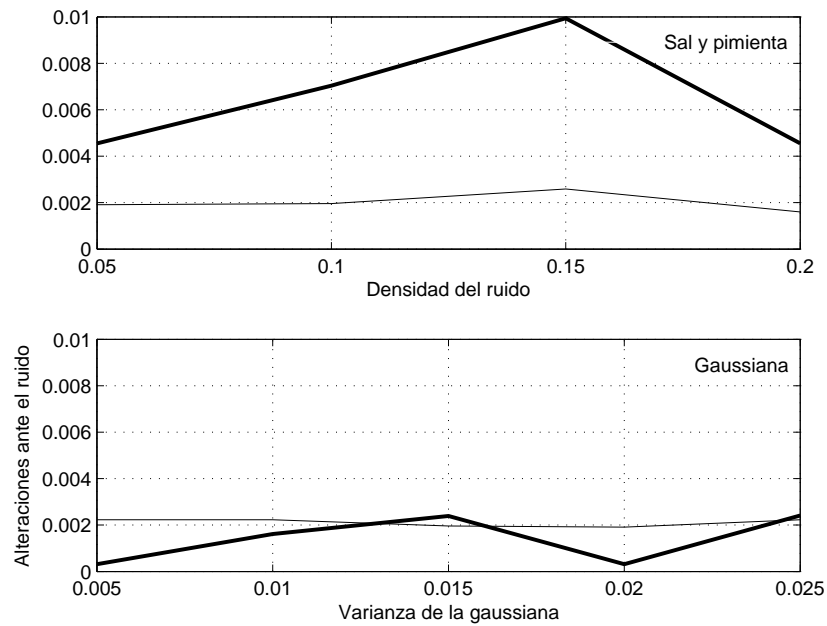


Figura 4.9: Se ha promediado las alteraciones ante el ruido (el error producido) para todos los valores del número de clases  $C$ . El eje horizontal varía según la cantidad de ruido añadida. **Los resultados del método de  $DCV$  se muestran en trazo grueso.**

ciencia, es decir, no se han marcado objetivos de coste temporal ni se ha buscado optimizar los algoritmos ni el código. Diseñar una aplicación real atendiendo a estas cuestiones es una interesante posibilidad para un futuro inmediato.

Se ha establecido un punto de partida para posibles desarrollos futuros en la línea de extender los métodos del *vector común* y *vectores comunes discriminantes*. Tanto para que sean capaces de generalizar mejor como en contraste con otros métodos basados en subespacios que en este trabajo no se han probado así como en otro tipo de problemas de clasificación con dimensionalidad alta o muy alta.



# Bibliografía

- [1] W. Zhao, R. Chellappa, A. Rosenfeld and P.J. Phillips *Face recognition: A literature survey* ACM Computing Surveys, 2003
- [2] Xiaoguang Lu *Image analysis for face recognition* personal notes, 2003
- [3] Xiaoguang Lu, Anil K. Jain and Dirk Colbry *Matching 2.5D Scans for Face Recognition* 2005, ©IEEE Computer Society
- [4] Alessandro Colombo, Claudio Cusano and Raimondo Schettini *3D face detection using curvature analysis* Copyright ©2005 Pattern Recognition Society Published by Elsevier B.V.
- [5] Matthew Escarra, Marshall Robinson, Jon Krueger and Doug Kochelek *Face Recognition Background* Colección Conexions 2004 [www.cnx.org](http://www.cnx.org)
- [6] A.J. Goldstein, L.D. Harmon and A.B. Lesk *Identification of human faces* Proc. IEEE Vol 59. No. 5 (748-760) May 1971
- [7] William A. Barret *A survey of Face Recognition Algorithms and Testing Results* 2000 National Biometrics Test Center Collected works
- [8] Erik Hjelmås and Boon Kee Low *Face detection: A survey* Copyright ©2001 Academy Press
- [9] Valery Starovoitov and Dmitry Bryliuk *Access control by face recognition using neural networks and negative examples* 2002 The 2nd international conference of Artificial Intelligence
- [10] Chakka Siva Sai Prasanna, N. Sudha and V. Kamakoti *A principal component neural network-based face recognition system and ASIC implementation* VLSI Design, 2005. 18th International Conference
- [11] Andrew R. Webb *Statistical Pattern Recognition* second ed. Copyright © 2002 John Wiley & Sons, Ltd.
- [12] Brian S. Everitt and Graham Dunn *Applied multivariate data analysis*, second ed. Copyright © 2001 Arnold Publishers.
- [13] Daniel Peña *Análisis de datos*, primera ed. Copyright © 2002 McGraw-Hill Interamericana de España, S.A.U.
- [14] Juan De Burgos *Curso de álgebra lineal y geometría*, Copyright ©1994 Alhambra Longman
- [15] Juan De Burgos *Álgebra lineal y geometría cartesiana*, Copyright ©2000 McGraw Hill

- [16] L. Sirovich and M. Kirby (1987) *Low-dimensional procedure for the characterization of human faces* Journal of the Optical Society of America A, 4(3), 519-524
- [17] Matthew Turk and Alex Pentland *Eigenfaces for Recognition*, Copyright ©1991 Massachusetts Institute of Technology, (*Journal of Cognitive Neuroscience*)
- [18] Peter N. Belhumeur, João P. Hespanha and David J. Kriegman *Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection*, Copyright ©1996 Computer Conference on Computer Vision
- [19] Kamran Etemad and Rama Chellappa *Discriminant analysis for recognition of human face images*, Copyright ©1997 Optical Society of America
- [20] M. Bilginer Gülmezoğlu, Vakıf Dzhafarov, Mustafa Keskin and Atalay Barkana *A novel approach to isolated word recognition* Copyright ©1999 IEEE
- [21] M. Bilginer Gülmezoğlu, Vakıf Dzhafarov and Atalay Barkana *The common vector approach and its relation to principal component analysis* Copyright ©2001 IEEE
- [22] Hakan Cevikalp, Marian Neamtu, Mitch Wilkes and Atalay Barkana *Discriminative common vectors for face recognition* Copyright ©2005 IEEE Computer Society