

# **DISEÑO DE UN SISTEMA BIOMETRICO DE RECONOCIMIENTO FACIAL EN TIEMPO REAL**

Cristian Bertino Salazar

Sebastián Orozco Alzate

Tecnología Mecatrónica  
Facultad de Tecnologías  
Universidad Tecnológica de Pereira  
Colombia



## *Dedicado a*

Nuestras familias por la confianza depositada y el apoyo incondicional durante el proceso de desarrollo del proyecto.



## *Agradecimientos*

De antemano agradecemos a nuestro director de proyecto Ing. Deivy Alejandro Muñoz Acosta, por confiar en nosotros y permitirnos trabajar con él, aportándonos su conocimiento y consejos en el tema. También agradecemos a todas las personas que nos han apoyado en el proceso y han hecho que este proyecto sea posible.



## Resumen

A medida que el mundo evoluciona, la humanidad se ve en la necesidad de optimizar la forma en que protege sus empresas o industrias. La biometría informática, es una rama de la biometría, que nos permite restringir el acceso a zonas privadas de las empresas o industrias. Con el reconocimiento facial, podemos realizar la identificación y reconocimiento de personas, permitiendo elegir que personas tienen acceso a ciertas zonas. Este proyecto estudia de manera general algunos de los tipos de reconocimiento biométricos del rostro. Los métodos seleccionados, estudiados y puestos a prueba son: la técnica de Análisis de Componentes Principales (PCA) y la Transformada Discreta del Coseno (DCT). Por otra parte, se tienen como estudio de los pasos para el reconocimiento, las técnicas LPP y EBGm. Para las técnicas puestas a prueba, se toman dos bases de datos con imágenes del rostro de diferentes personas a las cuales se entrenan dos imágenes y se ponen a prueba 4 imágenes en diferentes posiciones y gestos. Se realizó el procedimiento y análisis pertinente para el reconocimiento de los individuos, y los resultados obtenidos por la prueba de validez de los algoritmos, se sometieron a procesos estadísticos con porcentajes de eficiencia de: PCA, con base de datos MUG 67,5% y ORL 22,5%. DCT, base de datos MUG 80% y ORL 75%.



# Introducción

Durante los últimos años, el reconocimiento facial se ha convertido en uno de los estudios con mayor importancia de la biometría, en la cual se desarrollan aplicaciones con el procesado de la imagen y reconocimiento de patrones característicos faciales. Una de las grandes razones de estos estudios y desarrollos, es la necesidad constante de aplicaciones en el campo de la seguridad y vigilancia. El creciente interés se puede ver reflejado en el aumento en el número de soluciones en la identificación, vigilancia y control de individuos en el sector comercial e industrial, mediante el reconocimiento facial. Un ejemplo de ello claro son los sistemas basados en el reconocimiento para la identificación de personas con antecedentes criminales y los sistemas de video de vigilancia.



# Índice general

<b>Resumen</b> .....	7
<b>Introducción</b> .....	9
<b>Capítulo 1</b> .....	17
1.1    Introducción al reconocimiento facial y adquisición de datos .....	17
1.1.1    Adquisición de la imagen .....	18
1.1.2    Detección del rostro .....	18
1.1.3    Acondicionamiento y normalización .....	19
1.1.4    Extracción de características .....	21
1.1.5    Algoritmo de reconocimiento .....	21
1.1.6    Base de datos .....	22
<b>Capítulo 2</b> .....	24
2.1    Estudio de técnicas existentes para la extracción de características faciales .....	24
2.1.1    Análisis de componentes principales (PCA, Principal Component Analysis) .....	24
2.1.2    Análisis lineal discriminante (LDA, Linear Discriminant Analysis) .....	28
2.1.3    Transformada discreta de coseno (DCT, Discrete Cosine Transform) .....	31
2.1.4    Preservación de proyecciones locales (LPP, Locality Preserving Projections) .....	33
2.1.5    EBGM (Elastic Bunch Graph Matching) .....	35
2.2    Clasificadores de características .....	36
2.2.1    Clasificador por vecinos más cercanos (KNN, k-nearest neighbours) .....	36
2.2.2    Clasificador de líneas de características más cercanas (NFL, nearest feature lines) .....	37
<b>Capítulo 3</b> .....	39
3.1    Pruebas en el software Matlab® .....	39
3.1.1    PCA .....	39
3.1.2    LPP .....	44
3.1.3    DCT .....	47
3.2    Bases de datos utilizadas en las pruebas .....	49
3.2.1    The MUG (Multimedia Understanding Group) .....	49
3.2.2    Base de datos Olivetti (ORL Database) .....	50

<b>Capítulo 4</b> .....	51
4.1 Resultados obtenidos .....	51
<b>Capítulo 5</b> .....	56
5.1 Presupuesto del proyecto .....	56
<b>Capítulo 6</b> .....	57
6.1 Conclusiones y recomendaciones .....	57
<b>Bibliografía</b> .....	59



# Índice de imágenes

Figura 1.1.1: Fases del reconocimiento facial. [Los autores] .....	17
Figura 1.1.2: Selección del rostro [Modificada por los autores [19]].....	18
Figura 1.1.3: Ejemplo base de datos BANCA [6]. .....	22
Figura 1.1.4: Diferencia de iluminación [Modificado por los autores de [21]].....	23
Figura 1.1.6: Rostro con vello facial. [7].....	23
Figura 2.1.1: Resultado del análisis de los componentes principales. (a) primer componente principal, (b) segundo componente principal, (c) tercer componente principal, (d) cuarto componente principal. [9].....	25
Figura 2.1.2: Componentes principales de un conjunto de puntos bidimensional. La primera componente ofrece una reducción lineal óptima de dimensión de 2D a 1D en cuanto a error cuadrático medio se refiere. [8] .....	25
Figura 2.1.3: Ejemplo de reducción dimensional al aplicar PCA. [8].....	26
Figura 2.1.4: Autovectores correspondientes a los 7 mayores autovalores mostrados como imágenes $p \times p = n$ (Obtenidos de la base de datos de caras ORL). [8] .....	27
Figura 2.1.5: Cara media de los 7 primeros autovectores (Obtenidos de la base de datos de caras ORL). [8] .....	27
Figura 2.1.6: Autovectores correspondientes a los 7 menores autovalores mostrados como imágenes $p \times p = n$ (Obtenidos de la base de datos de caras ORL). [8] .....	27
Figura 2.1.7: LDA, ejemplo en dos dimensiones. [11].....	28
Figura 2.1.8: Las 6 Primeras FisherFaces calculadas en la base de datos ORL. [8] .....	30
Figura 2.1.9: a) Imagen original (112x92); b) Coeficientes DCT (112x92). [13].....	32
Figura 2.1.10: a) PCA (b) LPP. [14].....	33
Figura 2.1.11: Representación de la estructura de los datos en el nuevo sub-espacio, ejemplo formado por 5 personas y cuatro imágenes por persona .....	34
Figura 2.1.12: Se puede notar como a diferencia de PCA, en LPP se conserva la estructura local de los datos. [8].....	34
Figura 2.2.1: Ejemplo clasificador KNN [Los autores].....	36
Figura 3.1.1: Rostro pre-procesado [Modificado por los autores [19]].....	40
Figura 3.1.2: Base de datos normalizada [Los autores].....	41
Figura 3.1.3: Matriz de Eigenfaces [Los autores] .....	42
Figura 3.1.4: Ejemplo matriz de adyacencia [8].....	45
Figura 3.1.5: Rostro pre-procesado [Modificado por los autores [19]].....	47
Figura 3.1.6: Ejemplo de aplicación DCT sobre una imagen [Los autores] .....	48
Figura 3.1.7: Ejemplo selección en zigzag de los (d) coeficientes.....	48
Figura 3.2.1: Ejemplo de algunos personajes de la base de datos MUG [Modificado por los autores [19]].....	49

Figura 3.2.2: Ejemplo base de datos ORL [20] .....	50
Figura 4.1.1: Pruebas realizadas con las diez imágenes de la persona uno de la base de datos MUG, evaluando el algoritmo PCA [Los autores] .....	51
Figura 4.1.5: Pruebas realizadas con las diez imágenes de la persona uno de la base de datos MUG, evaluando el algoritmo DCT [Los autores] .....	52
Figura 4.1.9: Pruebas realizadas con las ocho imágenes de la persona uno de la base de datos ORL, evaluando el algoritmo PCA [Los autores] .....	53
Figura 4.1.13: Pruebas realizadas con las ocho imágenes de la persona uno de la base de datos ORL, evaluando el algoritmo DCT [Los autores] .....	54
Figura 4.1.17: Tabla de porcentajes de eficiencia, algoritmos PCA y DTC [Los autores] ..	54
Figura 4.1.18: Tabla de tiempos de procesamiento para los algoritmos [Los autores] .....	55



# Capítulo 1

## 1.1 Introducción al reconocimiento facial y adquisición de datos

El reconocimiento facial es un sistema para la identificación de personas por medio de imágenes, las cuales pueden ser tomadas anteriormente o adquiridas en un sistema de tiempo real. Para el reconocimiento facial principalmente se deben de conocer algunos ítems, ya que existen variables externas que afectan exponencialmente el procesamiento de la imagen; al igual se puede realizar un proceso más efectivo dependiendo de la técnica de reconocimiento que se desea trabajar.

El reconocimiento facial tiene un proceso estándar de desarrollo el cual puede ser muy explícito en la siguiente imagen.

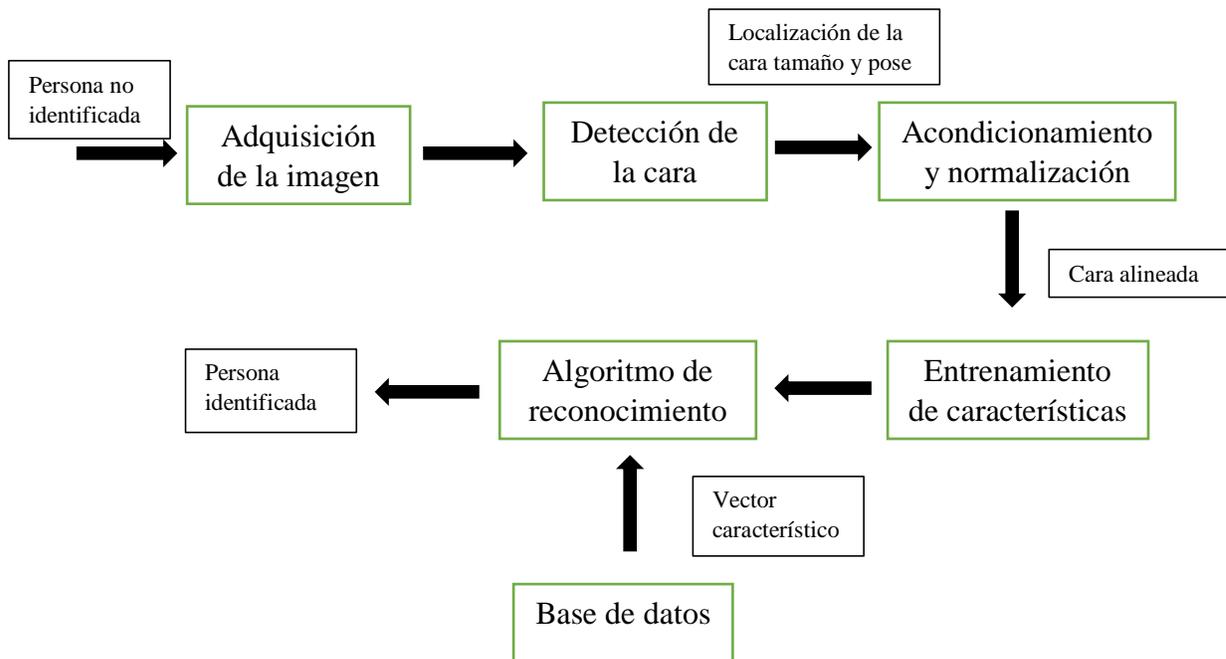


Figura 1.1.1: Fases del reconocimiento facial. [Los autores]

### 1.1.1 Adquisición de la imagen

El proceso de adquisición de la imagen desde el objeto hasta su ingreso en el procesador, se logra a través de un dispositivo que debe ser sensible a los cambios de la luz, el cual es una cámara compuesta por elementos fotosensibles que transforman los fotones de la luz en una corriente eléctrica. En la adquisición de la imagen se tienen dos parámetros:

- **Resolución:** En la actualidad existen dispositivos de captura de imágenes que poseen una resolución de alta calidad, con cámaras digitales de poco más de 50 Mp. Sin embargo, es necesario tener conocimiento de que resolución es la adecuada para trabajar en el proceso que se va a realizar, ya que una cámara de baja calidad puede distorsionar la imagen y suprimir objetos.
- **Formato:** Las imágenes digitales se pueden guardar en diferentes formatos, cada formato tiene una extensión específica, ventajas y desventajas. Una de las desventajas de utilizar formato JPG es que comprime la imagen, eliminando objetos de esta, a diferencia de formatos como TIF y PNG que son formatos que no comprimen la imagen, dejándola completamente original. Las extensiones más utilizados son: BMP, JPG, GIF, TIF, PNG [1].

### 1.1.2 Detección del rostro

A partir de la adquisición de la imagen, se busca la zona de interés, que en este caso es el rostro del individuo.



Figura 1.1.2: Selección del rostro [Modificada por los autores [19]]

### 1.1.3 Acondicionamiento y normalización

Se localizan los componentes del rostro y se normaliza respecto a propiedades geométricas como el tamaño y fotométricas como la exposición a la luz. Esta normalización se puede realizar a partir de la posición de la nariz, la distancia entre pupila o simplemente escalando la imagen [2]. En algunas ocasiones la adquisición de la imagen no es tan precisa y perfecta como se espera, para corregir, mejorar o alterar la imagen se implementan una serie de procesos con el objetivo de que la imagen quede con las condiciones adecuadas de trabajo. Existen muchos procesos que se aplican a las imágenes, sin embargo en este trabajo se presentan los más utilizados:

- **Histograma de brillo:** Esta herramienta muestra las características del brillo de la imagen y es una representación gráfica de la distribución de los niveles de gris de los píxeles en una imagen digital. El histograma proporciona una representación sencilla de la concentración de píxeles frente al brillo, en una imagen [3]. El brillo de una imagen puede ser representado por la ecuación 1.1.

$$B = \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} I(i, j) \quad (1.1)$$

- **Contraste y rango dinámico:** Es la diferencia en la intensidad entre un punto de una imagen y sus alrededores. El rango dinámico es el número de niveles de gris que realmente aparece en una imagen [3]. El contraste de una imagen puede ser representado por la ecuación 1.2.

$$C = \sqrt{\frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} [I(i, j) - B]^2} \quad (1.2)$$

- **Histograma de color:** Es una versión triple del histograma de brillo de escala de grises. En este caso se calculan tres histogramas, uno para cada componente de color. Ayudan a determinar las distribuciones de brillo, contraste y rangos dinámicos de cada componente [3].
- **Conversión de una imagen a escala de grises:** Una imagen en escala de grises es un arreglo matricial de dos dimensiones que aporta información de la intensidad de la luz presente para cada punto de la imagen [4]. Técnicas de conversión:

- **Técnica de promedio (Average):** La forma más simple de lograrlo es mediante la suma de las componentes RGB de cada capa pixel a pixel y dividirles por la cantidad (3) [5]. Representación de la técnica mediante la ecuación 1.3.

$$Grey(i, j) = \frac{R(i, j) + G(i, j) + B(i, j)}{3} \quad (1.3)$$

- **Técnica de Luminosidad (Luminosity):** Este método es una versión más sofisticada del método del promedio. Los valores de cada color presentan un valor teniendo en cuenta la percepción humana. El ojo humano es más sensible al verde que a los otros colores, por lo que el porcentaje de este es mayor. Estos valores fueron establecidos a través de la recomendación *Rec. 601 NTSC* por la *International Telecommunication Union – Radiocommunications*; sector que le ha hecho un estándar mundial para la televisión a color con compatibilidad a blanco y negro [5]. Representación de la técnica mediante la ecuación 1.4.

$$Grey(i, j) = 0.299 * R(i, j) + 0.587 * G(i, j) + 0.114 * B(i, j) \quad (1.4)$$

- **Binarizado de imagen:** La binarización es una variante de la umbralización y consiste en dejar a cero todos los pixels menores de un umbral y a uno aquellos que son iguales o mayores, quedando constituida la imagen final por un conjunto de unos y ceros. Se puede interpretar mediante la ecuación 1.5.

$$b(r, c) \begin{cases} t_0 & \text{si } I(r, c) < T \\ t_1 & \text{si } I(r, c) \geq T \end{cases}$$

Cuando el valor  $t_0=0$  y el valor  $t_1=1$  se dice que la imagen se ha binarizado. El algoritmo es muy sencillo de implementar:

$$IMB(i, j) = (IMA(i, j) \geq umbral)$$

## 1.1.4 Extracción de características

Al tener las zonas u objetos de interés de la imagen, se procede a extraer las características mediante técnicas de extracción de información específicas para el reconocimiento facial, para distinguir entre los rostros de diferentes personas. Entre las técnicas más usadas se encuentran: *Principal Component Analysis* (PCA), *Locality Preserving Projections* o Laplacianface (LPP), *Linear Discriminant Analysis* (LDA), *Elastic bunch graph matching* (EBGM), entre otras.

## 1.1.5 Algoritmo de reconocimiento

Consiste en comparar las características extraídas con las incluidas en una base de datos. Aquí se pueden distinguir dos sistemas:

- **Sistema de verificación o autenticación de identidad:** en este caso el vector de características se compara (se miden distancias) con uno en concreto para determinar si es quién dice ser o quién se quiere probar. Se habrá establecido un umbral por debajo del cual el resultado es positivo, o negativo en caso contrario [2].
- **Sistema de reconocimiento o identificación:** en este caso se busca identificar a la persona, por lo que se realiza una comparación contra cada muestra recogida en la base de datos, obteniendo las diferentes distancias y siendo la menor, la seleccionada para ver si se encuentra bajo el umbral de aceptación. Si fuera así, la persona estaría identificada [2].

Existen varias técnicas de clasificación entre las más comunes se encuentran: Clasificador Vecino más Cercano o *K Nearest Neighbors* (KNN), Clasificador Maquinas de Soporte Vectoriales o *Support Vector Machine* (SVM), Clasificador Red Neuronal Artificial o *Artificial Neural Network* (ANN), clasificador Naives Bayes, entre otros.

## 1.1.6 Base de datos

Para evaluar los algoritmos de reconocimiento facial se necesita una base de datos de prueba. Se pueden utilizar bases de datos de prueba gratuitas como: *The Facial Recognition Technology Database (FERET)*, *Extended Yale Database*, *Olivetti Research Lab (ORL)*, *The BANCA database* o crear la propia base de datos.



Figura 1.1.3: Ejemplo base de datos BANCA [6].

Por otro parte, las aplicaciones basadas en el reconocimiento facial, presentan algunas limitaciones o inconvenientes como pueden ser:

- **Iluminación( incluyendo interior/ exterior ):** Es necesario emplear una adecuada iluminación en la toma de las imágenes para el análisis de características deseadas, para que una característica aparezca en una imagen, la luz debe venir de la fuente de iluminación, reflejarse en el objeto y ser recolectada por el lente del dispositivo. En la figura 1.4 (lado derecho) se muestra una adecuada iluminación frente a una poca iluminación (lado izquierdo).



Figura 1.1.4: Diferencia de iluminación [Modificado por los autores de [21]]

- **Oclusión debido a objetos o accesorios:** Existen artefactos en el rostro que podrían ocasionar problemas en el procesamiento y extracción de características deseadas. Tales artefactos son: anteojos, aretes, piercing, entre otros.
- **Vello facial:** Característica física de las personas. Véase figura 1.6



Figura 1.1.5: Rostro con vello facial. [7]

- **Envejecimiento:** Es un proceso biológico que sufren las personas a medida que pasa el tiempo. Véase figura 1.7

## Capítulo 2

### 2.1 Estudio de técnicas existentes para la extracción de características faciales

Existen dos tipos de técnicas extractoras de características, unas que se basan inicialmente en la apariencia y la otra basada en modelos. Como en este diseño se desea trabajar únicamente con las técnicas de extracción basadas en la apariencia, se explica brevemente la fundamentación de dichas técnicas.

#### 2.1.1 Análisis de componentes principales (PCA, Principal Component Analysis)

PCA es un método que convierte una cantidad de variables probablemente recíprocas en una pequeña cantidad de variables no recíprocas llamadas componentes principales, es decir, es un proceso de reducción dimensional que deja descubrir los vectores que mejor simbolizan la repartición de un grupo de imágenes [8].

Este método se basa en la transformada de Karhunen-loève (KLT), que a la vez se basa en las propiedades estadísticas de la imagen y además no cumple propiedad de separabilidad, esta transformada consiste en la representación de una imagen multi-banda con  $m$  número de bandas en una imagen multi-espectral con igual número de bandas, la diferencia, es que las bandas de la imagen transformadas están ordenadas de acuerdo con la información que contienen.

La primera banda contiene la mayor información, la segunda banda contiene el siguiente contenido más grande y así sucesivamente. Las últimas bandas contienen la información más pequeña que al parecer puede ser tomado como ruido. El ahorro en el almacenamiento depende del descarte y supresión de la información que pueda ser tomada como ruido [9].

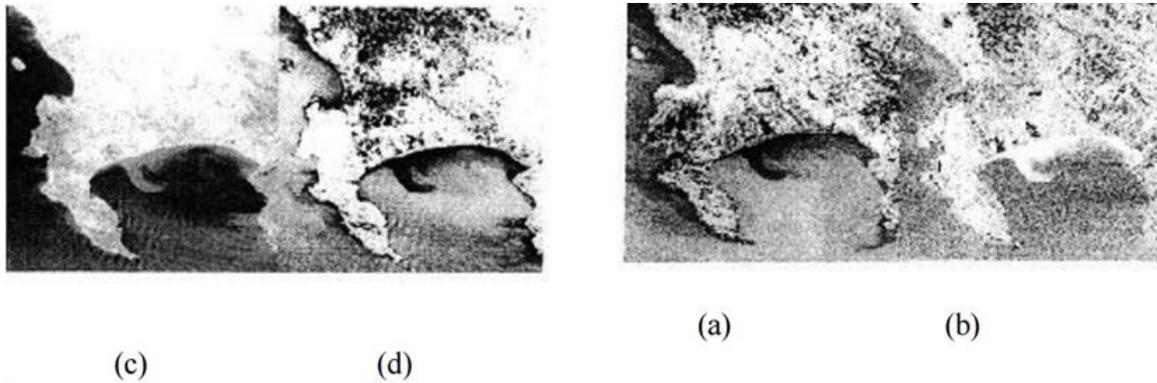


Figura 2.1.1: Resultado del análisis de los componentes principales. (a) primer componente principal, (b) segundo componente principal, (c) tercer componente principal, (d) cuarto componente principal. [9]

El objetivo de este método consiste en representar una imagen en términos de un sistema de coordenadas óptimo reduciendo el número final de componentes que tendrá la imagen.

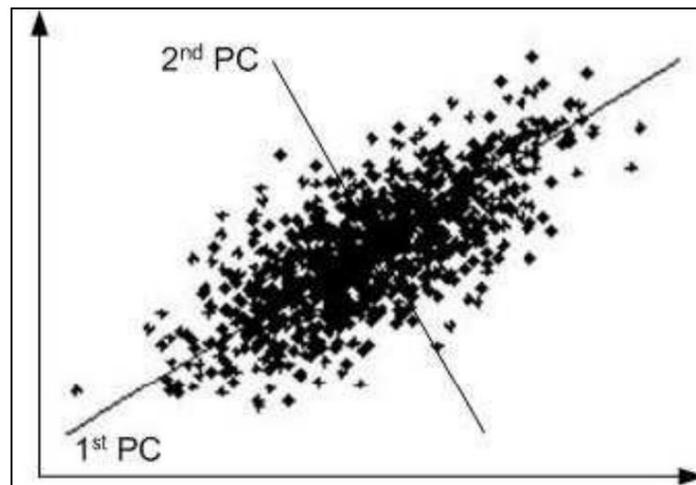


Figura 2.1.2: Componentes principales de un conjunto de puntos bidimensional. La primera componente ofrece una reducción lineal óptima de dimensión de 2D a 1D en cuanto a error cuadrático medio se refiere. [8]

Esta base de transformación depende de las observaciones y se forma a partir de los  $d$  (tamaño de la imagen proyectada) auto-valores más significativos de la matriz de covarianza, los cuales representan las componentes principales que son más comunes en imágenes de diferentes caras.

La matriz covarianza de la matriz  $U$ , la cual contiene a los sujetos descritos ( $j$ ) y las variables de cada uno de ellos estarán descritos por ( $i$ ) conformando una matriz de  $i \times j$ . la vemos representada por:

$$U = \begin{bmatrix} u_{11} & \dots & u_{1j} \\ \vdots & \ddots & \vdots \\ u_{i1} & \dots & u_{ij} \end{bmatrix} \text{ Entonces; } C = \begin{bmatrix} c_{u_1}^2 & \dots & c_{u_1 u_j}^2 \\ \vdots & \ddots & \vdots \\ c_{u_1 u_j}^2 & \dots & c_{u_i u_j}^2 \end{bmatrix}$$

Donde  $c_{u_1}^2$  representa la varianza de la variable de  $U_i$  en el conjunto y  $c_{u_1 u_j}^2$  es la covarianza de  $U_i$  y  $U_j$  [10].

La reducción dimensional realizada por PCA es equivalente al número de auto-vectores que se utilicen. Por lo tanto la imagen proyectada por PCA tendrá una dimensión de valor  $d$ , como se puede ver en la figura 2.1.3 [8].

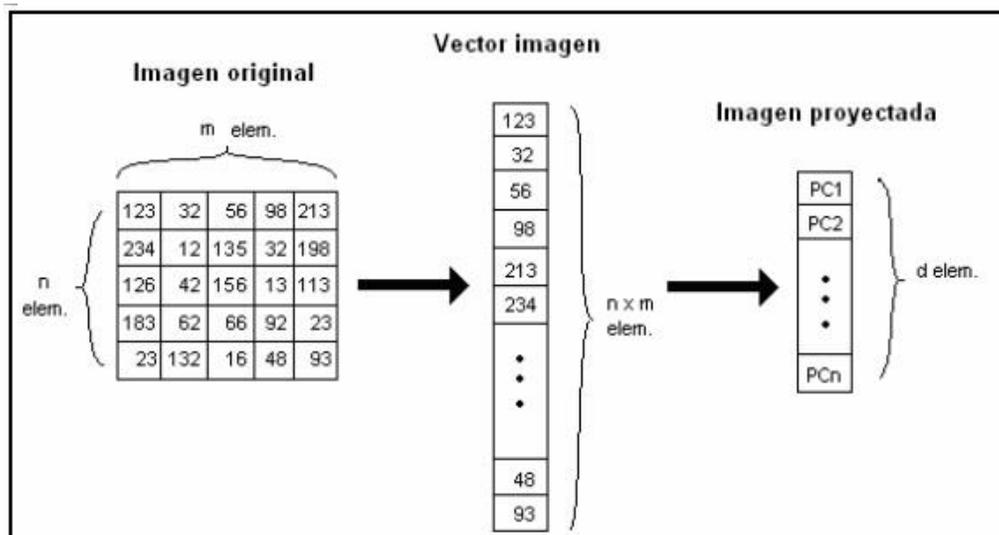


Figura 2.1.3: Ejemplo de reducción dimensional al aplicar PCA. [8]

Estos auto-vectores representan las componentes principales que son más comunes en imágenes de diferentes caras, otra forma de verlo, es que son los ejes del sub-espacio donde proyectaremos las imágenes de caras humanas los cuales se corresponden con las direcciones de máxima varianza de los datos. La matriz de transformación, está formada por los auto-vectores correspondientes a los  $d$  auto-valores más significativos. En la figura 2.1.4 se muestran los primeros 7 auto-vectores que derivan del uso de la base de datos ORL.

La imagen media correspondiente a estos 7 auto-vectores la podemos ver en la figura 2.1.5 y en la figura 2.1.6 podemos ver las imágenes correspondientes a los 7 auto-valores más bajos. Estos últimos se corresponden con auto-valores muy pequeños y son considerados como ruido de manera que no se tienen en cuenta para el reconocimiento [8].

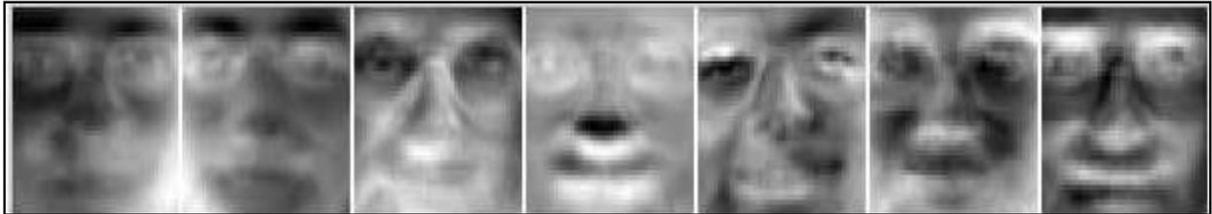


Figura 2.1.4: Autovectores correspondientes a los 7 mayores autovalores mostrados como imágenes  $p \times p = n$  (Obtenidos de la base de datos de caras ORL). [8]

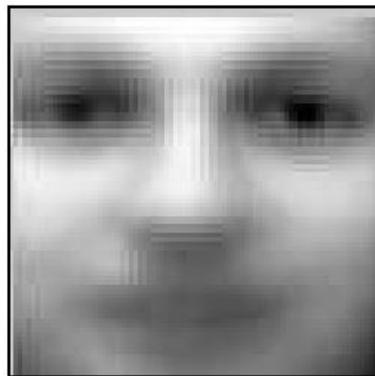


Figura 2.1.5: Cara media de los 7 primeros autovectores (Obtenidos de la base de datos de caras ORL). [8]

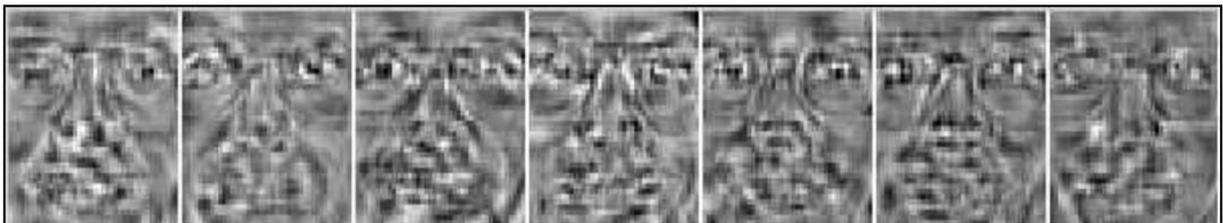


Figura 2.1.6: Autovectores correspondientes a los 7 menores autovalores mostrados como imágenes  $p \times p = n$  (Obtenidos de la base de datos de caras ORL). [8]

## 2.1.2 Análisis lineal discriminante (LDA, Linear Discriminant Analysis)

LDA tiene como objetivo convertir un problema de alta dimensionalidad en uno de baja. Para ello LDA proyecta los datos (imágenes) en un espacio vectorial de baja dimensionalidad de manera que la ratio entre la distancia entre clases y la distancia dentro de la clase se maximiza. De este modo se garantiza una máxima discriminación entre las clases.

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 [8].

En este sentido el análisis discriminante de LDA 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.

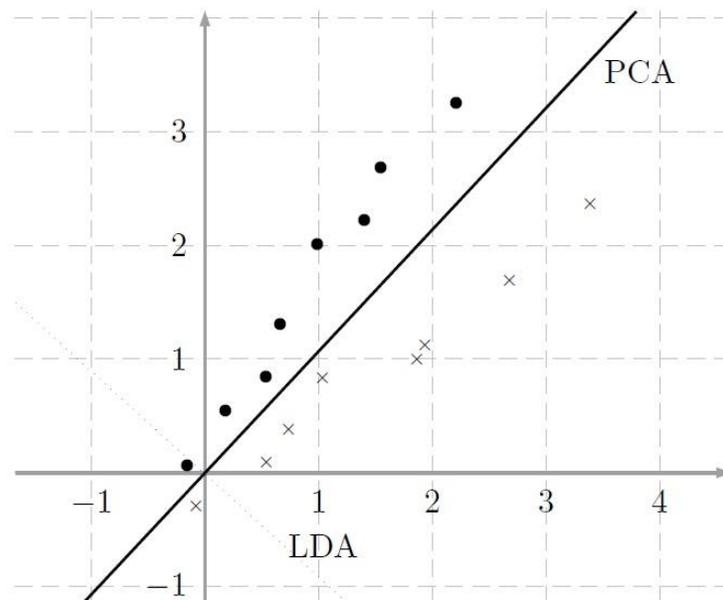


Figura 2.1.7: LDA, ejemplo en dos dimensiones. [11]

La figura 2.1.7 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 [11].

### Descripción del algoritmo

Sea  $X$  un vector aleatorio con muestra de patrones etiquetados en  $c$  clases de cada clase con  $X_i$  patrones:

$$X = \{X_1, X_2, \dots, X_c\}$$

$$X_i = \{X_1, X_2, \dots, X_n\}$$

La matriz de dispersión  $S_B$  y  $S_W$  son calculados como:

$$S_B = \sum_{i=1}^C N_i * (\mu_i - \mu) * (\mu_i - \mu)^T \quad (2.1)$$

$$S_W = \sum_{i=1}^C \sum_{x_j \in X_i} (x_j - \mu_i) * (x_j - \mu_i)^T \quad (2.2)$$

Siendo  $\mu_i$  la media de cada clase,  $N_i$  la cantidad de patrones de la clase  $i$ ,  $\mu$  la media de todos los datos:

$$\mu = \frac{1}{N} * \sum_{i=1}^N x_i \quad (2.3)$$

Y  $\mu_i$  es la media de la casa  $i \in \{1, \dots, \dots, c\}$ :

$$\mu_i = \frac{1}{|X_i|} * \sum_{x_j \in X_i} x_j \quad (2.4)$$

Todo este proceso se realiza con el fin de maximizar el criterio de separabilidad proyectándola en una matriz  $W$  [12].

$$W_{opt} = \operatorname{argmax}_W * \frac{|W^T * S_B * W|}{|W^T * S_W * W|} \quad (2.5)$$

Cuando se obtiene la matriz de proyección, se reduce un problema de alta dimensionalidad a uno de baja. Cada una de las columnas que forman esta matriz serán las bases del nuevo sub-espacio, las cuales se conocen como FisherFaces (figura 2.1.8). Estas bases son aquellas que garantizan una máxima discriminación entre clases por encima de ser las que mejor describen los datos, a diferencia de PCA.

Una de las limitaciones que tiene LDA, es que se requiere de matrices de dispersión no singulares. Esto en aplicaciones de reconocimiento facial es complicado debido a que en muchas ocasiones la alta dimensionalidad de las imágenes es mucho mayor que el número de imágenes lo que desemboca en un problema de matrices singulares, este problema se soluciona aplicando pseudo-inversa en vez de una inversa simple [8].

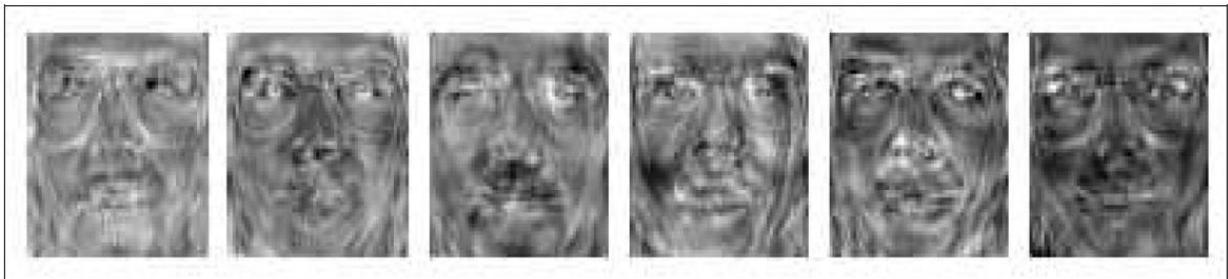


Figura 2.1.8: Las 6 Primeras FisherFaces calculadas en la base de datos ORL. [8]

### 2.1.3 Transformada discreta de coseno (DCT, Discrete Cosine Transform)

La Transformada Discreta del Coseno es un caso especial de la Transformada Discreta de Fourier (DFT) en la que se eliminan los coeficientes senoidales. Ésta toma un conjunto de puntos de un dominio espacial y los transforma en una representación equivalente en el dominio de frecuencias. La DCT está bastante relacionada con la DFT, con la diferencia de que es una transformada real, debido a que los vectores base se componen exclusivamente de funciones coseno muestreadas. Además, la DCT minimiza algunos de los problemas que surgen con la aplicación de la DFT a series de datos. Una DCT produce tantos coeficientes útiles como muestras de entrada [13].

Se definen 8 variantes de DCT, de las cuales las más relevantes son: DCT-I, DCT-II, DCT-III y DCT-IV. En lo que se refiere al reconocimiento facial, la transformación más utilizada es DCT-II, que se define como:

$$v(k_1, k_2) = \alpha(k_1) * \alpha(k_2) * \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} u(n, m) * \cos\left(\frac{(2*n+1)*\pi*k_1}{2*N}\right) * \cos\left(\frac{(2*m+1)*\pi*k_2}{2*M}\right) \quad (2.6)$$

$$0 \leq k_1 \leq N + 1$$

$$0 \leq k_2 \leq M + 1$$

Donde  $U(n)$  es una secuencia de  $N$  elementos,  $v(k)$  es su transformada y:

$$\alpha(0) = \sqrt{\frac{1}{N*M}}, \quad (2.7)$$

$$\alpha(k_1) = \sqrt{\frac{2}{N}}, \quad (2.8)$$

$$\alpha(k_2) = \sqrt{\frac{2}{M}} \quad (2.9)$$

$$1 \leq k_1 \leq N - 1$$

$$1 \leq k_2 \leq M - 1$$

Si la interpretación de esta ecuación, siendo la entrada al sistema una imagen  $U$  y la DCT es la matriz de transformación para obtener  $W$ , se tiene una expresión donde  $W = c(k, x) = c(k_1, n) = c(k_2, m)$  y  $k$  y  $x$  son las filas y columnas:

$$c(k, x) \begin{cases} \frac{1}{\sqrt{N \cdot M}} \\ k = 0, 0 \leq n \leq N - 1, 0 \leq m \leq M - 1 \\ \sqrt{\frac{2}{N}} * \sqrt{\frac{2}{M}} * \cos\left(\frac{(2-n+1) * \pi * k_1}{2 * N}\right) * \cos\left(\frac{(2-m+1) * \pi * k_2}{2 * M}\right) \\ 1 \leq k_1 \leq N - 1, 0 \leq n \leq N - 1, 1 \leq k_2 \leq M - 1, 0 \leq m \leq M - 1 \end{cases} \quad (2.10)$$

De esta manera se puede tratar este algoritmo del mismo modo que PCA o Locality Preserving Projections (LPP), es decir, como un sistema que tiene una matriz de proyección (véase “Estudio de técnicas de reconocimiento facial”. Roger Gimeno Hernández, Página 28 caso única para cada imagen) gracias a la cual se puede proyectar las imágenes a un subespacio de baja dimensionalidad de igual modo que se hace en PCA [8].

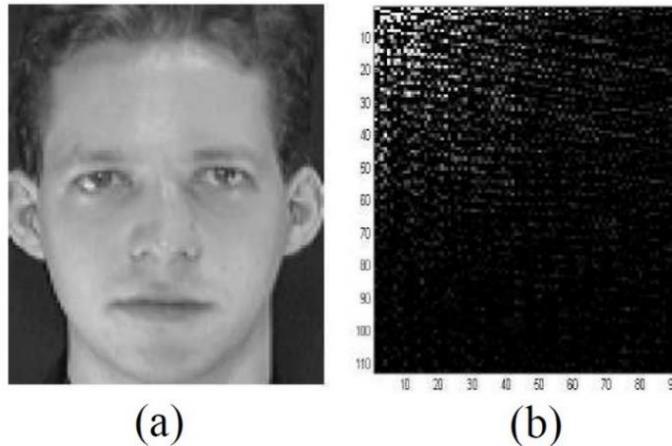


Figura 2.1.9: a) Imagen original (112x92); b) Coeficientes DCT (112x92). [13]

Con relación a PCA, esta técnica tiene un menor coste computacional por lo tanto suele ser muy utilizada en la extracción de características de las imágenes, ya que a diferencia de otras técnicas, no requiere un entrenamiento previo del sistema, además si se incluye una nueva imagen no requiere la ejecución y recálculo de las demás matrices ya que trabaja una matriz de proyección por cada imagen [14].

### 2.1.4 Preservación de proyecciones locales (LPP, Locality Preserving Projections)

Esta técnica, proyecta los datos en la dirección de la máxima varianza, motivo por el cual se puede ver como una variación de PCA. Sin embargo LPP, es menos sensible a los datos atípicos como se puede ver en las siguientes figuras, la diferencia entre estas dos técnicas.

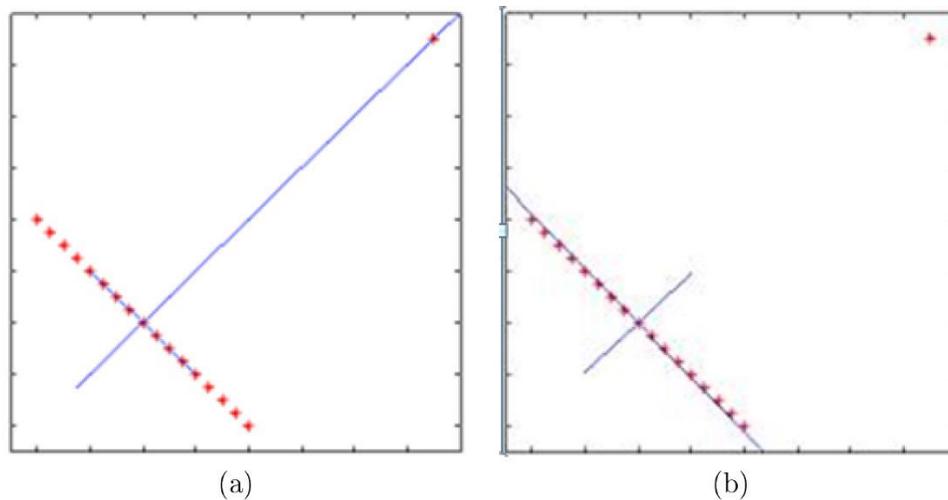


Figura 2.1.10: a) PCA (b) LPP. [14]

LPP posee un algoritmo lineal similar al que se emplea en PCA, es rápido y útil para aplicaciones prácticas ya que reduce la dimensionalidad de los datos, pero a diferencia de PCA que conserva la estructura global de estos datos, en LPP se conserva la estructura local; con esto se logra que los “vecinos” para un dato específico sean los mismos en el espacio original que tiene alta dimensionalidad, y en el nuevo sub-espacio de baja dimensionalidad; así las imágenes pertenecientes a un mismo individuo estarán cercanas entre si y alejadas de las de otros individuos, es decir, hay una discriminación entre clases tal como se muestra en la figura 2.1.11.

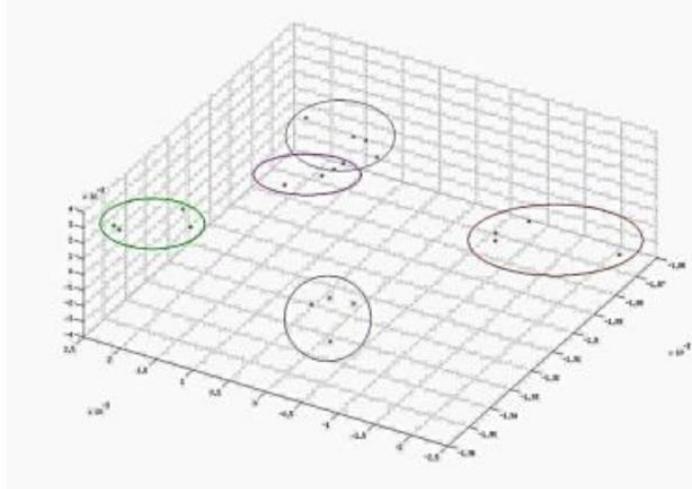


Figura 2.1.11: Representación de la estructura de los datos en el nuevo sub-espacio, ejemplo formado por 5 personas y cuatro imágenes por persona

Pero esta técnica presenta dificultad cuando se requiere recuperar los datos originales a partir de los datos proyectados al nuevo sub-espacio, esto se debe al hecho de hacer uso de bases no ortogonales. También surge otro problema al trabajar con matrices singulares ya que los datos de entrada tienen una mayor dimensionalidad que el número de muestras ( $n \gg N$ ), donde  $n$  son los datos y  $N$  las muestras. Este inconveniente se soluciona mediante técnicas que permitan reducir la dimensión de antes de utilizarlos de manera que  $n = N$  o  $n > N$  [15].

Para conservar la estructura local de los datos se hace uso de un grafo de adyacencias que incluye información de la estructura de los datos. Este grafo, consiste en la creación de una matriz de tamaño  $N \times N$ , donde  $N$  es el número de imágenes, que tiene asignados unos pesos dependiendo de si los elementos  $i$  y  $j$  son vecinos o no. En el momento de crear este grafo cabe la opción de crearlo de manera automática (caso no supervisado), utilizando métodos de búsqueda de ‘vecinos’ como *K nearest neighbors* o  *$\epsilon$ -neighborhoods*, o hacerlo de manera manual (caso supervisado) asignando manualmente quienes son o no vecinos. Todas estas configuraciones serán vistas con más detalle en la gráfica 2.1.12.

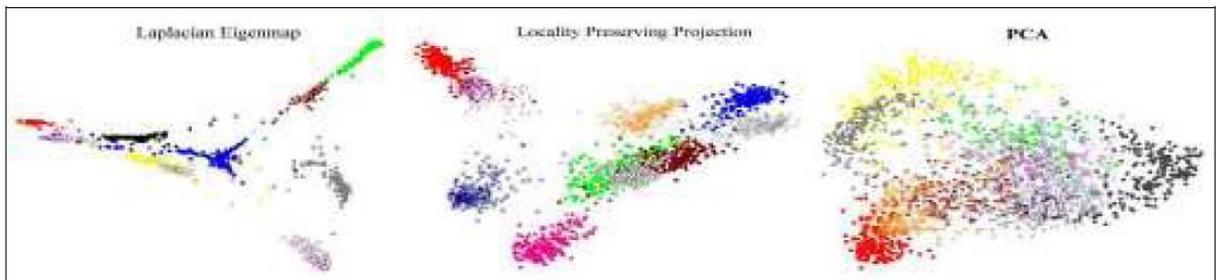


Figura 2.1.12: Se puede notar como a diferencia de PCA, en LPP se conserva la estructura local de los datos. [8]

## 2.1.5 EBGM (Elastic Bunch Graph Matching)

EBGM, es un método que combina imágenes de dos dimensiones, con imágenes de tres dimensiones; por lo que el desarrollo de este puede ser algo complejo. Cuando se habla de EBGM, hablamos de la creación de un grafo en el rostro, el cual se forma a partir de puntos característicos en él; éstos puntos pueden encontrarse en objetos de zonas que conforman el rostro como lo son: los ojos, las cejas, la nariz, la boca y pueden hallarse de forma manual o automática.

Para encontrar los puntos característicos del rostro se tienen dos variantes, una de ellas es separando las zonas mencionadas, realizando un procesado y acondicionamiento a cada objeto, creando un bordeado, de tal forma que se puedan ubicar puntos en los alrededores de éstos. Es de tener en cuenta un adecuado procesamiento, ya que es muy sensible a las perturbaciones por variables externas como la resolución de la imagen y la iluminación. La otra variante de desarrollo es por medio de frecuencias, es decir trabajar las imágenes en el mundo de las frecuencias (Filtros de Gabor) y no en un procesamiento simple de las mismas.

El proceso adecuado para un reconocimiento por EBGM son los siguientes.

### 2.1.5.1 Entrenamiento de la base de datos

1. Se realiza la ubicación de los puntos característicos para crear el grafo de adyacencia, esto se puede hacer por dos métodos.
  - a. Procesamiento de bordes. Este método emana más trabajo, ya que se realiza un procesamiento en la imagen bastante arduo. Además debemos tener una base de datos con condiciones bastante controladas. Se halla el contorno de cada una de los objetos predominantes en el rostro (ojos, cejas, nariz, boca, entre otros); después de tener el contorno exacto del objeto, lo que se hace es recorrer la imagen de borde para ubicar puntos dentro de esta. Estos son; punto central y a partir de este los puntos extremos, es decir donde la línea horizontal y vertical interceptan el borde el objeto.
  - b. Filtro de Gabor. Aunque es un método más sencillo para el posicionamiento de los puntos, se debe tener un conocimiento previo del procesamiento de las imágenes orientado a las frecuencias. Este método lo que hace es convertir la imagen en frecuencias, creando una arquitectura de enlace dinámico que proyecta el rostro sobre una plantilla elástica. Se hallan los Jet's de Gabor los cuales se describen como los nodos de la plantilla elástica o puntos característicos.

2. Después de tener nuestro grafo de adyacencia, se procede a crear una matriz, la cual albergara las distancias entre cada uno de los puntos característicos. Estas distancias varían, por lo que son diferentes para cada individuo, es decir que esta matriz es la que describirá a cada uno de ellos.

### 2.1.5.2 Reconocimiento

1. Se procesa la nueva imagen de entrada como se hizo en la etapa de entrenamiento paso 1 y 2 del capítulo 2.1.5.1.
2. Se realiza la comparación de la matriz descriptora de la nueva imagen con cada una de las matrices descriptoras que se encuentran en la base de datos anteriormente calculadas. El método que se suele usar es el criterio de la distancia mínima para tomar una decisión acertada.

## 2.2 Clasificadores de características

### 2.2.1 Clasificador por vecinos más cercanos (KNN, k-nearest neighbours)

Se define clasificadores KNN, a todos aquellos que se basan en los vecinos más cercanos [16].

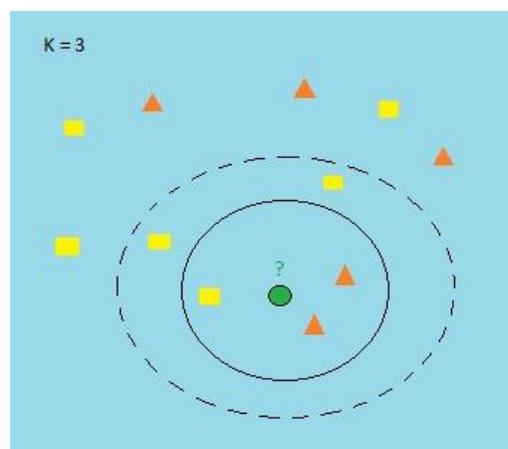


Figura 2.2.1: Ejemplo clasificador KNN [Los autores]

Como se puede visualizar en la imagen anterior, el principio en el que se basa el clasificador KNN es identificar los vecinos más cercanos, a partir de un dato previamente tomado con una base de datos predeterminada. Se puede identificar que la muestra tomada, presente tres vecinos cercanos, los cuales dos de ellos son iguales y el otro diferente, por lo que el algoritmo realiza los cálculos de una distancia euclidiana para cerciorarse, de cuál de ellos se encuentra más cerca, para clasificar la muestra en el tipo de clase que se encuentra.

$\{z_d, 1 \leq c \leq C, 1 \leq l \leq n_c\}$ , donde  $C$  es el número de clases y  $n_c$  el número de prototipos por clase.

$$\{d_1(z, z_{c_1 l_1}), \dots, d_k(z, z_{c_k l_k})\} = \min_{1 \leq c \leq C, 1 \leq l \leq n_c} d(z, z_d) = \min_{1 \leq c \leq C, 1 \leq l \leq n_c} \|z - z_d\|$$

$$\hat{c} = \text{moda}(C), C = \{c_1, \dots, c_k\}$$

Donde  $\| \cdot \|$  representa la distancia Euclidiana. Usando el clasificador KNN el número de cálculos de distancia es [17]:

$$N_{nm} = \sum_{c=1}^C n_c \quad (2.11)$$

KNN es un algoritmo indolente, ya que en el entrenamiento, guarda poca información por lo que, no construye modelos de diferencia, por ejemplo, arboles de decisión. Tampoco es paramétrico, ya que no hace predicciones sobre la distribución de los datos, a diferencia de por ejemplo, un modelo lineal.

KNN es local, ya que acepta que la clase de los datos es dependiente del número de vecinos más cercanos. Las limitaciones e inconvenientes del KNN es que son demasiado sensibles a datos que tienen poca relevancia y la condena de dimensionalidad, esto es consecuencia de ser un modelo local y no global [16].

### 2.2.2 Clasificador de líneas de características más cercanas (NFL, nearest feature lines)

El método NFL es una medida de distancia, pero desde una perspectiva un poco diferente ya que no se calcula la distancia entre muestras propiamente, sino entre líneas formadas por las muestras de diseño. El método se puede comparar en desempeño con otras medidas de distancia como la Euclidiana, Mahalanobis, Manhattan, etc. [18].

La línea  $L_{lk}^c$  que pasa a través de  $z_d$  y  $z_{ck}$  de la misma clase es una línea de características expresada por el generado,  $L_{lk}^c = \text{gen}(z_d - z_{ck})$ . El punto de proyección  $P_{lk}^c$  de un punto objetivo  $z$  sobre la línea  $L_{lk}^c$  es:

$$P_{lk}^c = z_d + \tau(z_{ck} - z_d), \text{ donde } \tau = (z - z_d) * (z_{ck} - z_d) / \|z_{ck} - z_d\|^2 \quad (2.12)$$

La distancia entre el punto objetivo y la línea de características está determinada por  $d(z, L_{lk}^c) = \|z - P_{lk}^c\|$ . El clasificador NFL busca la etiqueta de la clase  $\hat{c}$  de acuerdo a las  $k$  líneas de características más cercanas:

$$\left\{ d_1(z, L_{l_1 k_1}^{\hat{c}_1}), \dots, d_k(z, L_{l_k k_k}^{\hat{c}_k}) \right\} = \min_{\substack{1 \leq c \leq C, \\ 1 \leq l, \\ l \neq k}} d(z, L_{lk}^c) \quad (2.13)$$

$$\hat{c} = \text{moda}(C), \quad C = \{c_1, \dots, c_k\}$$

El número de cálculos de distancia que usa el clasificador NFL es [17]:

$$N_{nfl} = \sum_{c=1}^C n_c(n_c - 1)/2 \quad (2.14)$$

## Capítulo 3

### 3.1 Pruebas en el software Matlab®

Las pruebas en el software se realizan inicialmente para dos métodos de reconocimiento, los cuales son PCA y DTC, los dos basados en la apariencia. El primero que lo hace de forma indirecta a través de una matriz de proyección transformando las imágenes de entrada a un nuevo subespacio y el segundo transformando la imagen original a una matriz de coeficientes mediante la transformada discreta del coseno. Por otra parte, se estudia el proceso básico para el reconocimiento facial mediante LPP.

#### 3.1.1 PCA

PCA es un proceso que permite realizar una reducción dimensional, de manera que nuestro análisis se reduce a unos pocos componentes de estudio. Para ello se necesita tener la matriz de proyección PCA. A continuación se detallan los pasos a seguir para preparar esta matriz y el procedimiento a seguir para el reconocimiento.

##### 3.1.1.1 Creación de la matriz de proyección

Para la creación de esta matriz, se entrena PCA con una cantidad de imágenes ( $\mathcal{X}_i$ ), creando una base de datos de  $N$  personas, donde podremos apreciar a cada persona con una vista diferente, es decir que podremos encontrar fotos frontales, de perfil, con cambios de iluminación, con gafas, con diferentes expresiones faciales. Esto se hace con el fin de que las variables externas mencionadas anteriormente no afecten el reconocimiento.

El proceso que se realiza para crear la matriz de proyección ( $W$ ), consta de los siguientes pasos:

1. Se realiza un reconocimiento previo con una función del software, que hace la identificación un rostro en la imagen, esto se hace con fin de despreocupar objetos fuera de contexto. Después de tener el rostro se crea una nueva imagen con un tamaño recomendado de ( $n=64 \times m=64$ ), donde solo se vea el rostro que se desea procesar

por el algoritmo de PCA. Para este procesamiento se lleva la imagen a una sola capa, es decir a escala de grises.



Figura 3.1.1: Rostro pre-procesado [Modificado por los autores [19]]

2. Se crea el vector característico de la base de datos, esto se realiza generando una tabla con las distintas imágenes, en pocas palabras tendremos una tabla de uno por el número de imágenes ingresadas para la base de datos.

$$X_1 = \begin{bmatrix} x_{1,1} & \dots & x_{1,m} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \dots & x_{n,m} \end{bmatrix} \quad (3.1)$$

Representación vector característico de la base de datos (ecuación 3.2).

$$data = \{X_1 \dots X_i\} \quad (3.2)$$

3. Después de tener la tabla de imágenes o la base de datos, se normaliza la imagen llevando cada una de estas a promedio cero. Para calcular el promedio se hace la suma de cada una de las imágenes y se divide entre el número de imágenes de entrenamiento.

$$\mu = \frac{1}{N} * \sum_{i=1}^N x_i \quad (3.3)$$

Para normalizar la imagen, lo que se hace es restar el promedio previamente calculado a cada una de las imágenes que se encuentran en la base de datos.



Figura 3.1.2: Base de datos normalizada [Los autores]

4. Cuando se tiene la base de datos normalizada, se calcula la matriz de covarianza. Para calcular esta matriz debemos llevar todas las imágenes de la base de datos a un vector característico, es decir llevar una matriz cuadrada de  $n \times m$  (véase ecuación 3.1) a un vector columna (véase ecuación 3.4). Con estos vectores característicos se crea una nueva matriz donde las columnas son las personas y las filas son los valores que caracterizan la imagen.

$$X_1 = \begin{bmatrix} x_{1,1} \\ \vdots \\ x_{1,m} \\ \vdots \\ x_{n,1} \\ \vdots \\ x_{n,m} \end{bmatrix} \quad (3.4)$$

La matriz para el cálculo de la covarianza es:

$$A = \begin{bmatrix} x_{1,1} & \dots & x_{1,N} \\ \vdots & \ddots & \vdots \\ x_{n*m,1} & \dots & x_{n*m,N} \end{bmatrix} \quad (3.5)$$

$$C = A^T * A \quad (3.6)$$

$$C = \begin{bmatrix} c_{1,1} & \dots & c_{1,N} \\ \vdots & \ddots & \vdots \\ c_{N,1} & \dots & c_{N,N} \end{bmatrix} \quad (3.7)$$

- Después de tener la matriz de covarianza, se calculan los eigenvectores y eigenvalores de dicha matriz. Se toman los valores de la diagonal de la matriz de los eigenvalores y se organizan en una matriz columna con etiquetas de 1 hasta N, se ordenan descendientemente. Se multiplica la matriz A, previamente calculada con la matriz de los eigenvalores, esto se hace para calcular los eigenfaces. Se organiza nuevamente la matriz A, para visualizar las imágenes de entrenamiento y se usan las etiquetas para ordenar las imágenes.



Figura 3.1.3: Matriz de Eigenfaces [Los autores]

- Se calculan los valores propios para cada una de las imágenes, esto se puede realizar multiplicando cada fila del vector columna que caracteriza la imagen original (véase ecuación 3.8) con cada fila del vector columna que caracteriza el eigenfaces y realizar la suma del total de los productos (véase ecuación 3.10).

$$X_1 = \begin{bmatrix} x_{1,1} \\ \vdots \\ x_{1,m} \\ \vdots \\ x_{n,1} \\ \vdots \\ x_{n,m} \end{bmatrix} \quad (3.8)$$

$$Eig_1 = \begin{bmatrix} e_{1,1} \\ \vdots \\ e_{1,m} \\ \vdots \\ e_{n,1} \\ \vdots \\ e_{n,m} \end{bmatrix} \quad (3.9)$$

$$Vp = \sum_{i=1}^{n*m} x_i * e_i \quad (3.10)$$

### 3.1.1.2 Reconocimiento

Al tener una nueva imagen para realizar un test por medio del reconocimiento PCA, el siguiente procedimiento se realiza para cada una de ellas.

1. Para proceder al reconocimiento inicialmente se procesa la imagen como se hizo anteriormente en el paso 1 del capítulo 3.2.1, esto se realiza con el fin de dar una estandarización a las imágenes que se intervienen en el proceso de reconocimiento.

2. Al igual que todas las imágenes de entrenamiento, la imagen de entrada en el reconocimiento debe ser llevada a una media cero. Hacemos uso del promedio anteriormente calculado en el capítulo 3.2.2.

$$Xs - \mu \quad (3.11)$$

Donde  $Xs$  es la nueva imagen

3. Proyectamos nueva imagen al nuevo sub-espacio para realizar la comparación entre la nueva imagen de entrada y la base de datos previamente almacenada.

$$VpXs = \sum_i^N (Xs - \mu) * Eig_1(i) \quad (3.12)$$

4. Para finalizar se realiza un método de decisión. En el este caso se aplicara el criterio de la distancia mínima.

### 3.1.2 LPP

Este método realiza una reducción dimensional y requiere, como PCA, de un entrenamiento previo del sistema con otras imágenes de caras para crear una matriz de proyección. Al realizar los pasos para la creación de la matriz de proyección de LPP, se deben usar algunos parámetros que anteriormente se calcularon en capítulo 3.2.1. Se hace necesario el uso de la matriz de proyección de PCA, para el cálculo de la matriz previa de LPP.

#### 3.1.2.1 Creación de la matriz de proyección.

1. Inicialmente se realizan todos los pasos del capítulo 3.2.1, donde se hace el entrenamiento para el reconocimiento de PCA.
2. Se pre-procesan las imágenes como se realiza en los pasos 1, 2, 3 del capítulo 3.2.2, en la sección de reconocimiento para PCA.
3. Se proyectan las imágenes mediante de PCA.
4. Una vez se tienen las imágenes proyectadas, se crea un grafo de adyacencia donde se enlazan las imágenes con vecindades más cercanas, es decir que las imágenes de la mismas persona se encuentran enlazadas hasta formar el grafo. Esto se puede realizar a través de un clasificador, siendo así un sistema supervisado.

- Con el grafo de adyacencia, se crea una matriz de adyacencia la cual describe los pesos ( $W$ ) de cada una de las imágenes dependientes a sí mismas, es decir que se evalúa cada una de las imágenes comparadas con las existentes en la base de datos, cuando las imágenes son vecinas en dicha posición se pondrá un 1 para el caso de Simple-minded (método binario); si se encuentra en una posición donde la comparación es consigo misma, se pondrá un cero. Previamente se visualizara gráficamente.

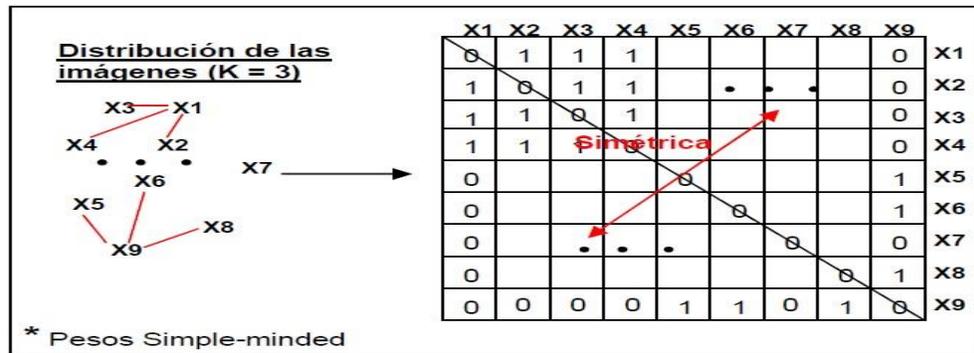


Figura 3.1.4: Ejemplo matriz de adyacencia [8]

- Antes de realizar los cálculos de los eigenvectores y los eigenvalores, se debe resolver las ecuaciones para preparar los pesos asignados para cada imagen, siendo todos dependientes de la matriz de adyacencia.

$$D_{ii} = \sum_j^N W_{ij} \quad (3.13)$$

- Se resuelve la siguiente ecuación, para ello debemos crear una matriz diagonal con el vector fila anteriormente calculada.

$$L = D - W \quad (3.14)$$

- Se procede a calcular los eigenvectores y eigenvalores para este caso, al contrario de PCA se ordenan los eigenvalores ascendentemente.
- Finalmente se realiza el cálculo de los valores propios para la matriz de proyección de LPP, esto se puede resolver igual que el paso 5 del capítulo 3.2.1. Teniendo en cuenta que no se hacen uso de la matriz de proyección de PCA, sino la de LPP previamente calculada.

### 3.1.2.2 Reconocimiento

Cuando procedemos a realizar el reconocimiento de una nueva imagen por medio de LPP, podremos decir que llevan la misma estructura que se realizó anteriormente en PCA.

1. Se realizan los pasos 1, 2 y 3 del capítulo 3.2.2, donde encontraremos la etapa de reconocimiento de PCA.
2. Se proyectan las imágenes al nuevo sub-espacio, como anteriormente se mencionó haremos uso de las dos matrices de proyección de PCA y LPP. Para realizar la proyección de la nueva imagen de entrada debemos cumplir con la siguiente ecuación.

$$VpXr = (W_{PCA} * A_{LPP})^T * x \quad (3.15)$$

3. Después de obtener los valores propios o la imagen proyectada procedemos a la comparación o identificación por medio de un método de selección. En el este caso se aplicara el criterio de la distancia mínima.

### 3.1.3 DCT

DCT (Discrete Cosine Transform) es otro sistema de clasificación basado en la apariencia, a diferencia de PCA y LPP la matriz de transformación es fija e independiente para cada imagen y no necesita un entrenamiento previo del sistema con otras imágenes. Para el desarrollo del algoritmo en este trabajo se tienen los siguientes pasos:

1. Se acondiciona la imagen para el procesamiento utilizando una función propia de Matlab que identifica el rostro en una imagen, recortándola y dejando solo el rostro en una nueva imagen. Se crea la nueva imagen con un tamaño ( $n=64 \times m=64$ ) y en formato de escala de grises.



Figura 3.1.5: Rostro pre-procesado [Modificado por los autores [19]]

2. Teniendo la imagen en escala de grises, se procede a calcular la transformada discreta del coseno (DCT) sobre la imagen original. Para este proyecto se hace uso de la función propia de Matlab (`dct2`). Al transformar la imagen se crea una representación de secuencia finita de datos como la suma de una serie de funciones cosenoidales oscilando a diferentes frecuencias, generando una matriz de coeficientes del mismo tamaño de la imagen original como se muestra a continuación.

$$\begin{array}{ccc}
 128 & 4 & 32 \\
 56 & 9 & 8 \\
 4 & 478 & 93
 \end{array}
 \xrightarrow{\text{DCT}}
 \begin{array}{ccc}
 \alpha(0) & \alpha(2) & \alpha(5) \\
 \alpha(1) & \alpha(4) & \alpha(7) \\
 \alpha(3) & \alpha(6) & \alpha(8)
 \end{array}$$

Figura 3.1.6: Ejemplo de aplicación DCT sobre una imagen [Los autores]

- Después de crear la matriz, se seleccionan (d) coeficientes que se utilizaran para el reconocimiento. La selección de los coeficientes se hace en forma de zigzag y la cantidad que se desee y luego se forma un vector columna (véase figura 3.1.3.3).

$$\begin{array}{ccc}
 \alpha(0) & \alpha(2) & \alpha(5) \\
 \alpha(1) & \alpha(4) & \alpha(7) \\
 \alpha(3) & \alpha(6) & \alpha(8)
 \end{array}
 \xrightarrow{\mathbf{d} = 3}
 \begin{array}{c}
 \alpha(1) \\
 \alpha(2) \\
 \alpha(3)
 \end{array}$$

Figura 3.1.7: Ejemplo selección en zigzag de los (d) coeficientes

### 3.1.3.1 Reconocimiento

Cuando se tiene una nueva imagen de prueba, se acondiciona la imagen de igual forma con los 3 pasos mencionados anteriormente. Después de crear el vector columna con los coeficientes de la imagen de prueba, se procede a comparar la imagen de prueba con la de referencia y se decide. Para este proyecto se utiliza un criterio de distancias, en este caso la distancia Euclidiana y el criterio de decisión es el de distancia mínima.

## 3.2 Bases de datos utilizadas en las pruebas

### 3.2.1 The MUG (Multimedia Understanding Group)

La base de datos MUG es creada por Multimedia Understanding Group. Es creada para superar algunas limitaciones de las otras bases de datos similares, tales como alta resolución, iluminación uniforme, muchos temas y muchas tomas por sujeto. El objetivo es ayudar a la investigación en el campo del reconocimiento facial [19].

#### 3.2.1.1 Descripción

La base de datos consta de secuencias de imágenes de 86 sujetos que realizan expresiones faciales diferentes. Los sujetos fueron sentados en una silla delante de una cámara. El fondo era una pantalla azul. Dos fuentes de luz de 300W cada uno, montados en soportes a una altura de 130 cm aproximadamente. En cada stand de un mismo paraguas se fijó con el fin de difundir la luz y evitar sombras. La cámara es capaz de capturar imágenes a una velocidad de 19 fotogramas por segundo. Cada imagen se guarda con un formato jpg,  $896 \times 896$  píxeles y un tamaño que va desde 240 a la 340 KB.

En la base de datos participaron 35 mujeres y 51 hombres todos de origen caucásico de entre 20 y 35 años de edad. Los sujetos no están usando gafas salvo a 7 sujetos en la segunda parte de la base de datos. No hay oclusiones, excepto para unos pocos.

Las imágenes de 52 sujetos están a disposición de los usuarios de Internet autorizados. 25 sujetos están disponibles bajo petición y el resto 9 sujetos sólo están disponibles en el laboratorio MUG.

Hay dos partes en la base de datos. En la primera parte se pidió a los sujetos que realizaran las seis expresiones básicas, que son la ira, asco, miedo, alegría, tristeza, sorpresa. La segunda parte contiene las emociones inducidas en el laboratorio [19].



Figura 3.2.1: Ejemplo de algunos personajes de la base de datos MUG [Modificado por los autores [19]]

## 3.2.2 Base de datos Olivetti (ORL Database)

### 3.2.2.1 Descripción

Se trata de una base de datos formada por 40 individuos, a razón de 10 imágenes por individuo tomada entre 1992 y 1994 [27]. En esta base de datos se pueden encontrar tanto imágenes de hombres como de mujeres.

Como es posible observar en la figura 3.2.2.1, en esta base de datos se pueden encontrar imágenes con diferentes posiciones y expresiones faciales (ojos abiertos/cerrados, sonriendo o no,...), lo cual será una dificultad añadida para el sistema. Del mismo modo se puede observar como en algunos sujetos, se tendrán imágenes con distintos detalles faciales (gafas o no, bigote o no,...).

Por último decir que todas estas fotos han estado tomadas bajo unas condiciones de iluminación y de fondo de imagen poco controladas. El tamaño de todas estas imágenes es de 92x112 píxeles, con 256 niveles en la escala de grises para cada píxel [20].



Figura 3.2.2: Ejemplo base de datos ORL [20]

# Capítulo 4

## 4.1 Resultados obtenidos

Las pruebas que se realizan para este proyecto, se hacen con las bases de datos MUG y ORL mencionadas anteriormente. Se toman cuatro personas para realizar el entrenamiento previo de los algoritmos PCA y DCT, con el algoritmo de LPP no se realizan pruebas, ya que se presentan algunas inconsistencias dentro del entrenamiento o creación de la matriz de proyección, por lo que la identificación es poco efectiva.

Como se tienen bases de datos con diferente número de imágenes por persona. Para el caso de la MUG se toman diez imágenes de pruebas, mientras que en la base de datos ORL, por ser una base de datos tan reducida, se toman dos imágenes para el entrenamiento y ocho para las pruebas.

Se realiza el entrenamiento con las imágenes (asignación aleatoria), después de tener los algoritmos entrenados se procede a realizar el reconocimiento para cada una de las imágenes de prueba. En el algoritmo de DCT se toman para estas pruebas cincuenta coeficientes de la transformada discreta del coseno y para PCA se toman todos los eigenfaces.

Cuando se realiza las pruebas con PCA y la base de datos MUG, los resultados que se obtienen son los siguientes

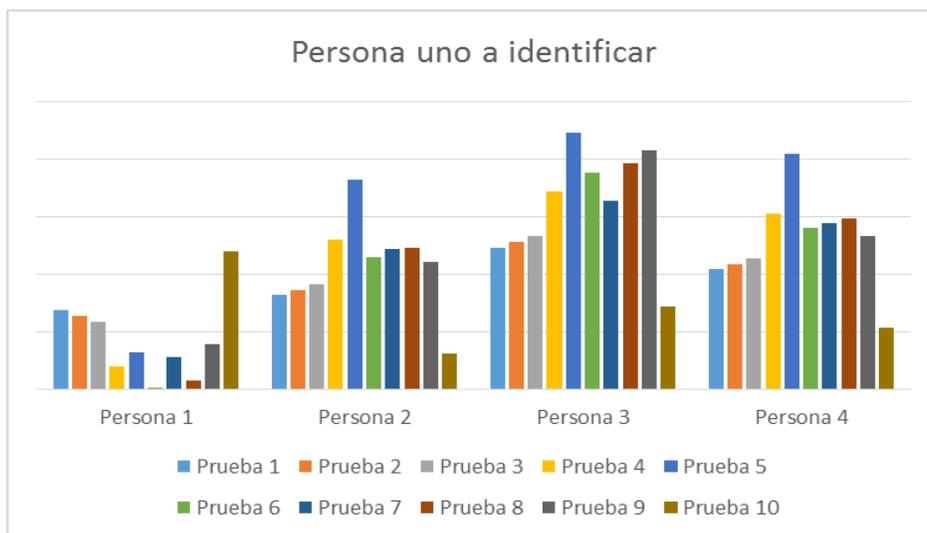


Figura 4.1.1: Pruebas realizadas con las diez imágenes de la persona uno de la base de datos MUG, evaluando el algoritmo PCA [Los autores]

Lo que se visualiza en la Figura 4.1.1 es la gráfica identificando la primera persona, con las diez imágenes de prueba seleccionadas. Debido a que se toma el método de selección de distancia mínima, los valores que se observan son una identificación correcta de la persona, como se puede ver en este caso la persona uno tiene el promedio más bajo comparado con las otras tres personas de prueba; como también podemos observar, la prueba diez confunde la persona uno con la persona dos, donde se evalúa la efectividad del algoritmo.

Las pruebas se realizan para las siguientes tres personas y se obtienen los siguientes resultados.

Se observa en cada una de las imágenes que la identificación resulta bastante bien, aunque se nota que existen algunos casos donde el algoritmo sufre confusión, cuando se realizan las pruebas observamos que las imágenes dos, tres y cuatro; tienen similitudes por lo que la diferencia entre ellas es poco notable a comparación de la uno que es bastante variable.

Se realizan ahora las pruebas con DCT, con la base de datos MUG y se obtienen los siguientes resultados.

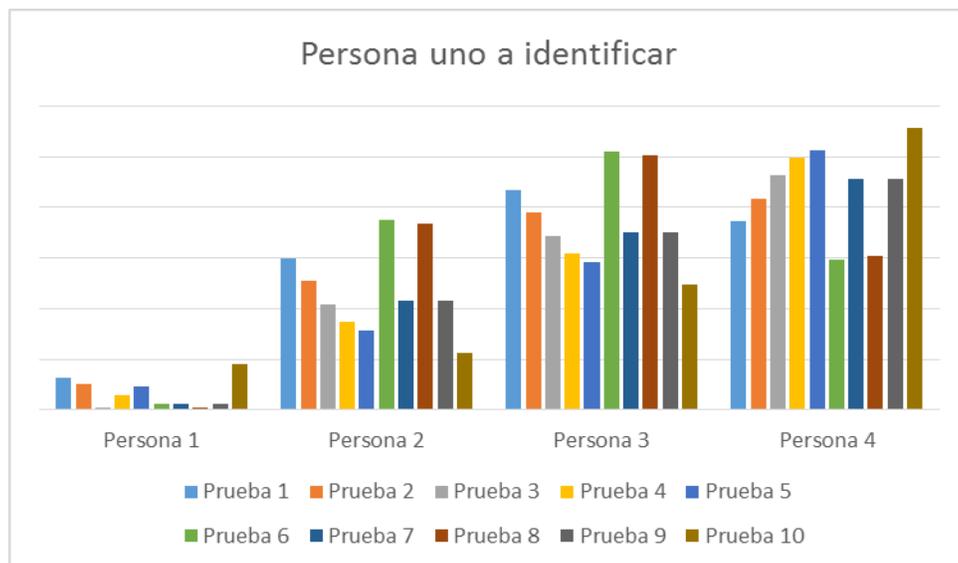


Figura 4.1.2: Pruebas realizadas con las diez imágenes de la persona uno de la base de datos MUG, evaluando el algoritmo DCT [Los autores]

Se observa que la identificación en esta primera prueba es 100% efectiva, con distancias muy lejanas de la persona uno con respecto a las otras tres personas, en este algoritmo también se realiza la selección del reconocimiento por distancia mínima.

Al igual que PCA, las pruebas se realizan para las imágenes de las otras tres personas y se obtienen los siguientes resultados.

Para estas pruebas se pueden observar que las diferencias entre las personas son bastante notables entre cada una de las imágenes de prueba, sin embargo, también existen fallos o confusiones entre la persona tres y cuatro.

Ahora se procede a realizar las mismas pruebas pero con la base de datos ORL. En este caso los algoritmos deben ser modificados antes de realizar la prueba, ya que esta base de datos contiene todas sus imágenes en escala de grises.

Se inicia con el algoritmo de PCA, y se obtienen los siguientes resultados.

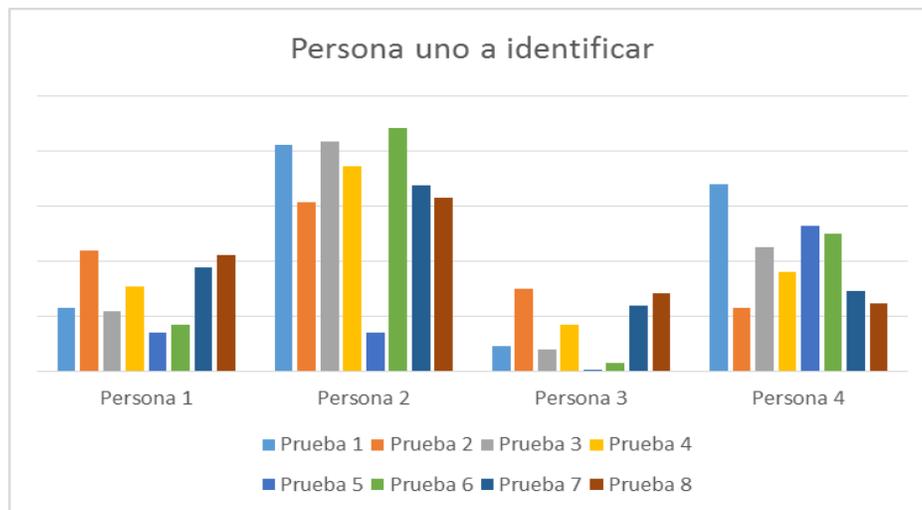


Figura 4.1.3: Pruebas realizadas con las ocho imágenes de la persona uno de la base de datos ORL, evaluando el algoritmo PCA [Los autores]

En estas pruebas realizadas se observa que las tendencias del reconocimiento son a la persona tres, es decir que en cada una de las pruebas realizadas para cada persona se tiende más a reconocer la persona tres y es la que predomina en las cuatro gráficas.

Ahora se procede a iniciar las pruebas con el algoritmo DCT, con la base de datos seleccionada ORL, obteniendo los siguientes resultados

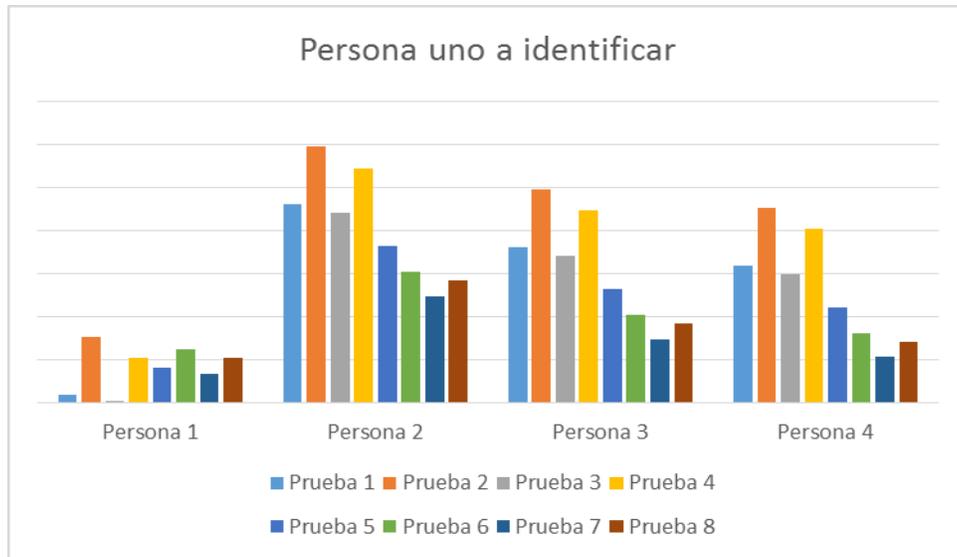


Figura 4.1.4: Pruebas realizadas con las ocho imágenes de la persona uno de la base de datos ORL, evaluando el algoritmo DCT [Los autores]

Se puede notar que para el reconocimiento de la primera persona con esta base de datos, el algoritmo es muy eficaz, las diferencias entre los valores de distancia son bastante notables, por lo que lo hace aún más efectivo.

Se visualiza que existe un deterioro en el algoritmo, ya que está en algunas de las pruebas de la persona cuatro, confunde con personas distintas a las de entrada. También podemos observar que las pruebas de la persona dos solo hubo una inconsistencia, mientras en las pruebas de la persona tres y cuatro las inconsistencias aumentan entre tres y seis.

Con los resultados que se tienen podremos dar un porcentaje de eficiencia para cada uno de los algoritmos, se toman el número de errores o inconsistencia que se obtuvieron en las pruebas y calculamos sus porcentajes.

ALGORITMO	BASES DE DATOS	
	MUG	ORL
PCA	67,5 %	22,5 %
DCT	80 %	75 %

Figura 4.1.5: Tabla de porcentajes de eficiencia, algoritmos PCA y DTC [Los autores]

Como se utilizan dos bases de datos distintas con condiciones diferentes, MUG es una base de datos rigurosamente controlada a todo tipo de perturbaciones externas en la captura, mientras ORL, es una base de datos con un control de perturbaciones más sencilla. En estas estadísticas podemos ver que PCA es más propenso a reducir su rendimiento cuando las imágenes son poco controladas en comparación con DCT.

Finalmente, se miden los tiempos de procesamiento para las pruebas realizadas con cada uno de los algoritmos. Inicialmente se mide el tiempo que transcurre mientras se realiza el entrenamiento. Esto se realiza para para cada uno de los algoritmos. Después se procede a realizar el cálculo del tiempo que tarda en hacer el reconocimiento.

	PCA	DCT
Entrenamiento	10.662194 seg.	10.267460 seg.
Reconocimiento	1.440923 seg.	2.361641 seg.

Figura 4.1.6: Tabla de tiempos de procesamiento para los algoritmos [Los autores]

Como se observa, los tiempos de entrenamiento son similares, pero los tiempos del reconocimiento aumentan para el algoritmo DCT, pero se observa en todos los resultados que a pesar que PCA es más rápido, DCT es más efectivo.

## Capítulo 5

### 5.1 Presupuesto del proyecto

ACTIVIDAD	DESCRIPCION	RUBRO	TIEMPO EJECUCION	VALOR UNITARIO	VALOR TOTAL
1.1	Investigación bases de datos científicas	Gastos de administración	30 días	\$3.500	\$105.000
		Acceso a información especializada	20 días	\$0	\$0
1.2	Visita técnica Universidad Tecnológica de Pereira	Transporte	70 retornos	\$500	\$35.000
1.3	Investigación en bases de datos antiguas	Gastos de administración	40 días	\$3.500	\$140.000
1.4	Obtención de libros y CDS	Acceso a información especializada	15 días	\$0	\$0
		Gastos de administración	10 días	\$3.500	\$35.000
2.1	Análisis de la información	Costos personal técnico no especializado	30 días	\$36.000	\$1.080.000
2.2	Realización de pruebas	Costos personal técnico especializado	60 días	\$64.000	\$3.840.000
		Costos personal técnico no especializado	85 días	\$36.000	\$3.060.000
3.1	Información sobre elementos necesarios	Gastos de administración	35 días	\$3.500	\$122.500
		Transporte	40 retornos	\$500	\$20.000
3.2	Selección de los elementos para el proyecto	Costos personal técnico especializado	10 días	\$64.000	\$640.000
		Costos personal técnico no especializado	25 días	\$36.000	\$900.000
				Total presupuesto	\$9.977.500

# Capítulo 6

## 6.1 Conclusiones y recomendaciones

- En el proyecto se trabajó con la base de datos MUG y ORL, para obtener unos resultados competitivos y poner a prueba los algoritmos implementados, ya que MUG es una base de datos rigurosamente controlada a diferencia de ORL que es poco controlada. Los resultados en términos de porcentaje para la eficiencia de los algoritmos con las dos base de datos fueron: PCA con la base datos MUG 67,5% y ORL 22,5%; y para DCT con MUG 80% y ORL 75%.
- PCA es una técnica que requiere un entrenamiento previo del sistema y es muy importante el número de imágenes utilizadas, aun siendo estas ligeramente diferentes a las que se usan para el reconocimiento o con mucha variedad de individuos, de esto depende mucho su eficiencia.
- Para los resultados se escogieron una cantidad de componentes apropiado para un buen funcionamiento y mejores resultados. Para DCT, se utilizaron 56 coeficientes y para PCA todos sus componentes.
- En cuanto al número de componentes utilizado para el reconocimiento, se concluye que cada uno de los métodos requiere un número de componentes diferente para obtener su mejor rendimiento.
- Cabe destacar que la cantidad de componentes utilizados se pueden modificar en caso de utilizar las técnicas bajo condiciones diferentes y específicas del entorno para un adecuado tratamiento de las imágenes.
- Se debe tener en cuenta a la hora de seleccionar un métodos de reconocimiento facial, buscar un compromiso entre respuesta del sistema, el entorno de uso y tiempo de respuesta.
- Para un buen funcionamiento del sistema, es necesario que las imágenes estén recortadas y alineadas, teniendo así solo la región de interés en la imagen, en este caso el rostro.

- Es recomendable para el algoritmo DCT, en la selección de los coeficientes descartar los primeros coeficientes para tener más control sobre los cambios de luminosidad, ya que estos son los que tienen mayor energía y por esta razón se ven afectados por los cambios de luminosidad. También se pueden descartar los últimos coeficientes, ya que son poco relevantes en la comparación.

## Bibliografía

- [1] C. Ordoñez Santiago, "LA ESCRITURA Y LOS SOPORTES DIGITALES", *Revista Digital Universitaria*, vol. 6, no. 1, p. 7, 2005.
- [2] V. Fernández, 'Biometría facial - Dolthink', *Dolthink*, 2014. [Online]. Available: <http://www.dolthink.com/biometria-facial.html>. [Accessed: 07- Nov- 2015].
- [3] M. Bravo Rojas, L. Oviedo Sandoval and A. Perez de Paul, "Procesamiento digital de imagen", *M.sb-10.com*, 2015. [Online]. Available: <http://m.sb-10.com/medicina/16962/index.html>. [Accessed: 07- Nov- 2015].
- [4] Wilhelm Burger, Mark J. Burge, *Principles of Digital Image Processing*, Springer-Verlag London Limited, 2009.
- [5] J. A. Cortés Osorio, W. Ardila Urueña, and J. A. Mendoza Vargas, "Técnicas alternativas para la conversión de imágenes a color a escala de grises en el tratamiento digital de imágenes" no. 47, pp. 207–212, 2011.
- [6] C. Chan, "The BANCA Database", *Ee.surrey.ac.uk.*, 2004 [Online]. Available: <http://www.ee.surrey.ac.uk/CVSSP/banca>. [Accessed: 05- Mar- 2016].
- [7] ECE Department, Aristotle University of Thessaloniki, *The MUG Facial Expression Database - Example 2*. 2010.
- [8] R. Gimeno, "Estudio de técnicas de reconocimiento facial" pp. 19–28, 2010.
- [9] J. Pérez, "Trasformada de karhunen-loève (KLT)," pp. 1–23, 2007.
- [10] O. E. Gualdrón, O. Manuel, D. Suárez, M. Anastasio, and C. Rojas, "Diseño de un sistema de reconocimiento de rostros mediante la hibridación de técnicas de reconocimiento de patrones , visión artificial e IA , enfocado a la seguridad e interacción robótica social.," vol. 6, pp. 4–5, 2013.
- [11] M. Armengot, "Análisis comparativo de métodos basados en subespacios aplicados al reconocimiento de caras," 2006.
- [12] C. Guillermo and N. Andrade, "Autenticación por reconocimiento facial para aplicaciones web, utilizando software libre.," 2012.
- [13] C. M. Travieso, J. Solé-Casals, V. Zaiats, J. B. Alonso, and M. A. Ferrer, "Reducción del vector de características," no. 1, pp. 22–24, 2008.
- [14] O. L. Echeverry, "Sistema integrado para reconocimiento de rostros," 2014.
- [15] A. Amaya, "Sistema alternativo de seguridad vehicular basado en reconocimiento facial.," 2015.
- [16] R. Aler, "Clasificadores KNN-I," 2015.

- [17] M. Orozco, "Selección efectiva de características wavelet en la identificación de bioseñales 1-d y 2-d usando algoritmos genéticos.," pp. 13–14, 2005.
- [18] P. Rivas, "Reconocimiento facial en ambientes no cooperativos," 2006.
- [19] A. Delopoulos, "Multimedia Understanding Group | ECE Department, Aristotle University of Thessaloniki", *Mug.ee.auth.gr*. [Online]. Available: <http://mug.ee.auth.gr>. [Accessed: 04- Mar- 2016].
- [20] A. Hopper, 'The Database of Faces', *Cl.cam.ac.uk*, 1994. [Online]. Available: <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>. [Accessed: 07- Nov- 2015].
- [21] D. Muñoz Acosta and D. Cardona Cortes, "Diseño e implementación de un sistema biométrico mediante el iris", Ingeniería, Universidad tecnológica de Pereira, facultad de tecnología, Mecatrónica, 2012.