



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

TP 2 - Tu cara me suena...

MÉTODOS NUMÉRICOS

Resumen

En este trabajo práctico, desarrollamos un método para el reconocimiento de caras basado en el enfoque de *eigenfaces*. Para ello, manipulamos la base de imágenes de entrenamiento matricialmente; obteniendo las componentes principales (autovectores) de su matriz de covarianzas a través del método de la potencia con deflación. Exploramos algunos métodos de clasificación, su tasa de efectividad y su relación con posibles variaciones en el conjunto inicial de entrenamiento.

Keywords: *Face Recognition, Componentes Principales, SVD*

Grupo 2

Integrante	LU	Correo electrónico
Franco Bartalotta		franco.bartalotta@hotmail.com
Fernando Gasperi Jabalera	56/09	fgasperijabalera@gmail.com
Ana Sarriés	144/02	abarloventos@gmail.com

Reservado para la cátedra

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.exactas.uba.ar>

Índice

1. Introducción Teórica	3
2. Desarrollo	4
2.1. La implementación y algunas decisiones de diseño	4
2.2. Elección conveniente de la matriz de covarianzas para imágenes grandes	5
2.3. Comparación de la complejidad de los métodos	6
2.4. Algoritmos	7
2.5. Tests	8
2.5.1. Cantidad de componentes principales vs tasa de efectividad	8
2.5.2. Análisis de la tasa de eficiencia en función de la cantidad de personas	8
2.5.3. <i>nimgp</i> vs Tasa de efectividad	9
2.5.4. Resolución de las imágenes vs Tasa de efectividad	9
2.5.5. Comparación de los métodos de identificación	10
3. Resultados	10
3.1. Test de tasa de eficiencia en función de la cantidad de componentes principales consideradas	10
3.2. Test de tasa de eficiencia en función de la cantidad de personas	11
3.3. Test de tasa de eficiencia en función de <i>nimgp</i>	12
3.4. Test de comparación de los métodos de identificación	13
3.5. Test de tasa de eficiencia en función de la resolución de las imágenes	14
4. Discusión	14
4.1. Análisis de resultados de tasa de efectividad en función de las componentes principales	14
4.2. Análisis de la tasa de eficiencia en función de la cantidad de personas	15
4.3. Análisis de la tasa de efectividad en función de <i>nimgp</i>	15
4.4. Análisis de comparación de los métodos de identificación	15
4.5. Análisis de la tasa de eficiencia en función de la resolución de las imágenes	15
5. Conclusiones	16

1. Introducción Teórica

El objetivo principal de este trabajo es implementar un método que nos permita identificar personas casi en tiempo real aplicando el análisis de componentes principales, conocido como PCA por sus siglas en inglés (Principal Component Analysis). Partimos de una base de datos de fotos, cada una asociada a un sujeto en particular y el objetivo es recibir una foto, que puede o no pertenecer a la base original, e identificar a qué sujeto pertenece.

Se utilizó el enfoque propuesto en “Eigenfaces for recognition” [2]. El mismo propone trabajar con las fotografías en 2 dimensiones sin reconstrucción geométrica de la tercera contando con un conjunto inicial -que llamaremos de entrenamiento- de imágenes pertenecientes al universo de rostros que se pretende reconocer. El enfoque propone trabajar con una transformación de ese espacio que permita reducir su dimensionalidad considerablemente sin perder la capacidad de distinguir a los elementos que viven en él.

Para $i = 1, \dots, n$ sea $x_i \in \mathbb{R}^m$ la i -ésima imagen de nuestra base de datos almacenada por filas en un vector. Vamos a restarle a cada uno de nuestros x_i el vector $\mu = (x_1 + \dots + x_n)/n$ para que tengan media igual cero. De esta forma construimos la matriz X que tiene en la i -ésima fila al vector $(x_i - \mu)^t$. Queremos extraer las direcciones que más información nos provean sobre cada una de estas imágenes, las direcciones que más las caracterizan. Para conseguirlo intentaremos conseguir una transformación de los datos que disminuya su redundancia, es decir, que disminuya la covarianza entre ellos. Entonces construimos la matriz de covarianza M_x :

$$M_x = \frac{1}{n-1} \hat{X}^t \hat{X}$$

Esta matriz tiene la varianza de cada variable, en nuestro caso los píxeles de las imágenes, en la diagonal y la covarianza entre ellos en las restantes posiciones. Usaremos como transformación a un cambio de base apropiado:

$$\hat{X}^t = P X^t$$

para obtenerlo diagonalizaremos la matriz de covarianza valiéndonos de que es simétrica y esto implica que es diagonalizable en la forma $M_x = P D P^{-1}$ donde D es una matriz diagonal. Luego, al diagonalizar, buscamos variables que tengan covarianza cero entre sí y la mayor varianza posible. Si aplicamos nuestro cambio de base a M_x nos queda:

$$M_x = \frac{1}{n-1} \hat{X}^t \hat{X} = \frac{1}{n-1} (P \hat{X}^t) (P \hat{X}) = P M_x P^t$$

Como M_x es simétrica existe V ortogonal tal que $M_x = V D V^t$:

$$M_x = P M_x P^t = P (V D V^t) P^t = (V^t V) D (V V^t) = D$$

habiendo tomado $P = V^t$. De esta forma vemos como las columnas v_j de la matriz V son los autovectores de M_x . Éstas van a ser las componentes principales de nuestros datos, la base de imágenes de entrenamiento. En la práctica, en caso de que la dimensión de M_x sea muy grande, es posible tomar sólo un subconjunto de las componentes principales, aquellas que capturen mayor proporción de la varianza de los datos.

De esta forma queda definida nuestra transformación característica por la aplicación del cambio de base a cada una de nuestras imágenes:

$$tc(x_i) = \bar{V}^t x_i = (v_1^t x_i, \dots, v_k^t x_i)$$

Ésta misma transformación característica se la aplicaremos a las imágenes que queramos identificar para evaluar a qué sujeto pertenecen.

2. Desarrollo

2.1. La implementación y algunas decisiones de diseño

La implementación consta de 2 fases. Primero está la fase de entrenamiento:

1. se obtiene la matriz $I \in \mathbb{R}^{n \times m}$ a partir de todas las imágenes de la base de entrenamiento. Para $i = 1, \dots, n$, llamamos x_i a la i -ésima fila de I que se corresponde con la i -ésima imagen de nuestra base vectorizada por filas. Cada posición de x_i se corresponde con un pixel de la imagen a la cual corresponde. Además, sabemos que $n = \text{nimgp} * \#\text{sujetos}$, siendo nimgp la cantidad de imágenes que tengo por cada sujeto y $\#\text{sujetos}$ la cantidad de sujetos distintos a los que pertenecen las imágenes de mi base de entrenamiento, y $m = \text{alto} * \text{ancho}$, siendo alto y ancho los respectivos anchos y altos de las imágenes medidos en pixeles.
2. se calcula la matriz A a partir de la matriz I . Sus dimensiones son iguales. Obtenemos el vector $\mu = (x_1 + \dots + x_n)/n$ que es el promedio de las imágenes. Para calcular A , en primer lugar restamos a cada una de las filas de I el vector μ . Obtenemos I' que contiene en la i -ésima fila al vector $(x_i - \mu)^t$. En segundo lugar obtenemos A como: $A = I' * \sqrt{n-1}$.
3. calculamos la matriz de covarianzas $M_x = A^t * A$. (Para este punto y el siguiente efectuamos un análisis alternativo que se puede ver en 2.2)
4. aplicamos el método de la potencia con deflación k veces sobre la matriz M_x . k es el número de componentes principales que nos interesa obtener. Conseguimos una matriz $B \in \mathbb{R}^{m \times k}$ cuya columna b_i se corresponde con el autovector a_i de la matriz M_x asociado al autovalor λ_i tal que $\lambda_1 < \lambda_2 < \dots < \lambda_k \leq \lambda_{k+1} \leq \dots \leq \lambda_m$. Esto es así por que el método de la potencia, si encuentra un autovalor, es el de mayor módulo. [1].
5. aplicamos la transformación característica a todas las imágenes de nuestra base de entrenamiento. La transformación característica de x_i está dada por

$$tc(x_i) = (b_1^t x_i, b_2^t x_i, \dots, b_k^t x_i) \in \mathbb{R}^k$$

y luego está la fase de identificación en la que recibimos una nueva imagen e identificamos a qué sujeto pertenece:

1. recibimos la imagen y la vectorizamos por filas de la misma forma que lo hicimos con las imágenes en nuestra base de entrenamiento en la primera fase y obtenemos $x_{\text{identificar}}$.
2. calculamos su transformación característica de la misma manera que lo hicimos en la fase de entrenamiento y con la misma matriz B

$$tc(x_{\text{identificar}}) = (b_1^t x_{\text{identificar}}, b_2^t x_{\text{identificar}}, \dots, b_k^t x_{\text{identificar}}) \in \mathbb{R}^k$$

3. decidimos a cuál sujeto pertenece por alguno de los dos métodos que describimos más adelante.

Los métodos que implementamos para decidir a qué sujeto pertenece una imagen luego de haber calculado su transformación característica son los siguientes:

Distancia mínima calculamos la distancia entre la transformación característica de la imagen que queremos identificar y todas las imágenes de nuestra base. Decidimos que la imagen nueva corresponde al sujeto al cual pertenece la imagen cuya transformación característica es la más cercana a la transformación característica de la nueva imagen. Es decir, con el de distancia mínima.

Distancia mínima al promedio calculamos el promedio de las distancias entre la transformación característica de la imagen que queremos identificar y las imágenes de cada uno de nuestros sujetos. De esta forma obtenemos una distancia promedio a los diferentes sujetos. Finalmente, decidimos que la imagen nueva corresponde al sujeto al cual pertenece el promedio de distancias más cercano a la transformación característica de la nueva imagen.

Aclaremos que el método de la potencia que implementamos recibe como vector inicial $b_0 = (1, \dots, 1)$.

La técnica de deflación arrastra el error generado por el método de la potencia, haciendo que los últimos autovalores tengan un error de cálculo mayor que los primeros. Este problema es mitigado por el hecho de que sólo tomamos los k mayores pero de todas formas probablemente quede reflejado en la experimentación. Además, tomamos $iter = 300$ en la implementación del método de la potencia, el máximo para que los tiempos de cómputo se mantuvieran en un rango aceptable, para obtener mayor precisión en el cálculo de los autovalores y así compensar el error arrastrado por la deflación.

2.2. Elección conveniente de la matriz de covarianzas para imágenes grandes

El método alternativo consiste en hallar una relación entre los autovalores y autovectores de $B = A^t A$, y los autovectores y autovalores de $C = A A^t$. Veamos primero como se escribe la matriz C . Si consideramos la descomposición en valores singulares de la matriz A :

$$C = U \Sigma V^t (U \Sigma V^t)^t = U \Sigma V^t V \Sigma^t U^t = U \Sigma \Sigma^t U^t$$

De esta manera encontramos la forma SVD de la matriz C . Y en consecuencia podemos intuir que U^t contiene los autovectores de C en sus columnas. Pasemos a confirmar la hipótesis anterior, sabemos que $u_i = \frac{1}{\theta_i} A v_i$ con $\theta_i = \sqrt{\lambda_i}$, y que $A^t A v_i = \lambda_i v_i$, por lo tanto:

$$C u_i = A A^t u_i = \frac{1}{\theta_i} A A^t A v_i = \frac{1}{\theta_i} A \lambda_i v_i = \lambda_i \frac{1}{\theta_i} A v_i = \lambda_i u_i.$$

De lo anterior se deduce directamente que λ_i es el autovalor asociado al autovector u_i para la matriz C . Veamos cómo se puede hallar v_i conociendo u_i :

$$u_i = \frac{1}{\theta_i} A v_i$$

multiplicando por A^t en ambos lados se obtiene:

$$A^t u_i = \frac{1}{\theta_i} A^t A v_i,$$

que es igual a:

$$A^t u_i = \frac{\lambda_i}{\theta_i} v_i$$

despejando y reemplazando θ_i :

$$\frac{\sqrt{\lambda_i}}{\lambda_i} A^t u_i = v_i$$

De lo anterior podemos describir el siguiente método para hallar los autovectores de $B = A^t A$:

1. Calcular $C = A A^t$.
2. Mediante el método de la potencia hallar los autovectores y autovalores de C , notar que los autovalores coinciden para ambas matrices, por lo que si se podía utilizar el método para B , también se puede hacer para C .
3. Para cada autovector u_i hallado, lo multiplico por $\frac{\sqrt{\lambda_i}}{\lambda_i} A^t$ para obtener v_i .

2.3. Comparación de la complejidad de los métodos

El método 1 trabaja con la matriz AA^t y el método 2 trabaja con la matriz A^tA . La matriz $A \in \mathbb{R}^{n \times m}$ con $m = \text{pixels}(\text{img})$ y $n = \text{nimgp} \times \text{\#personas}$. Por lo tanto como $A^t \in \mathbb{R}^{m \times n}$ nos queda que $AA^t \in \mathbb{R}^{n \times n}$ y $A^tA \in \mathbb{R}^{m \times m}$.

En el único punto que divergen los algoritmos es en el cálculo de los k mayores autovalores, y sus autovectores asociados, de $M_x = A^tA$, la matriz de covarianza. El método 1 aplica el método de la potencia con deflación directamente sobre la matriz M_x . El método 2 primero aplica el método de la potencia sobre la matriz $M_x^t = AA^t$ y después realiza un despeje para obtener los autovalores y autovectores de M_x . Por lo tanto tenemos que los dos métodos utilizan la misma función para calcular autovectores y autovalores: y el método 2 además utiliza una función para despejar cuyo

Algorithm 2.1 calcularAutovaloresYAutovectores(Matriz A)

```

for  $i = 1$  hasta  $k$  do
     $[v_i, \lambda_i] = \text{metodoDeLaPotencia}(A)$ 
     $A = \text{deflacionar}(A, v_i, \lambda_i)$ 
end for

```

pseudo-código podemos encontrar en la sección algoritmos. Por lo tanto, la complejidad del método 1 es (abusando de la notación asintótica):

$$O(\text{metodo1}) = k(O(\text{metodoDeLaPotencia}) + O(\text{deflacionar}))$$

y del método 2:

$$O(\text{metodo2}) = k(O(\text{metodoDeLaPotencia}) + O(\text{deflacionar})) + kO(\text{despejar})$$

La complejidad del método de la potencia es:

$$O(\text{metodoDeLaPotencia}) = \text{iter}(O(Ax) + O(\frac{y}{\|y\|}))$$

$A \in \mathbb{R}^{n \times n}$ y $x \in \mathbb{R}^n$ entonces:

$$O(Ax) = n(n \times \text{mult} + (n-1) \times \text{sums}) = (n^2) \times \text{mult} + n(n-1) \times \text{sums} = O(n^2)$$

Si $y \in \mathbb{R}^n \Rightarrow \frac{y}{\|y\|} = O(n)$ por lo tanto reemplazando por los nuevos valores:

$$O(\text{metodoDeLaPotencia}) = \text{iter}(O(n^2) + O(n)) = \text{iter}O(n^2) = O(n^2)$$

porque iter es un valor fijo que elegimos de antemano. La complejidad de aplicar deflacionar con $v_i \in \mathbb{R}^n$ y $A \in \mathbb{R}^{n \times n}$ es:

$$O(\text{deflacionar}) = O(A - \lambda v_1 v_1^t) = n^2 \times \text{restas} + O(\lambda v_i v_i^t)$$

seguimos reemplazando:

$$n^2 \times \text{restas} + n^2 \times \text{mults} + O(v_i v_i^t) = n^2 \times \text{restas} + n^2 \times \text{mults} + n^2 \times \text{mults}$$

entonces nos queda que:

$$O(\text{deflacionar}) = n^2 \times \text{restas} + 2n^2 \times \text{mults} = O(n^2)$$

Por lo tanto la complejidad del método 1 queda de la siguiente forma:

$$O(\text{metodo1}) = k(O(\text{metodoDeLaPotencia}) + O(\text{deflacionar})) = k(O(n^2) + O(n^2)) = O(n^2)$$

y la complejidad del método 2 es:

$$O(\text{metodo2}) = O(\text{metodo1}) + O(\text{despejar})$$

Y la complejidad de despejar la planteamos como:

$$O(\text{despejar}) = O\left(\frac{\sqrt{\lambda_i}}{\lambda_i} A^t v\right)$$

con $v \in \mathbb{R}^n$ y $A \in \mathbb{R}^{n \times n}$ nos queda:

$$n \times \text{mults} + O(A^t v) = n \times \text{mults} + (n(n \times \text{mults} + (n - 1) \times \text{sums}))$$

lo que nos deja con:

$$O(n) + nO(n) = O(n^2)$$

reemplazando en la complejidad del método 2 obtenemos que:

$$O(\text{metodo2}) = O(\text{metodo1}) + O(\text{despejar}) = O(n^2) + O(n^2) = O(n^2)$$

Concluimos que el método 1 y el método 2 tienen la misma complejidad $O(n^2)$ tomando a n como el tamaño de la matriz A de entrada. Como habíamos analizado en el comienzo el tamaño de la matriz A que recibe cada uno es diferente. Para el método 1 es $n = \text{pixels}(\text{img})$ y para el método 2 es $n = \text{nimgp} \times \#\text{personas}$. Por lo tanto cuál de los dos cueste menos va a depender de la relación entre la resolución de las imágenes de nuestra base de entrenamiento y la cantidad de imágenes que tenemos en total. En nuestro caso particular la resolución de las imágenes excede ampliamente a la cantidad de imágenes. Nuestra base de entrenamiento tiene 410 fotos y en general usamos una resolución de $112 \times 92 = 10304$. Por lo tanto:

$$\text{nimgp} \times \#\text{personas} \ll \text{pixels}(\text{img}) \Rightarrow O(\text{metodo2}) \ll O(\text{metodo1})$$

2.4. Algoritmos

Algorithm 2.2 metodoDeLaPotencia(Matriz A , Vector v , Int $iter$)

```

for  $i = 1$  hasta  $iter$  do
     $y = Ax$ 
     $x = \frac{y}{\|y\|}$ 
end for

```

Algorithm 2.3 deflacionar(Matriz A , Vector $autovector$, Double λ)

```

 $V = vv^t$ 
 $B = A - \lambda V$ 

```

Algorithm 2.4 despejar(Matriz A , Vector v , Double λ)

```

 $u = v \frac{\sqrt{\lambda_i}}{\lambda_i} A^t$ 

```

Algorithm 2.5 identificarDistanciaMinima(Vector tc_{nueva} , Matriz TC)

```
for  $i = 1$  hasta  $\#filas(TC)$  do
     $tc_{base} = fila(i, TC)$ 
     $diff = tc_{nueva} - tc_{base}$ 
     $dist = norm(diff)$ 
    if  $i == 1$  then
         $minimo = dist$ 
    end if
    if  $dist < minimo$  then
         $minimo = dist$ 
    end if
end for
```

Algorithm 2.6 identificarDistanciaPromedioMinima(Vector tc_{nueva} , Matriz TC , Int $iSujetos$, Int $nimgp$)

```
for  $i = 1$  hasta  $iSujetos$  do
     $acum = 0$ 
    for  $j = 1$  hasta  $nimgp$  do
         $tc_{base} = fila(nimgp * i + j, TC)$ 
         $diff = tc_{nueva} - tc_{base}$ 
         $acum += norm(diff)$ 
    end for
     $iDistanciaPromedio = \frac{acum}{nimgp}$ 
    agregar( $iDistanciaPromedio$ ,  $aDist$ )
end for
sujeto = min( $aDist$ )
```

2.5. Tests

En esta sección haremos una introducción a las experimentaciones que decidimos realizar.

2.5.1. Cantidad de componentes principales vs tasa de efectividad

El objetivo de la siguiente experimentación es analizar cómo afecta el número de componentes principales utilizado a la tasa de efectividad que obtenemos al identificar fotos. Cuantas más componentes principales tomamos más información tenemos. Sin embargo, los k calculamos autovalores que tomamos son los k de mayor valor absoluto, es decir, los que más información reflejan. Por lo tanto es de esperar que cada componente principal que agregamos nos aporte menos información que la anterior. Es decir, que la tasa de efectividad aumente junto con la cantidad de componentes principales utilizadas pero a una velocidad sublineal. Además, sabemos que el método de la potencia con deflación arrastra error, esto implica que la exactitud de los autovalores obtenidos es cada vez menor $exactitud(\lambda_i) > exactitud(\lambda_{i+1})$. Para la experimentación vamos a mantener la cantidad de sujetos, 40, y la resolución de las imágenes, 112×92 , constantes. El análisis lo vamos a realizar considerando hasta 45 componentes principales. Los tests se correrán tomando $nimgp = 3, 6, 9$ caras de entrenamiento. Para cada combinación de k y $nimgp$ corremos el test 10 veces, tomando al azar las fotos de la base de entrenamiento y las utilizadas para identificación. Calculamos la tasa de efectividad para los 10 y nos quedamos con el promedio. Vamos a intentar identificar a todas las fotos restantes de la base para poder tener más casos.

2.5.2. Análisis de la tasa de eficiencia en función de la cantidad de personas

En este experimento se intenta analizar como influye la cantidad de personas en la tasa de eficiencia del reconocimiento de imágenes. Para esto vamos a concentrarnos solo en la variable

de la cantidad de personas por lo que hay que fijar el resto de los parámetros en sus valores óptimos. La experimentación se va a hacer sobre la base de imágenes grandes, con cinco imágenes por persona (valor generoso pero dentro de los límites reales), y con quince componentes principales y con la técnica de reconocimiento más efectiva. Se espera que cuantas más personas haya (y en consecuencia más imágenes), la probabilidad de que haya imágenes similares de distintas personas o incluso personas similares es más alta, por lo que tasa de reconocimiento va a ser menor.

2.5.3. *nimgp* vs Tasa de efectividad

En la siguiente experimentación analizaremos cómo afecta a la tasa de efectividad la cantidad de imágenes que tomamos por sujeto, *nimgp*, para la base de entrenamiento a la tasa de efectividad. Nos interesa realizar este test porque tiene una relevancia grande en la práctica: conocer qué valor *nimgp* necesito como mínimo para poder alcanzar una tasa de efectividad aceptable para el contexto en el que lo aplique.

La base total con la que contamos contiene 41 sujetos y 10 fotos por cada uno de ellos. La consideramos pequeña por lo cual vamos a intentar abarcar la mayor cantidad de combinaciones posibles entre *nimgp*, las fotos que tomamos de entrenamiento y las que tomamos para identificar test. Es por eso que vamos a tomar todos los *nimgp* posibles, $nimgp = 1, \dots, 9$, y para cada uno de ellos vamos a correr 20 veces los tests eligiendo aleatoriamente las imágenes que pertenecen al conjunto de entrenamiento y las que vamos a utilizar para la fase de identificación para luego quedarnos con el promedio de las tasas de efectividad. El número de sujetos va a ser 40, vamos a identificar todas las fotos que no pertenezcan a la base de entrenamiento, la resolución la mantenemos en 112×92 . El otro valor que variaremos para evaluar si afecta a la tasa obtenida para cada *nimgp* es la cantidad de + componentes principales, tomaremos $k = 15, 30, 45$.

Esperamos que al tomar *nimgp* mayores la tasa de efectividad aumente ya que estamos cubriendo más casos del rostro. Al cubrir un mayor espectro de posibilidades del rostro estamos acotando la probabilidad de que la cara que nos presenten sea suficientemente distinta como para que no podamos identificarla correctamente.

2.5.4. Resolución de las imágenes vs Tasa de efectividad

En esta experimentación se analizará la tasa de efectividad en función de la la resolución de las imágenes utilizadas. Para esto se utilizaron 5 escenarios posibles que corresponden a diferentes configuraciones de los parámetros. De esta manera se definen las distintas condiciones del experimento en:

- Peísima: Se utilizan 10 componentes y 1 imagen por persona.
- Mala: Se utilizan 10 componentes y 3 imágenes por persona.
- Regular: Se utilizan 15 componentes y 5 imágenes por persona.
- Buena: Se utilizan 30 componentes y 7 imágenes por persona.
- Excelente: Se utilizan 30 componentes y 9 imágenes por persona.

La cantidad de personas se fijó en 41 ya que este parámetro no modifica la condición de los escenarios tanto como los otros dos. Se realizaron varias experimentaciones para cada escenario y para cada resolución tomando imágenes al azar para cada sujeto. El método de identificación utilizado fue el de distancia mínima por que es el más efectivo. Este experimento intenta probar como afecta la resolución de las imágenes a la tasa de efectividad. Es natural pensar que cuanto mayor sea la resolución más información tendrán las imágenes por lo que resultados serán más exactos.

2.5.5. Comparación de los métodos de identificación

La siguiente experimentación tiene por objetivo comparar la tasa de efectividad entre el método de identificación por distancia mínima y el método de identificación por distancia promedio mínima. Además, nos interesa determinar si la combinación óptima, en términos de la tasa de efectividad obtenida, es la misma para los dos métodos de identificación. Correremos los tests con las mismas variables que en la primera experimentación pero aplicando el método de identificación por distancia promedio mínima para la fase de identificación.

3. Resultados

3.1. Test de tasa de eficiencia en función de la cantidad de componentes principales consideradas

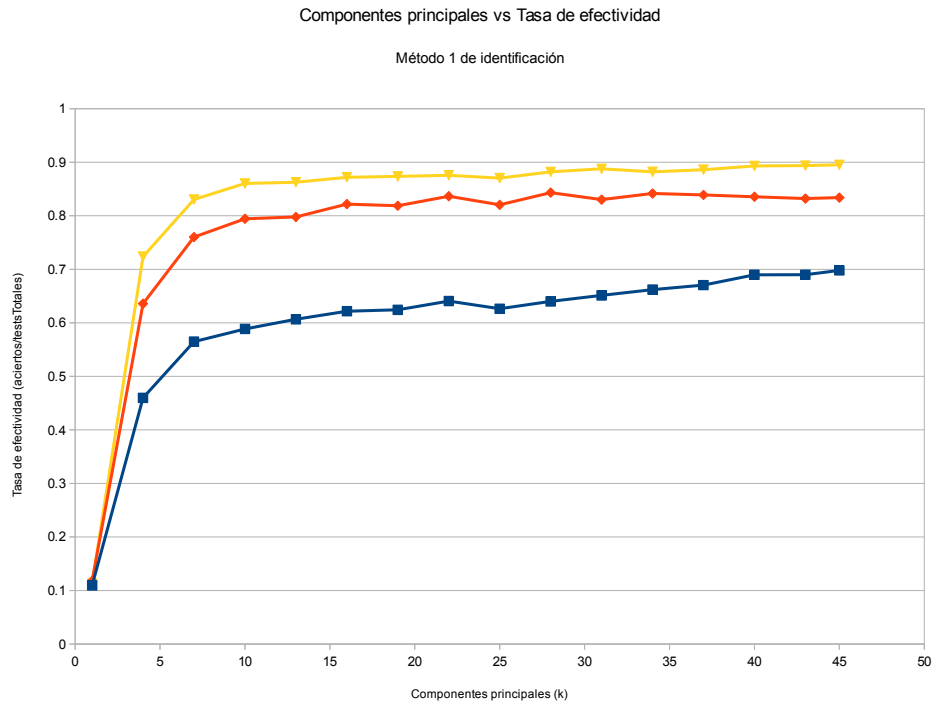


Figura 1: El método 1 de identificación se corresponde con el de Distancia Mínima

Línea azul: $nimgp = 3$. Línea roja: $nimgp = 6$. Línea amarilla: $nimgp = 9$.

3.2. Test de tasa de eficiencia en función de la cantidad de personas

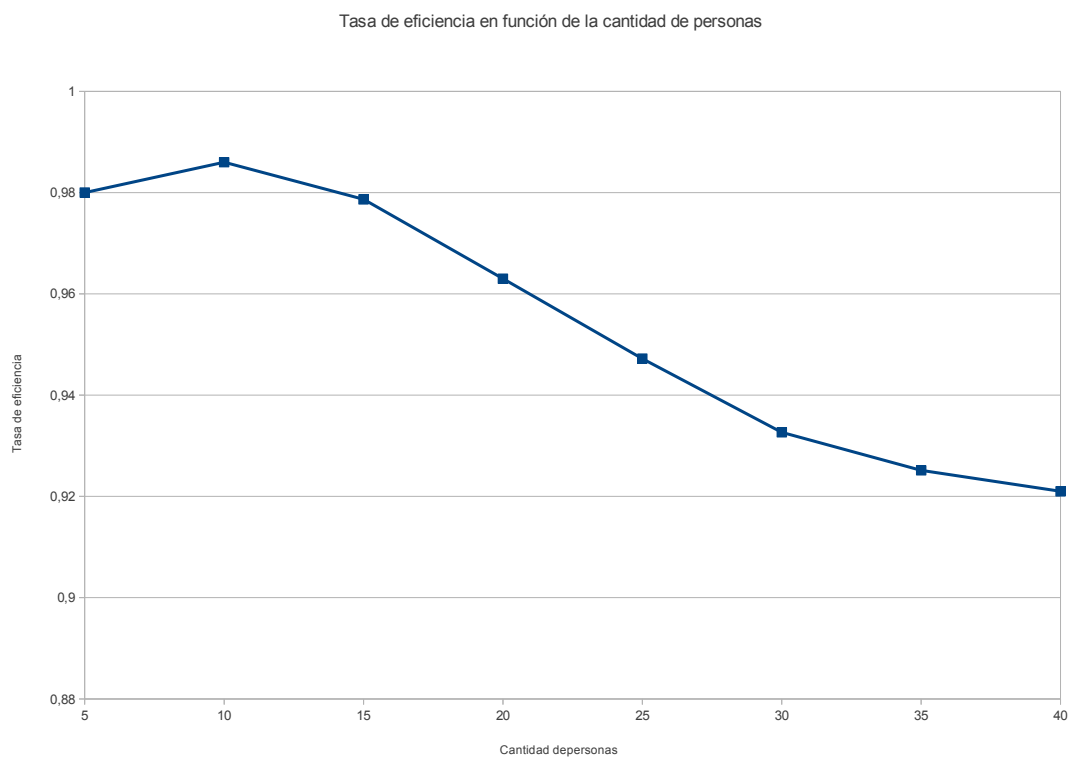


Figura 2: Tasa de eficiencia en función de la cantidad de personas.

3.3. Test de tasa de eficiencia en función de nimgp

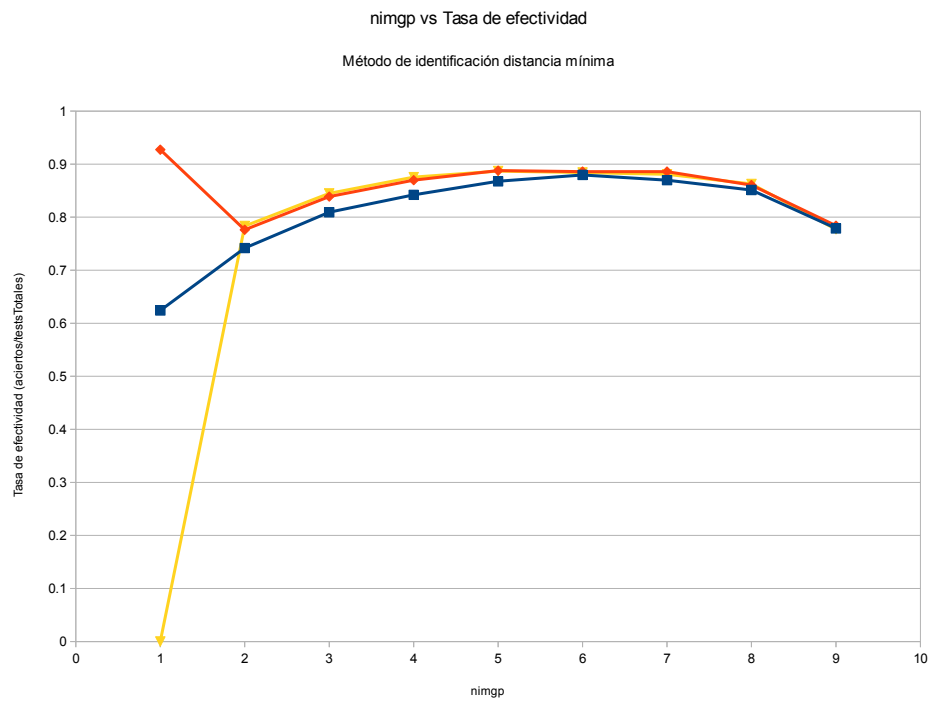


Figura 3: Se graficó para 15, 30 y 45 componentes principales

Línea azul: $k = 15$. Línea roja: $k = 30$. Línea amarilla: $k = 45$.

3.4. Test de comparación de los métodos de identificación

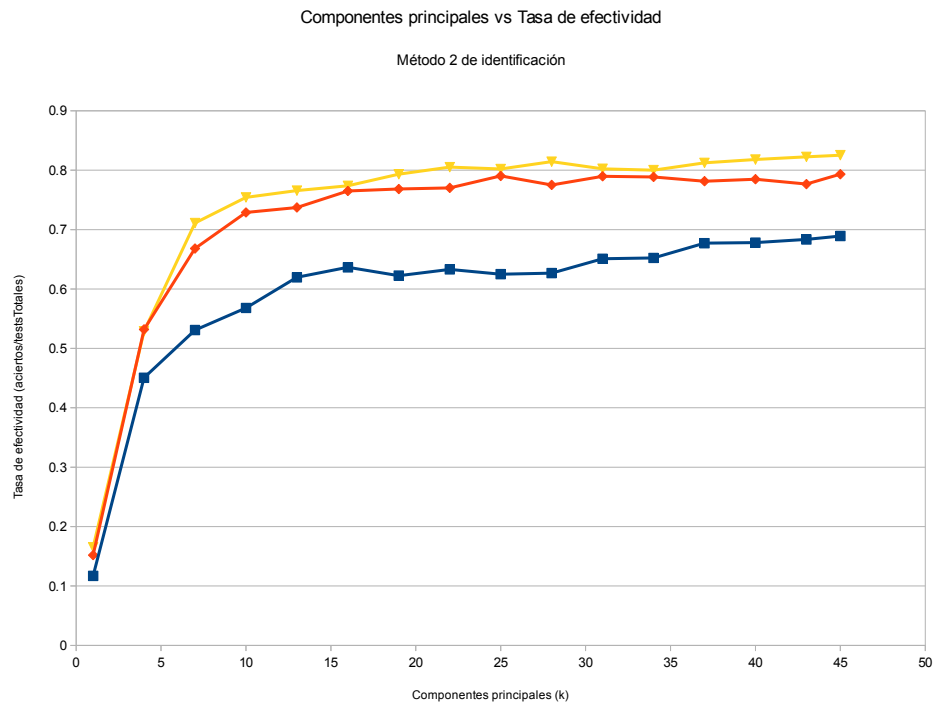


Figura 4: El método 2 de identificación se corresponde con el de Distancia Mínima Promedio

Línea azul: $nimgp = 3$. Línea roja: $nimgp = 6$. Línea amarilla: $nimgp = 9$.

3.5. Test de tasa de eficiencia en función de la resolución de las imágenes

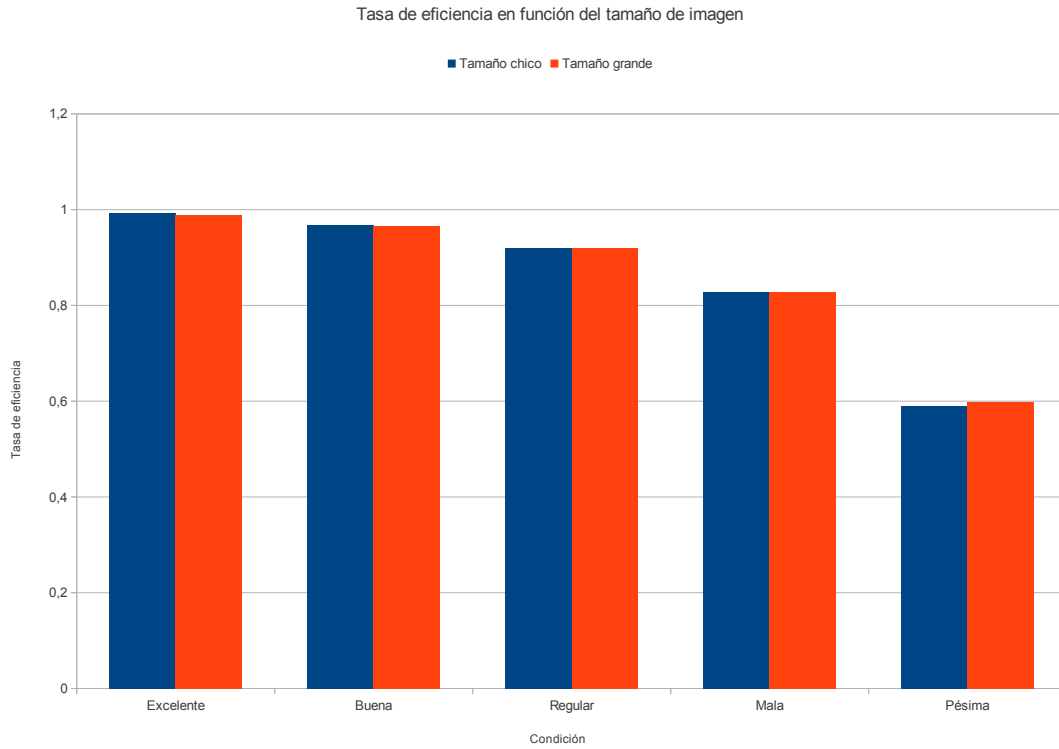


Figura 5: Tasa de eficiencia en función de la resolución para distintos escenarios.

4. Discusión

4.1. Análisis de resultados de tasa de efectividad en función de las componentes principales

Se ve claramente como la tasa de efectividad aumenta al incrementarse el número de componentes principales utilizadas. Éste es un comportamiento que esperábamos. Sin embargo, nos sorprende cuán rápido disminuye la velocidad a la que crece la tasa de efectividad. Para $k = 12$ podemos ver que ya alcanzamos un 90 % de la tasa de efectividad total que conseguimos hasta donde alcanzaron nuestros tests, $k = 45$. Incrementar las componentes principales a partir de ese momento logra aumentar la tasa de efectividad sólo un 10 % más. Éste comportamiento interpretamos que se debe a que las primeras 12 componentes principales sintetizan el 90 % de la información. Se desprende de dicha interpretación el grado de importancia que tiene calcular los autovalores en orden descendente por magnitud de su módulo. Si tenemos en cuenta que la matriz podría llegar a tener 10304 autovalores diferentes, dada la resolución de las imágenes, las 12 componentes principales representan menos de un 0,2 % de las mismas.

A nivel práctico podemos aconsejar que a menos que los requerimientos sobre la tasa de efectividad sean muy demandantes basta con tomar unas pocas componentes principales para lograr una buena tasa de efectividad.

Finalmente, podemos observar que la forma de las 3 curvas es prácticamente igual, simplemente sufren un desplazamiento sobre el eje vertical. Esto nos lleva a intuir que tomar mayores *nimgp* nos provee una mayor tasa de efectividad sin necesidad de aumentar el k que utilizamos.

4.2. Análisis de la tasa de eficiencia en función de la cantidad de personas

Como se puede observar en los resultados la tasa de efectividad disminuye a medida que aumenta la cantidad de personas, posiblemente debido a que, como se mencionó anteriormente, la probabilidad de que haya más sujetos parecidos entre sí o imágenes similares de distintos sujetos. Además la probabilidad de aciertos es menor al aumentar la cantidad de sujetos, es decir, si hay 2 sujetos la probabilidad de acierto sin ningún tipo de información de $\frac{1}{2}$, en cambio con 10 sujetos es $\frac{1}{10}$, en general la probabilidad de acertar sin ningún tipo de información es $\frac{1}{n}$ donde n es la cantidad de personas. Sería interesante analizar el comportamiento asintótico del gráfico para hallar una cota inferior para la tasa de eficiencia y deducir si el algoritmo puede ser arbitrariamente malo. Lamentablemente solo se dispone de 40 imágenes, por lo que solo podemos dar una aproximación de la tasa de eficiencia para 40 o menos imágenes.

4.3. Análisis de la tasa de efectividad en función de $nimgp$

Lo primero que nos llama la atención de estos resultados es la gran diferencia que devuelve en la tasa de efectividad que se obtuvo tomando $nimgp = 1$ y variando $k = 15, 30, 45$. No estamos seguros a qué se debe este comportamiento. Intuimos que como sólo se toma una foto por sujeto para la base de entrenamiento los resultados dependen mucho de la foto que se toma. Se intentó reducir este tipo de ruido realizando varias veces el mismo test con seleccionando aleatoriamente las imágenes y aparentemente no tuvimos éxito para el caso más sensible.

Se puede apreciar claramente como la tasa de efectividad aumenta a medida que incrementamos $nimgp$, comportamiento que predijimos, sin embargo, no nos esperábamos que luego de cierto umbral la tasa de efectividad comience a decrecer. Sin tener en cuenta el caso $nimgp = 1$, vemos claramente que la curva sigue una forma de parábola abriéndose hacia abajo, alcanzando su máximo en $nimgp = 6$. No estamos seguros a qué se debe el decrecimiento que sufre la tasa de efectividad para valores $nimgp > 6$.

4.4. Análisis de comparación de los métodos de identificación

Vemos que las curvas tienen la misma forma que en la primera experimentación, lo cual nos dice que los dos métodos de identificación se comportan de la misma forma ante la variación de la cantidad de componentes principales utilizadas. Además, se comportan de la misma forma ante los cambios en $nimgp$ ya que las 3 curvas aparecen una sobre la otra en el mismo orden en los dos gráficos. Sin embargo, podemos ver que para el método de identificación por distancia mínima las curvas roja y amarilla, correspondientes a $nimgp = 6$ y $nimgp = 9$ respectivamente, a partir de $k = 10$ obtienen tasas de efectividad superiores a 0,8. En cambio, para el método de distancia promedio mínima a penas superan la tasa de efectividad 0,8. Concluimos que el método de identificación por distancia mínima alcanza una mayor tasa de efectividad utilizando la misma cantidad de componentes principales y valores de $nimgp$. Sugerimos utilizar éste método para cualquier aplicación práctica por sobre el de distancia promedio mínima.

4.5. Análisis de la tasa de eficiencia en función de la resolución de las imágenes

Los resultados obtenidos si bien no son los que se esperaban implican ciertas cuestiones interesantes. Primero, la resolución de imágenes son prácticamente idénticas para todas las condiciones por lo que se puede suponer que para ambas resoluciones la información que contienen las imágenes es suficiente y necesaria para que el análisis de componentes principales funcione de la manera esperada. Si bien nuestra suposición inicial no era correcta, es decir que a una resolución mayor se obtiene una tasa más alta, el error partió de suponer que la resolución más chica era lo suficiente mala como para arrojar peores resultados. Esta percepción errónea provino de que las imágenes chicas son algo más difícil (pero no imposible con un poco de tiempo) de identificar a simple vista (con la cantidad apropiada a zoom).

5. Conclusiones

La aplicación de PCA ha demostrado ser una decisión acertada. Su poder quedó claro en la primera experimentación, tasa de efectividad en función de las componentes principales, cuando tomando las componentes de mayor valor absoluto nos bastó con tomar menos de un 0,2 % de las mismas para obtener hasta un 80 % de efectividad.

No pudimos apreciar cómo se comporta el arrastre de error de la deflación integrada al método de la potencia. Una experimentación posibles para analizarlo sería ver qué pasa al aumentar el número de componenetes principales que se utiliza hasta valores cercanos al máximo posible. La resolución de las imágenes debería ser reducida ya que de lo contrario el tiempo de ejecución puede llegar a valores poco prácticos.

El método de la potencia con deflación se mostró práctico y útil, ya que con tan solo 300 iteraciones pudimos compensar en buena medida el arrastre de error producido por la deflación. Esto lo intuimos a partir de que la tasa de efectividad al aumentar la cantidad de componentes principales en siempre continuó creciendo. Es decir, el error arrastrado fue compensado minimizado lo suficiente por la precisión obtenida con el método de la potencia como para no perturbar el crecimiento de la tasa de efectividad.

Por último, nos gustaría destacar un detalle que nos resultó muy interesante, la elección de la matriz de covarianza. La diferencia abismal en complejidad que puede introducir dependiendo el contexto elegir una u otra forma de plantear los sistemas nos dejó una fuerte impresión. Nos agrada mucho ver el gran impacto que pueden tener en la práctica los contenidos teóricos desarrollados en la materia.

Referencias

- [1] BURDEN, R. L., AND FAIRES, J. D. *Numerical Analysis*. Brooks Cole, 2000.
- [2] TURK, M., AND PENTLAND, A. Eigenfaces for recognition. *Journal of cognitive neuroscience* 3, 1 (1991), 71–86.