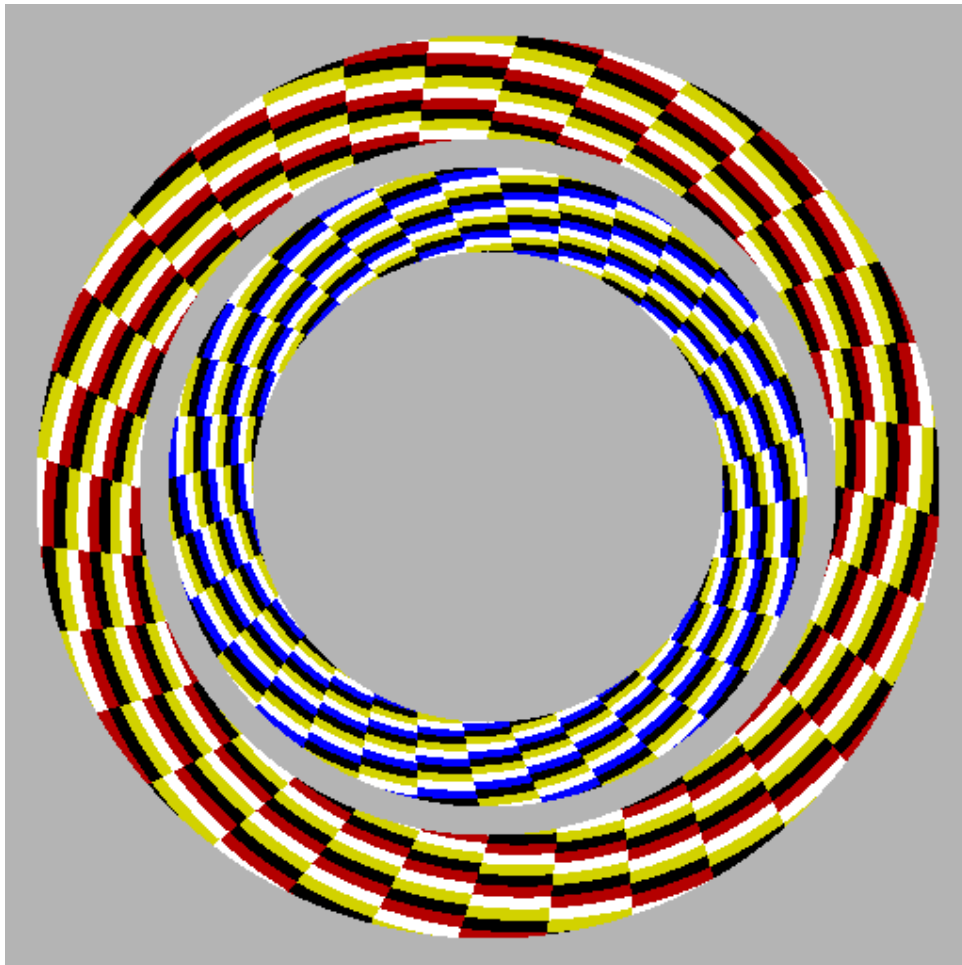


Práctica 8:

Sistema de reconocimiento de caras basado en PCA.

Objetivo:

- Realizar un sistema de reconocimiento de caras basado en PCA utilizando las imágenes capturadas en la práctica anterior.
-



1. Sistema de reconocimiento de caras basado en PCA

El objetivo de esta práctica es realizar un sistema de reconocimiento de caras basado en PCA utilizando las imágenes capturadas en la práctica anterior.

Se partirá de la función de matlab, *ejemplo.m*, que realiza el proceso de extracción de características mediante PCA de las imágenes de la librería ORL, vista también en la práctica anterior.

```
function p=ejemplo(c1,c2,c3)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%Ejemplo de PCA con caras
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%cl c2 c3 son las coordenadas a visualizar
%
% ejemplo(1,2,3)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

N=8 %numero de individuos en el sistema de reconocimiento
Nv=6 %numero de eigenvectores del subespacio

%imagenes utilizadas para generar el subespacio PCA
x1=imread('orl-faces/s1/1.pgm');
x2=imread('orl-faces/s2/1.pgm');
x3=imread('orl-faces/s3/1.pgm');
x4=imread('orl-faces/s4/1.pgm');
x5=imread('orl-faces/s5/1.pgm');
x6=imread('orl-faces/s6/1.pgm');
x7=imread('orl-faces/s7/1.pgm');
x8=imread('orl-faces/s8/1.pgm');

for i=1:N
comando=sprintf('x%d=x%d(:)';i,i);
eval(comando);
end

%matriz de datos X
X=[x1 x2 x3 x4 x5 x6 x7 x8]; %cada columna es una cara
X=double(X);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PCA %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[V,D,Media] = pc_eectores(X,Nv);

for i = 1:N
    Xm(:,i) = X(:,i) - Media; %Quito la media los datos para proyectar
end;

p=V'*Xm; %proyeccion de las caras generadoras en el subespacio PCA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Proyeccion del resto de caras de cada individuo en el subespacio PCA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for num_cara=1:N

for i=1:10
file_test=sprintf('orl-faces/s%d/%d.pgm',num_cara,i);
comando=sprintf('t%d=imread(file_test);',i);
eval(comando);
comando=sprintf('t%d=t%d(:);',i,i);
eval(comando);
end

T=[t1 t2 t3 t4 t5 t6 t7 t8 t9 t10]; %cada columna es una cara
T=double(T);

for i = 1:10
Tm(:,i) = T(:,i) - Media; %Quito la media los datos para proyectar
end;

proy=V'*Tm;

comando=sprintf('p%d=proy;',num_cara);
eval(comando);

end

%En las variables p1, p2,... p8 se almacenan las proyecciones de las caras

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% VISUALIZACION DE LAS CLASES

%graficos 3D
figure
plot3(p1(c1,:),p1(c2,:),p1(c3,),'k*')
hold on
plot3(p2(c1,:),p2(c2,:),p2(c3,),'b*')
plot3(p3(c1,:),p3(c2,:),p3(c3,),'g*')
plot3(p4(c1,:),p4(c2,:),p4(c3,),'r*')
plot3(p5(c1,:),p5(c2,:),p5(c3,),'y*')
plot3(p6(c1,:),p6(c2,:),p6(c3,),'bx')
plot3(p7(c1,:),p7(c2,:),p7(c3,),'ko')
plot3(p8(c1,:),p8(c2,:),p8(c3,),'mo')
grid
legend('S1','S2','S3','S4','S5','S6','S7','S8')

%graficos 2D
figure
plot(p1(c1,:),p1(c2,),'k*')
hold on
plot(p2(c1,:),p2(c2,),'b*')
plot(p3(c1,:),p3(c2,),'g*')
plot(p4(c1,:),p4(c2,),'r*')
plot(p5(c1,:),p5(c2,),'y*')
plot(p6(c1,:),p6(c2,),'bx')
plot(p7(c1,:),p7(c2,),'ko')
plot(p8(c1,:),p8(c2,),'mo')
grid
legend('S1','S2','S3','S4','S5','S6','S7','S8')

```

Al ejecutar la función anterior,

```
>> ejemplo(1,2,3)
```

se obtienen los gráficos de las figuras 1 y 2. En dichos gráficos se representan las proyecciones de las 10 caras de los 8 primeros individuos de la librería ORL en el subespacio PCA generado con la cara 1 (*1.pgm*) de cada uno de ellos.

El vector original de cada una de las imágenes tenía 10304 componentes, puesto que la imagen era de tamaño 112 x 92. Mediante la transformación PCA se consigue reducir considerablemente el tamaño de los patrones, así en los gráficos 1 y 2, se representan los patrones con sólo 2 y 3 componentes, y sin embargo, aún con un vector tan reducido, las clases entre los diferentes individuos están considerablemente separadas.

El número de vectores que obtiene el método PCA para el caso de imágenes (donde la dimensión del vector original es muy superior al número de ejemplos) es igual o inferior al número de ejemplos, es decir, en el caso de tener 8 individuos (8 clases) el subespacio PCA será de 8 dimensiones o inferior.

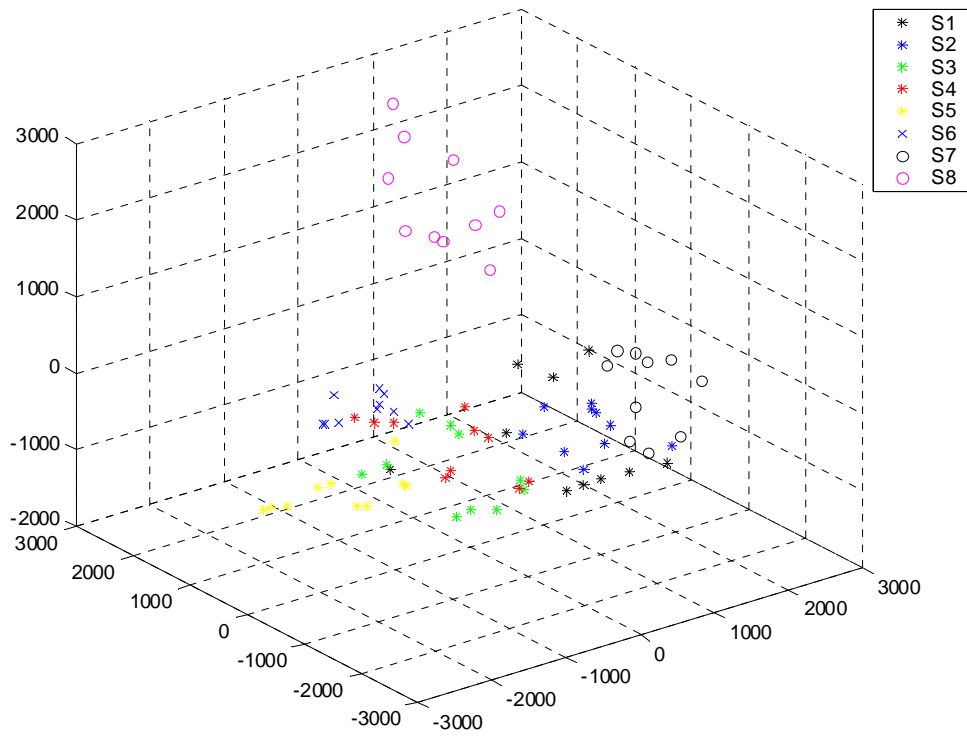


Figura 1: Gráfico 3D con las clases de los 8 individuos.

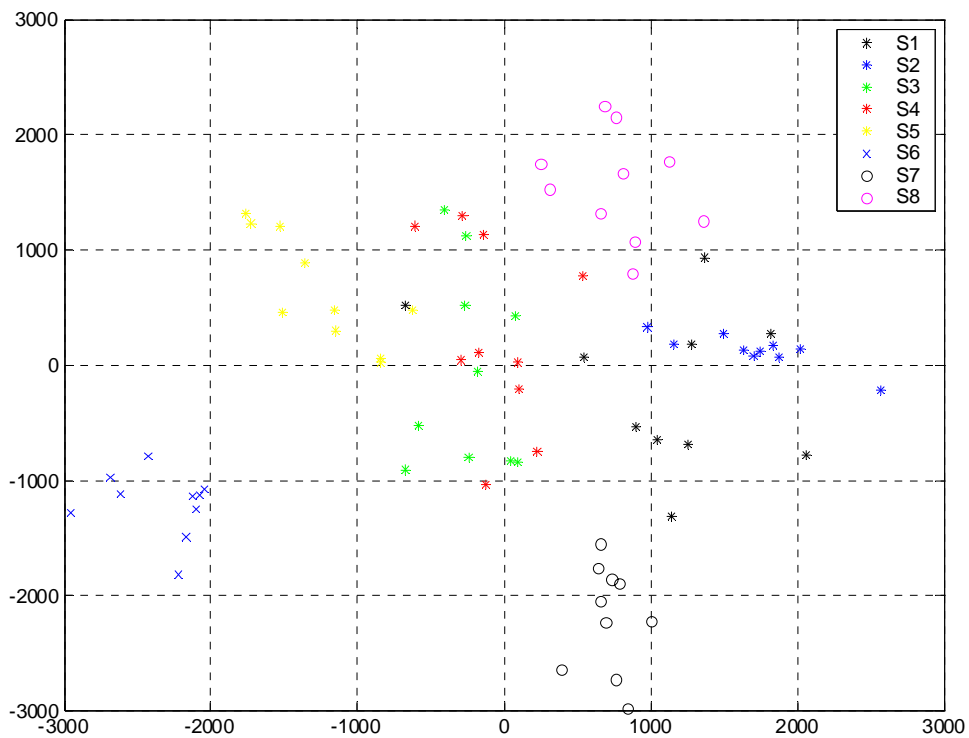


Figura 2: Gráfico 2D con las clases de los 8 individuos.

2. Trabajo propuesto

El trabajo a realizar consiste en la modificación de la función *ejemplo.m* con las caras de los alumnos capturadas y procesadas en la práctica 2.

El programa modificado puede ser una *función* o *script* y debe incluir los siguientes procesos:

1. Visualización en un gráfico de la imagen de cada alumno utilizada para la generación del subespacio PCA (similar a la de la figura 1 de la práctica 2).
2. Visualización en un gráfico de las imágenes de los eigenvectores obtenidos (similar a la de la figura 1 de la práctica 2).
3. Gráfico 2D con las 2 componentes que mejor separan a un alumno determinado respecto a los de su grupo de prácticas (3 o 4 personas). Dibujar el gráfico con sólo 2 clases, una para el alumno en concreto y la otra para el conjunto de las otras personas
4. Gráfico 2D con las 2 componentes que mejor separan a un alumno determinado respecto al resto de los alumnos de IARP (unas 20 personas, aprox.). Dibujar el gráfico con sólo 2 clases, una para el alumno en concreto y la otra para el conjunto de las otras personas.

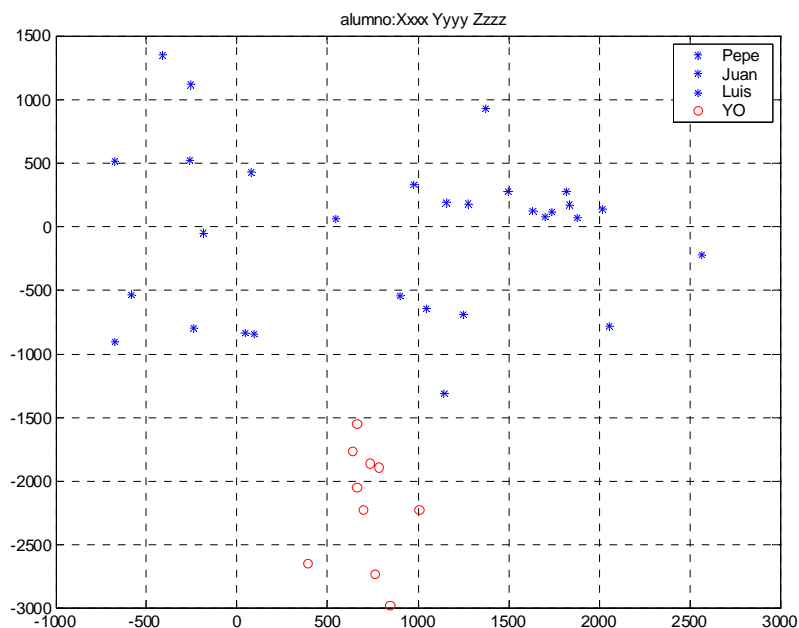


Figura 3: Ejemplo del gráfico con 2 clases

A incluir en el **informe** de la práctica:

- script o función de matlab
- gráficos 2D con las 2 clases: una clase estará formada por las caras del *alumno del informe* y la otra, por las caras del resto de los alumnos. Se utilizarán 3 conjuntos de imágenes de menor a mayor grado de dificultad en el reconocimiento:
 - Escenario 1 (caras: 1-10)
 - Escenarios 1 y 2 (caras: 1-20)
 - Escenarios 1, 2, 3 y 4 (caras 1_30)

En todos los listados de programas y en el título de cada gráfico ha de aparecer el nombre y apellidos del alumno
(*title('alumno Xxxx Yyyy Zzzz')*)