



Universidad  
de Huelva

## SISTEMAS DE PERCEPCIÓN

Grado en Ingeniería Informática – itinerario Computación

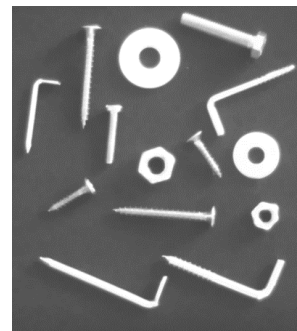
**Dpto. de Ingeniería Electrónica, Sistemas Informáticos y Automática**

**CURSO 2019/20 – CONVOCATORIA DE JUNIO**

**EVALUACIÓN PRÁCTICA – SEGUNDO BLOQUE ASIGNATURA**

### EJERCICIO 2: 8 PUNTOS

Implementar un algoritmo que reconozca entre arandelas, tuercas, alcayatas lisas, alcayatas tipo rosca, tornillos cabeza hexagonal, y tornillos tipo rosca, en imágenes de intensidad tal como esta:



#### ESPECIFICACIONES DE DISEÑO PARA LA IMPLEMENTACIÓN DEL ALGORITMO:

1. Reconocimiento inicial por N° de Euler entre las siguientes clases: arandelas-tuercas y alcayatas-tornillos (de los dos tipos, alcayatas lisas y tipo rosca, y tornillos cabeza hexagonal y tipo rosca).
2. Reconocimiento de las clases de cada conjunto:

##### 1.1. Reconocimiento entre **arandelas y tuercas**:

- Descripción matemática: **tres mejores descriptores de forma conjunta**
- Técnica de clasificación: **mínima distancia Euclidea**

Para reconocer entre alcayatas y tornillos, se aplicará una estrategia de clasificación en dos etapas:

##### 1.2. Reconocimiento entre **alcayatas lisas, alcayatas tipo rosca y tornillos** (los dos tipos de tornillos se consideran como una única clase):

- Descripción matemática: **Momento de Hu número 3, Descriptor Fourier número 6 y Descriptor de Fourier número 9**
- Técnica de clasificación: **k-NN con k=3, basado en distancia Euclidea.**

##### 1.3. Reconocimiento entre **tornillos de cabeza hexagonal y tornillos tipo rosca**:

- Descripción matemática: **descriptores de Fourier números 1, 3 y 9**
- Técnica de clasificación: **k-NN con k=5, basado en distancia Mahalanobis.**



Universidad  
de Huelva

## SISTEMAS DE PERCEPCIÓN

Grado en Ingeniería Informática – itinerario Computación

**Dpto. de Ingeniería Electrónica, Sistemas Informáticos y Automática**

**CURSO 2019/20 – CONVOCATORIA DE JUNIO**

**EVALUACIÓN PRÁCTICA – SEGUNDO BLOQUE ASIGNATURA**

### MATERIAL DISPONIBLE:

- **Conjunto de datos entrenamiento:** X-Y (fichero *conjunto\_datos\_entrenamiento.mat*)
  - X: matriz de tantas filas como objetos había disponibles en las imágenes de entrenamiento y tantas columnas como descriptores matemáticos se han utilizado para caracterizar cada objeto. En nuestro caso 23:

#### % Descriptores:

- % **Compacticidad:** Columna 1
- % **Excentricidad:** Columna 2
- % **Solidez\_CHull(Solidity):** Columna 3
- % **Extension\_BBBox(Extent):** Columna 4
- % **Extension\_BBBox(Invariante Rotacion):** Columna 5
- % **Momentos de Hu:** Hu1-Hu7: Columnas 6-12
- % **Descriptores de Fourier:** DF1-DF10: Columnas 13-22
- % **Número de Euler:** Columna 23

**Observación:** estos datos se facilitan SIN ESTANDARIZAR.

- Y: vector columna, en el que cada fila indica la codificación utilizada para la clase del objeto caracterizado por la descripción matemática dada por esa fila de X.

#### % Valores de codificación de las clases en Y:

- % **Arandelas:** valor 1
- % **Tuercas:** valor 2
- % **Alcayatas Lisas:** valor 3
- % **Alcayatas Roscadas:** valor 4
- % **Tornillos Cabeza Hexagonal:** valor 5
- % **Tornillos Roscados:** valor 6

- **Imágenes de test:** *Test01.bmp* y *Test02.bmp*, imágenes para la evaluación visual de la herramienta de clasificación diseñada.



Universidad  
de Huelva

## SISTEMAS DE PERCEPCIÓN

Grado en Ingeniería Informática – itinerario Computación

**Dpto. de Ingeniería Electrónica , Sistemas Informáticos y Automática**

**CURSO 2019/20 – CONVOCATORIA DE JUNIO**

**EVALUACIÓN PRÁCTICA – SEGUNDO BLOQUE ASIGNATURA**

### PROCEDIMIENTO DE DISEÑO Y APLICACIÓN – ENTREGA DE CÓDIGO MATLAB

DEBE SER ESTRUCTURADO Y ORGANIZADO EN LAS SIGUIENTES CARPETAS:

**1. Carpeta 1: estandarización de datos de entrenamiento (0.5 puntos).**

Se debe entregar el código matlab que genere los datos de entrenamiento estandarizados (media 0 y desviación típica 1) de los descriptores considerados.

**2. Carpeta 2: Diseño Clasificador Mínima Distancia para reconocer entre arandelas y tuercas (2 puntos)**

Se debe entregar código Matlab (organizado en uno o varios scripts o carpetas) que implemente las siguientes acciones:

- Selección de los tres descriptores que proporcionan la mayor separabilidad conjunta (1 punto)
- Diseño del clasificador de mínima distancia solicitado (0.5 puntos)
- Representación en el espacio de características considerado de los datos de entrenamiento de las clases junto con el plano de separación definido por el clasificador utilizado. La gráfica generada debe especificar el nombre de los ejes y las leyendas correspondientes a la información mostrada (0.5 puntos).

**3. Carpeta 3: Diseño Clasificador KNN para reconocer entre alcayatas lisas, alcayatas tipo rosca y tornillos (1 punto)**

Se debe entregar el código que implemente las siguientes acciones:

- Generar la información necesaria para la aplicación de este clasificador (0.5 puntos).
- Representación en el espacio de características considerado de los datos de entrenamiento de las clases. La gráfica generada debe especificar el nombre de los ejes y las leyendas correspondientes a la información mostrada (0.5 puntos).



Universidad  
de Huelva

## SISTEMAS DE PERCEPCIÓN

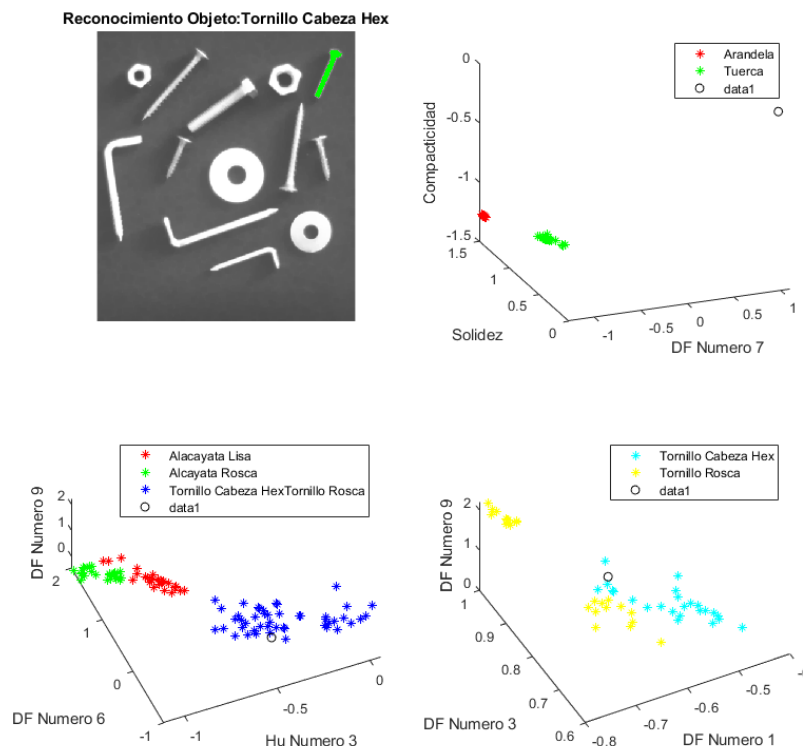
Grado en Ingeniería Informática – itinerario Computación

**Dpto. de Ingeniería Electrónica, Sistemas Informáticos y Automática**

**CURSO 2019/20 – CONVOCATORIA DE JUNIO**

**EVALUACIÓN PRÁCTICA – SEGUNDO BLOQUE ASIGNATURA**

4. **Carpeta 4: Diseño Clasificador KNN para reconocer entre tornillos de cabeza hexagonal y tornillos tipo rosca. (1 punto)** Se debe entregar el código que implemente las siguientes acciones:
- Generar la información necesaria para la aplicación de este clasificador (0.5 puntos)
  - Representación en el espacio de características considerado de los datos de entrenamiento de las clases. La gráfica generada debe especificar el nombre de los ejes y las leyendas correspondientes a la información mostrada (0.5 puntos).
5. **Carpeta 5: Algoritmo de reconocimiento (3.5 puntos – ver desglose a continuación).** Se debe entregar una función o script que reciba como entrada la ruta o el nombre de la imagen de test que debe analizar. Al ejecutarse, debe abrir una ventana tipo figure por cada pieza que haya en la imagen, con 4 gráficas:
- 1 gráfica donde se visualice, sobre la imagen original de entrada, el resultado de la segmentación del objeto en cuestión y cuyo título muestre el resultado del reconocimiento del objeto.
  - 3 gráficas específicas para cada estrategia de clasificación implementada. En cada una de ellas, se debe representar el conjunto de entrenamiento que se ha utilizado en el diseño de cada clasificador junto con la descripción matemática de la pieza en cuestión cuya clase se quiere predecir.





Universidad  
de Huelva

---

## SISTEMAS DE PERCEPCIÓN

Grado en Ingeniería Informática – itinerario Computación

**Dpto. de Ingeniería Electrónica, Sistemas Informáticos y Automática**

**CURSO 2019/20 – CONVOCATORIA DE JUNIO**

**EVALUACIÓN PRÁCTICA – SEGUNDO BLOQUE ASIGNATURA**

---

**Puntuación Algoritmo de reconocimiento – 3.5 puntos**, desglosados de la siguiente forma:

- Generación de datos (1 punto)

En esta etapa se debe utilizar binarización global con el umbral que genera el método de Otsu (se debe utilizar la función de Matlab correspondiente)

Se deben evitar analizar posibles pequeños objetos ruidosos en la segmentación de objetos y considerar únicamente agrupaciones conectadas de píxeles que tengan un número mínimo de píxeles (establecido de acuerdo a algún criterio).

- Aplicación individual de los clasificadores diseñados (1.5 puntos)
- Implementación de estrategia de clasificación global (0.5 puntos)
- Visualización de resultados (0.5 puntos)

### **OBSERVACIÓN GENERAL:**

UN ERROR DE CÓDIGO O PROVOCADO POR NO ENCONTRAR LA INFORMACIÓN REQUERIDA IMPLICARÁ PUNTUACIÓN CERO EN LA PARTE CORRESPONDIENTE