



Universidad
de Huelva

TRABAJO DE VISIÓN POR COMPUTADOR

Grado en Ingeniería Informática

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática

SISTEMA DE SEGMENTACIÓN Y RECONOCIMIENTO DE CARACTERES DE PLACAS DE MATRÍCULAS

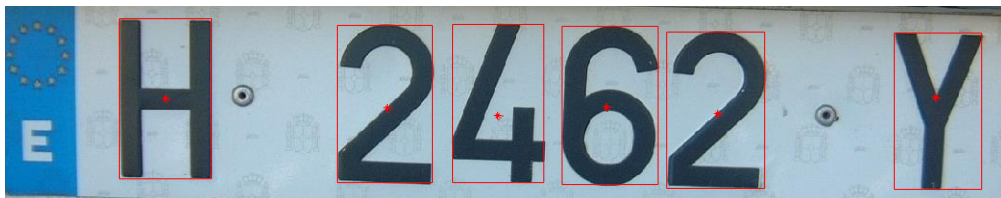
OBJETIVO:

- ❖ Implementar un algoritmo que segmente y reconozca cada uno de los números y letras que componen las imágenes de placas de matrícula facilitadas, a partir de la siguiente función:

`Funcion_Reconoce_Matricula (RutaImagen)`

- La función debe recibir como entradas la variable `RutaImagen`, cadena de texto con la ruta del archivo imagen a tratar.
- La función debe visualizar en una ventana tipo `figure` la imagen original de entrada junto con el centroide de cada uno de los caracteres que ha segmentado, así como el contorno del rectángulo más pequeño que los contiene. Además, el título de esta ventana debe estar formado por los caracteres reconocidos de la matrícula.

H2462Y





Universidad
de Huelva

TRABAJO DE VISIÓN POR COMPUTADOR

Grado en Ingeniería Informática

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática

MATERIAL DISPONIBLE:

- Plantillas: disponibles en `\Material_Imagenes_Plantillas\00_Plantillas`
- Imágenes *Training*: disponibles en `\Material_Imagenes_Plantillas\01_Training`
- Imágenes *Test*: disponibles en `\Material_Imagenes_Plantillas\02_Test`

ALGUNAS FUNCIONES MATLAB Y CONSIDERACIONES DE UTILIDAD:

- `bwlabel`, `double`, `eval`, `find`, `fspecial`, `graythresh`, `hold`, `imfilter`, `imread`, `imresize`, `imshow`, `length`, `line`, `mat2gray`, `max`, `mean`, `min`, `num2str`, `ones`, `ordfilt2`, `plot`, `size`, `sort`, `stdfilt`, `sum`, `title`, `unique`, `zeros`
- Ejemplo de aplicación de `num2str`:

```
Numero = 1 ; NumeroStr = num2str (Numero, '%02d');
```
- `stdfilt`: calcula, para cada punto de una matriz, valores de desviaciones estándar sobre ventanas locales centradas en los puntos bajo consideración. Para tratar los puntos del contorno, esta función “extiende” la matriz realizando un espejo de sus valores reales.
- `imfilter`, `ordfilt2`: utiliza la opción ‘`symmetric`’ para tratar los píxeles del contorno de la misma forma que lo hace `stdfilt`.
- `eval`: para evaluar el contenido de una cadena de texto.
- `unique`: para obtener los valores (sin repeticiones) de un vector.



Universidad
de Huelva

TRABAJO DE VISIÓN POR COMPUTADOR

Grado en Ingeniería Informática

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática

PRIMERA FASE DE DESARROLLO: Segmentación de Caracteres.

- ❖ **Objetivo:** segmentar en la imagen original los píxeles que integran cada carácter de la matrícula y obtener la región compuesta por el rectángulo más pequeño que delimita dichos píxeles ('*bounding box*'). Las segmentaciones o regiones de píxeles de cada carácter siempre se expresarán a través de matrices binarias cuyos valores serán 0's o 1's.



- ❖ **Diseño:** el algoritmo de segmentación debe ser desarrollado teniendo en cuenta exclusivamente las imágenes disponibles en `\Material_Imagenes_Plantillas\01_Training:`





Universidad
de Huelva

TRABAJO DE VISIÓN POR COMPUTADOR

Grado en Ingeniería Informática

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática

Estas imágenes recogen toda la casuística exigida y contemplada en el problema:

- Diferentes condiciones de iluminación en las imágenes.
- Defectos de iluminación en una imagen.
- Ruido o suciedad presente que podría situarse cerca de los caracteres y que podría condicionar el proceso de segmentación si no se trata. Debe asumirse que siempre se manifiesta en pequeñas agrupaciones de píxeles.
- Caracteres de interés situados en posiciones muy cercanas. Este hecho puede condicionar el proceso de segmentación si no se trata.
- Caracteres de interés situados en casi en el borde de la imagen.
- Pequeñas agrupaciones de píxeles de carácter con valores anómalos.

Por tanto, la estrategia y metodología de segmentación debe contemplar y dar respuesta a estas diferentes condiciones que podrían presentar las imágenes. Además, los parámetros representativos de los diferentes algoritmos utilizados deben ser ajustados con un criterio justificado utilizando exclusivamente las imágenes de entrenamiento.

❖ La metodología de segmentación debe basarse en los siguientes principios de aplicación:

- En la componente roja de las imágenes, los píxeles del logo de la Unión Europea presentan valores bajos similares a los de los píxeles que integran los caracteres de la matrícula.
- Los caracteres de la matrícula están centrados verticalmente en la imagen.
- Con excepción del objeto correspondiente al logo de la Unión Europea, los posibles objetos que no son de interés que pudieran también estar centrados verticalmente en la imagen, presentan un área inferior al de los caracteres de la matrícula.
- El objeto del logo de la Unión Europea siempre se localiza en el borde izquierdo de la imagen.



Universidad
de Huelva

TRABAJO DE VISIÓN POR COMPUTADOR

Grado en Ingeniería Informática

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática

SEGUNDA FASE DE DESARROLLO: Reconocimiento de Caracteres.

- ❖ **Objetivo:** reconocer los caracteres que integran las matrículas proporcionadas en las imágenes de *training* y de *test* disponibles en `\Material_Imagenes_Plantillas\01_Training` y `\Material_Imagenes_Plantillas\02_Test`.

El problema está restringido al reconocimiento de los siguientes caracteres, que son todos los que aparecen en las placas de matrícula facilitadas:

Caracteres = '0123456789ABCD EFGHKLNRSTXYZ'

- ❖ **Metodología de reconocimiento:** ajuste de plantillas mediante correlación bidimensional normalizada (*template matching*). Este procedimiento evalúa la similitud del carácter desconocido con plantillas generadas a priori de caracteres conocidos. El reconocimiento del carácter se decide según la plantilla para la que se alcanza la máxima similitud.
- ❖ **Material disponible:**

En el archivo `plantillas.mat` (disponible en `\Material_Imagenes_Plantillas\00_Plantillas`) se facilitan, para cada uno de los caracteres posibles, 7 plantillas correspondientes a distintas rotaciones del carácter. De esta forma, se pretenden considerar las distintas orientaciones que pueden tener los caracteres a reconocer.

Estas plantillas se proporcionan en matrices binarias, nombradas de forma que su nombre especifica el carácter al que pertenece (según la posición que ocupa en el vector `Caracteres` anterior) y la orientación con la que fue generada (de 1 a 7, correspondientes a ángulos -9, -6, -3, 0, 3, 6 y 9°).



Universidad
de Huelva

TRABAJO DE VISIÓN POR COMPUTADOR

Grado en Ingeniería Informática

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática

Ejemplos de nombres y contenido de las matrices de las plantillas de los caracteres 1 y A, que ocupan la segunda y undécima posición, respectivamente, del vector *Caracteres*:

Nombres de matrices de caracteres *I* y *A*:

Objeto02Angulo01, Objeto02Angulo02, Objeto02Angulo03, Objeto02Angulo04,
Objeto02Angulo05, Objeto02Angulo06, Objeto02Angulo07

Objeto11Angulo01, Objeto11Angulo02, Objeto11Angulo03, Objeto11Angulo04,
Objeto11Angulo05, Objeto11Angulo06, Objeto11Angulo07

Contenido visual de matrices de caracteres *I* y *A*:



Observación: todas las plantillas facilitadas han sido generadas a partir de las siguientes muestras:

A C F H L R T Y 0 2 4 6 8

B D G K N S X Z 1 3 5 7 9



Universidad
de Huelva

TRABAJO DE VISIÓN POR COMPUTADOR

Grado en Ingeniería Informática

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática

❖ Procedimiento:

- Una vez segmentado el carácter de una matrícula y obtenido su *bounding box*, hay que cuantificar el grado de similitud de este con cada una de las plantillas facilitadas.

Para ello, debe utilizarse la siguiente función que calcula la correlación normalizada entre dos matrices bidimensionales de igual dimensión:

`ValorCorrelacion = Funcion_CorrelacionEntreMatrices (Matriz1, Matriz2)`

$ValorCorrelacion =$

$$= \frac{\sum_x \sum_y \{ [Matriz1(x,y) - mean(Matriz1)] [Matriz2(x,y) - mean(Matriz2)] \}}{\sqrt{\{ [\sum_x \sum_y [Matriz1(x,y) - mean(Matriz1)]^2 \} [\sum_x \sum_y [Matriz2(x,y) - mean(Matriz2)]^2 \}}}$$

- El algoritmo decidirá que el carácter del objeto desconocido es aquel al que corresponde la plantilla para la que se alcanza la correlación máxima.

Criterios procedimentales a seguir:

- Antes de calcular la correlación entre los valores del *bounding box* de un carácter y una plantilla determinada, es necesario adaptar las dimensiones del *bounding box* del carácter al de la plantilla considerada. Utiliza para ello la función de matlab `imresize`.
- Por cada carácter analizado, debe generarse una matriz de datos con los valores de correlación entre el carácter de interés y cada una de las plantillas disponibles.