

# CHECKPOINT PRÁCTICA 2

## DESCRIPTORES UTILIZADOS

A fecha del checkpoint los descriptores implementados hasta ahora han sido los siguientes:

- **Número de agujeros.** Cabe esperar que con el número de agujeros en los números se pueda hacer una sencilla distinción entre 0, 8, 9, 4 y 6 y el resto de números.
- **Área total de los agujeros.** El tamaño de los agujeros de un número varía según el número y el agujero que sea.
- **Área total del número (partes blancas tras la binarización).** Puesto que el tamaño de los números está normalizado es de esperar que el número de píxeles (blancos) para un 1 no sea el mismo que el de un 9.
- **Número de corners.** El número de corners en los números difiere, por ejemplo, en un 1 perfecto cabría esperar detectar dos corners con Harris, mientras que en un 7 serían tres.

Estos descriptores que no han dado grandes resultados, por lo que se ha optado por seguir una estrategia diferente. En esta, se han analizado las concavidades y/o agujeros de cada número y se han considerado las siguientes características para cada uno:

- **Posición X e Y del centroide:** Describe la posición aproximada en la que se encuentra la concavidad o el agujero.
- **Área de las concavidades:** Describe el tamaño total de la concavidad o el agujero.
- **Tamaño del axis mayor y tamaño menor:** Describe cual es la forma tendencial de la concavidad o el agujero, que puede aproximarse más a una elipse más vertical o una más horizontal.

Se ha optado por considerar siempre 3 concavidades (faltaría experimentar variando este número) y 2 agujeros (debido a que el máximo número de agujeros en un número es 2, en el 8).

Así pues, se dan un total de  $3 \times 5$  variables (concavidades) +  $2 \times 5$  variables (agujeros) = 25 variables. Para que estas variables se correspondan entre distintas imágenes de números, dados los datos sobre las concavidades y/o agujeros de una imagen, estos se ordenan según el tamaño del área.

En el caso en que la imagen que se observe presente menos concavidades y/o agujeros que los que se están considerando, se rellenan las variable con valores a 0. Además, se eliminan aquellas concavidades y/o agujeros que no llegan a un mínimo de tamaño, por errores en la extracción.

Para no confundir las concavidades con agujeros, se extraen primero los datos sobre los agujeros, y para calcular los de las concavidades, se realiza sobre un relleno de los agujeros.

## CLASIFICADOR QUE SE UTILIZARÁ

Los clasificadores que se utilizarán son aquellos que han resultado ser más prometedores hasta ahora. En base a la performance en training., estos son SVM y kNN. Faltaría determinar cuál de ambos da mejores resultados una vez estén todos los descriptores y compararlos en test.

## RESULTADOS PRELIMINARES

Actualmente, teniendo en cuenta los descriptores obtenidos por las características sobre los agujeros y concavidades, se han obtenido los siguientes resultados, en training, con clasificadores basados en SVM y kNN.

1.8 ☆ SVM	Accuracy: 77.2%
Last change: Linear SVM	25/25 features
1.9 ☆ SVM	Accuracy: <b>84.0%</b>
Last change: Quadratic SVM	25/25 features
1.10 ☆ SVM	Accuracy: 83.9%
Last change: Cubic SVM	25/25 features
1.11 ☆ SVM	Accuracy: 76.5%
Last change: Fine Gaussian SVM	25/25 features
1.12 ☆ SVM	Accuracy: 82.0%
Last change: Medium Gaussian SVM	25/25 features
1.13 ☆ SVM	Accuracy: 68.3%
Last change: Coarse Gaussian SVM	25/25 features
1.14 ☆ KNN	Accuracy: 82.5%
Last change: Fine KNN	25/25 features
1.15 ☆ KNN	Accuracy: 77.9%
Last change: Medium KNN	25/25 features
1.16 ☆ KNN	Accuracy: 58.4%
Last change: Coarse KNN	25/25 features
1.17 ☆ KNN	Accuracy: 79.4%
Last change: Cosine KNN	25/25 features
1.18 ☆ KNN	Accuracy: 76.6%
Last change: Cubic KNN	25/25 features
1.19 ☆ KNN	Accuracy: 81.2%
Last change: Weighted KNN	25/25 features