
TEMA 4. MORFOLOGÍA BINARIA

Librerías que tenemos que importar:

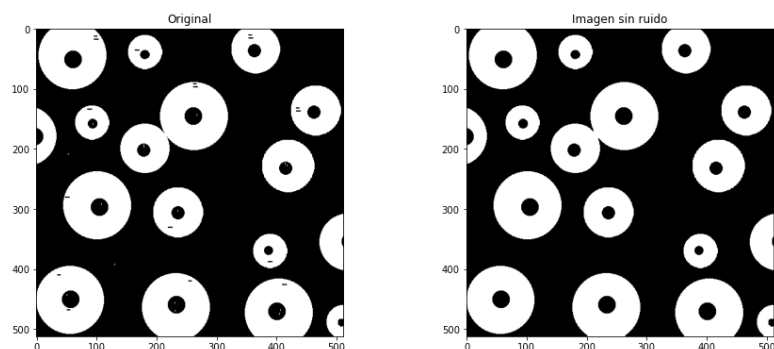
- **opencv:** Open Computer Vision (Visión Artificial Abierta), librería para visión artificial de código abierto disponible para Python.
 - **numpy:** librería de Python especializada en el cálculo numérico y el análisis de datos, especialmente para un gran volumen de datos; da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas.
 - **matplotlib.pyplot:** mostrar gráficos matemáticos e imágenes (como gráficos).
-

Ejercicio 0: Operaciones básicas

Practica, con las funciones proporcionadas por OpenCV, a realizar las operaciones de dilatación, erosión, apertura y cierre. Comprueba las relaciones que existen entre ellas. ¿Están implementadas para imágenes binarias o en escala de grises?

Ejercicio 1: Eliminación de ruido

- a) Dada la imagen **circulos_ruido.png**, se pretende eliminar el ruido usando operaciones morfológicas binarias. **Razona** cuál será la mejor secuencia de operaciones para ello con los mejores elementos estructurales (que podrían ser distintos para eliminar un ruido y otro). Queremos que los elementos estructurales sean lo más pequeños posibles para obtener el menor error posible. Calcula el Error Cuadrático Medio comparando la imagen procesada tras eliminar el ruido con la imagen original para corroborar la mejor elección.



- b) Sobre la imagen sin ruido obtenida, determinar la mejor operación morfológica para eliminar los pequeños círculos negros dentro de los blancos cometiendo un error lo más pequeño posible.

Ejercicio 2: Extracción de fronteras

Dada una imagen binaria I , los píxeles de la frontera de I se obtienen restando a I el resultado de dilatar la imagen con un elemento estructural blanco 3×3 . Define una función llamada ***extraer_fronteras*** que realice el proceso anterior. Aplícala a la imagen **pez.png**.

