

Procesado de imagen y visión por computador

Curso 2018/2019

Operadores morfológicos

0. Introducción

Los operadores morfológicos¹ son herramientas que actúan sobre la forma de los objetos en una imagen, bien para extraer determinadas componentes de estos, como el contorno o el esqueleto, o bien para modificar su forma. Aunque pueden ser definidos sobre cualquier tipo de imagen, en general, su principal aplicación es sobre matrices binarias, es decir, imágenes cuyos píxeles toman sólo dos valores, por ejemplo, las matrices obtenidas a la salida de los algoritmos de segmentación.

A partir de ahora, y dado que en nuestro caso, las matrices binarias son empleadas para diferenciar entre los píxeles que corresponden a objetos y los que corresponden al fondo, las matrices binarias serán implementadas como imágenes de OpenCV que toman los valores 0 para el fondo, y 255 para el objeto.

1. Erosión y dilatación

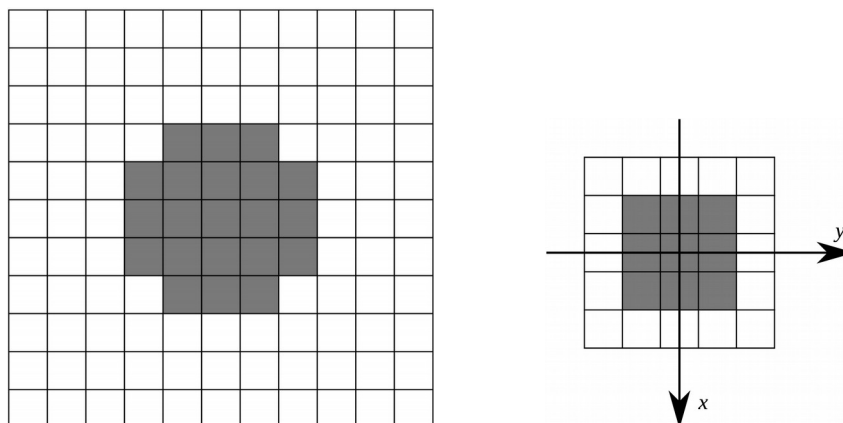
La erosión y dilatación son la base de la mayoría de las operaciones morfológicas. La erosión se define matemáticamente como:

$$C = B \ominus M$$

donde M se denomina como *máscara de convolución*, B es la matriz binaria original, y C la matriz binaria erosionada. El cálculo del valor de los píxeles de la matriz erosionada se realiza de acuerdo a:

$$c_{xy} = \sum_{i,j \in M} b_{x+i,y+j}$$

donde el operador \sum corresponde a la operación AND binaria. De forma práctica, el proceso de obtención de la matriz erosionada sería el siguiente. Supongamos que la matriz original es la que se muestra en la figura de la izquierda, donde los píxeles de *fondo* se representan en blanco, y los de *objeto* en gris oscuro, y la máscara de convolución es la representada a la derecha.



El proceso consistiría en situar, sobre el píxel (x,y) de la matriz original, la máscara de convolución, y si todos los píxeles que caen dentro de la máscara son *objeto*, el píxel en cuestión en la matriz resultante también será *objeto*. En el caso de que al menos uno de los píxeles no corresponda a

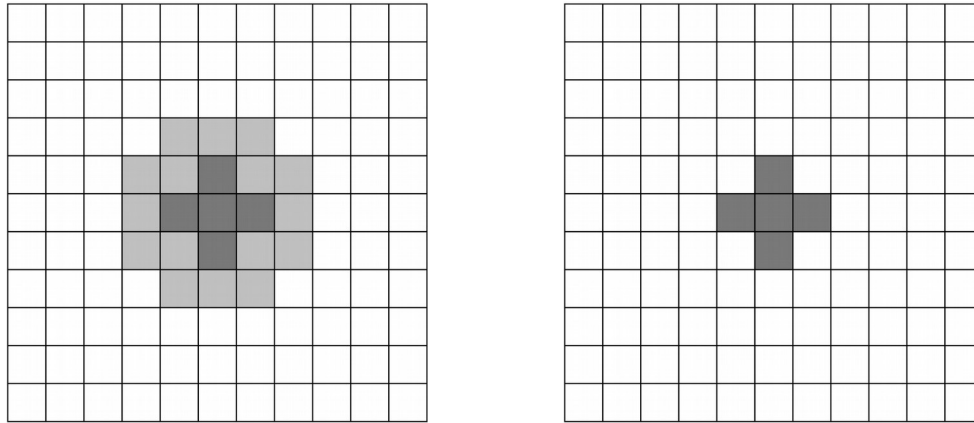
¹ http://docs.opencv.org/master/d9/d61/tutorial_py_morphological_ops.html

Procesado de imagen y visión por computador

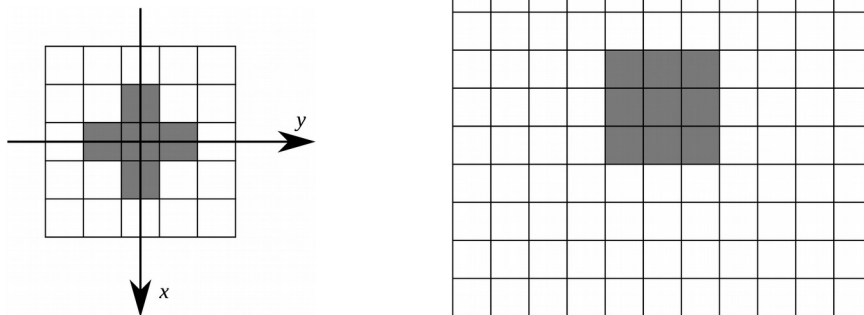
Curso 2018/2019

Operadores morfológicos

objeto, el píxel en cuestión toma el valor *fondo*. En la siguiente figura, a la izquierda, se muestra el proceso de erosión sobre la matriz original. En gris claro se muestran los píxeles que en la matriz original corresponden a *objeto* y que en la matriz erosionada pasan a valer *fondo*, mientras que en gris oscuro se muestran los que continúan valiendo *objeto* después del proceso. En la figura de la derecha se muestra el resultado final.



La máscara de convolución puede tener cualquier forma. Por ejemplo, si tomamos como máscara de convolución la representada a la izquierda, el resultado sería la imagen de la derecha:



El proceso contrario a la erosión se denomina dilatación. La definición matemática de la dilatación sería:

$$C = B \oplus M$$

y el cálculo del valor de los píxeles de la matriz dilatada se realiza de acuerdo a:

$$c_{xy} = \prod_{i,j \in M} b_{x+i,y+j}$$

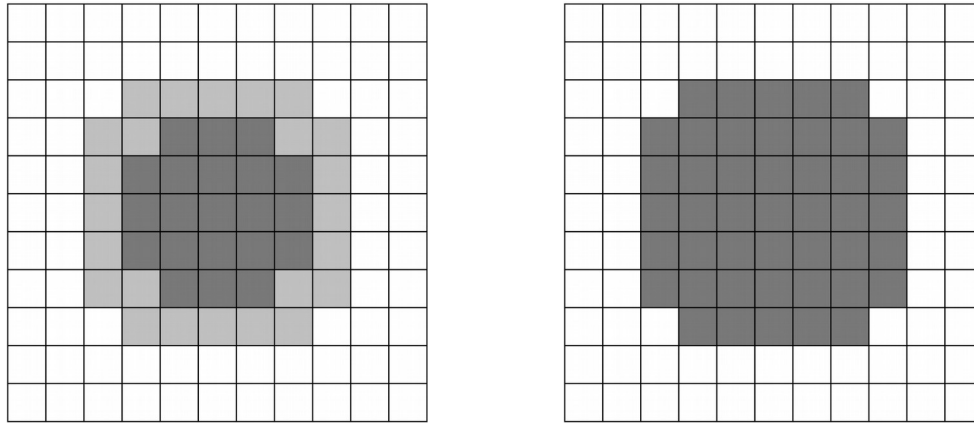
donde el operador \prod corresponde a la operación OR binaria. En este caso el proceso de obtención de la matriz dilatada sería al contrario. Es decir, un píxel de la matriz dilatada tomará el valor *objeto* si cualquiera de los píxeles que caen dentro de la máscara de convolución tiene el valor *objeto* en la imagen original. Si suponemos la matriz binaria del ejemplo anterior, la dilatación de

Procesado de imagen y visión por computador

Curso 2018/2019

Operadores morfológicos

dicha matriz con la máscara cuadrada de 3×3 píxeles daría lugar a la matriz que se representa a continuación.



2. Erosión de una matriz binaria con OpenCV.

OpenCV implementa la operación de erosión con la función:

```
cv2.erode(img, kernel, ancla, iteraciones);
```

donde *kernel* es la máscara de convolución. El proceso de erosión puede realizarse de forma iterativa y la variable *iteraciones* indica cuántas veces se repite dicho proceso. Por defecto, se realiza una única vez. La variable *ancla* especifica el origen de coordenadas de la máscara de convolución. Si se omite toma el valor $(-1, -1)$ coincidiendo con el elemento central de la máscara.

Implementar con OpenCV un programa que realice la erosión de una matriz binaria. Para ello use la imagen *Morfologico.png*. Utilice el código implementado en el archivo *Apartado2.py*, y añada en el lugar correspondiente la función de erosión. Realice el proceso de erosión 1, 2, 3 y 4 veces, y guarde las imágenes resultantes como *erosion1.png*, *erosion2.png*, *erosion3.png* y *erosion4.png*.

3. Dilatación de una matriz binaria con OpenCV.

En este caso, OpenCV implementa la operación de dilatación en la función *dilate*, con los mismos parámetros que la primera. Realizar el mismo programa que en el apartado anterior, cambiando la erosión por la dilatación, sobre la misma imagen. Guardar las imágenes con los nombres *dilatacion1.png*,...

4. Apertura y cierre.

La principal utilidad de los operadores anteriores es que permiten realizar un suavizado de los contornos de los objetos, eliminando principalmente los píxeles ruidosos que quedan próximos a los contornos y los aislados no pertenecientes a un objeto definido. Sin embargo, como se ha visto en los ejemplos anteriores, la dilatación expande el tamaño de los objetos, mientras que la erosión la contrae. Si es necesario conservar el tamaño de los objetos una vez realizado el proceso anterior, una buena estrategia puede ser el realizar los dos procesos anteriores de manera consecutiva. Así, si sobre una matriz binaria realizamos en primer lugar una erosión, y posteriormente una dilatación, al proceso se le denomina *apertura*. Si por el contrario, en primer lugar se realiza la dilatación, y seguidamente una erosión, entonces se denomina como *cierre*. Estas operaciones pueden realizarse

Procesado de imagen y visión por computador

Curso 2018/2019

Operadores morfológicos

en OpenCV con las mismas funciones anteriores, sin más que situar las funciones en el orden correcto. OpenCV también incluye la función *morphologyEx*, que permite realizar la apertura y cierre, aparte de otras funciones morfológicas, con una única llamada.

En este apartado, el código a desarrollar es el siguiente. La imagen *ellipse2.png* es una versión ruidosa de la imagen *ellipse.png*. El código implementado en el archivo *Apartado4.py* abre ambas imágenes y las umbraliza con un umbral arbitrario. Posteriormente las visualiza en la pantalla, y por último, compara ambas imágenes, sacando por pantalla el número de píxeles que no coinciden (píxeles marcados como objeto en la imagen ruidosa que en la imagen original son fondo, y viceversa). El objetivo de este apartado es añadir, en el punto marcado dentro del código, las operaciones morfológicas más convenientes para conseguir reducir el error anterior al máximo posible. Puede probar con las funciones de erosión, dilatación, apertura, cierre, a cambiar el kernel... Así mismo, también puede probar a variar el umbral de segmentación de la imagen ruidosa.