UNIVERSIDAD
DE GRANADA

Visión por Computador
grado en ingeniería informática

Cuestionario 1

Filtrado y Detección de regiones

Autor
José María Sánchez Guerrero

Rama
Computación y Sistemas Inteligentes

Escuela Técnica Superior de Ingenierías Informática y de
Telecomunicación

Curso 2019-2020

Índice

Ejercicio 1

Diga en una sola frase cuál cree que es el objetivo principal de la Visión por Computador. Diga también cuál es la principal propiedad de las imágenes de cara a la creación algoritmos que la procesen.

El objetivo principal de la Visión por Computador es obtener información significativa de imágenes digitales, posteriormente analizarla, tratarla y comprender su contenido para tomar decisiones sobre ella de la forma más similar posible a la humana.

Pese a la respresentación en forma matricial de las imágenes que facilita los cálculos a los algoritmos, no podemos centrarnos únicamente en una posición (píxel) de ésta, ya que los valores alrededor suyo también contienen información relevante sobre él.

Ejercicio 2

Expresar las diferencias y semejanzas entre las operaciones de correlación y convolución. Dar una interpretación de cada una de ellas que en el contexto de uso en visión por computador.

Ambas son operaciones que transforman localmente una imagen calculado nuevos valores para cada uno de los píxeles. Esto lo hacen utilizando una máscara 2D de tamaño NxN, siendo N un número impar, y aplicándola de la siguiente forma:

Para la correlación:

$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i+u, j+v] \tag{1}$$

Para la convolución:

$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i-u, j-v] \tag{2}$$

Como podemos ver, correlación y convolución son prácticamente iguales, excepto que en la convolución volteamos el filtro antes de correlacionar. Por ejemplo, convolucionar una imagen 1D con un filtro $(1,3,5)$ sería lo mismo que correlacionarla con el filtro $(5,3,1)$. En caso de que fuese una convolución 2D voltearíamos tanto horizontal como verticalmente.

Otra cosa que tienen en común es que, como ambos son filtros lineales, ambos cumplen las siguientes propiedades: la **superposición**, la cual dice que es lo mismo aplicar una máscara a una composición de imágenes $f_1 + f_2$, que aplicarsela a f_1 y a f_2 por separado: $h * (f_1 + f_2) = (h * f_1) + (h * f_2)$; y son ***Shift-Invariant System***, es decir, sistemas cuyo valor de entrada no cambian los valores de salida, por lo que no dependen de ellos.

La diferencias más importantes entre ellos es que la convolución es **conmutativa**, **distributiva en la adición** y, la más relevante, **asociativa**. Es decir, siendo f y g dos filtros distintos, entonces $(f * g) * h = f * (g * h)$. Esto es muy útil por ejemplo para calcular el filtro *Difference of Gaussian (DoG)*, en el que no tendríamos que convolucionar la imagen con un filtro Gaussiano y posteriormente con uno derivado, simplemente convolucionaríamos el filtro Gaussiano con el derivado y ya se lo aplicamos a la imagen.

Ejercicio 3

¿Cuál es la diferencia "esencial" entre el filtro de convolución y el de mediana? Justificar la respuesta.

La principal diferencia entre estos filtros es que el filtro de convolución, como hemos visto antes, es lineal, mientras que el filtro de mediana no lo es. Para justificar la respuesta veamos un ejemplo: si tenemos los filtros $A = (0, 1, 2, 3, 4)$, $B = (5, 0, 1, 4, 5)$ y $(A + B) = (5, 1, 3, 7, 9)$, el cálculo de sus medianas es:

$$Med(A) = 2 \qquad Med(B) = 4 \tag{3}$$

Y por tanto, como:

$$Med(A) + Med(B) = 6 \quad \neq \quad Med(A + B) = 5 \tag{4}$$

podemos justificar que los filtros de mediana no son lineales.

Ejercicio 4

Identifique el "mecanismo concreto" que usa un filtro de máscara para transformar una imagen.

El mecanismo que utilizan todos los filtros de máscara para transformar una imagen es que lo hace utilizando **información local**, es decir, que el valor de cada píxel se calcula teniendo en cuenta los píxeles cercanos a él (los que están dentro de la máscara).

Podremos encontrar varios tipos de filtros, ya sean lineales o no lineales como hemos visto antes, pero ninguno realizará cálculos en la imagen sin tener en cuenta sus píxeles asociados en la máscara.

Ejercicio 5

¿De qué depende que una máscara de convolución pueda ser implementada por convoluciones 1D? Justificar la respuesta.

Depende de que la máscara de convolución solo se pueda descomponer como producto de dos matrices de dimensión 1. Por ejemplo, la siguiente máscara Gaussiana se puede descomponer en:

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} \tag{5}$$

Para comprobar si esto es posible, podemos realizar la **descomposición en valores singulares** de la matriz. Si esta descomposición nos da un valor distinto de 0, la máscara será separable en dos matrices 1D, y por tanto, la podremos represetar como:

$$\sum_{r}^{i=0} \sigma_i u_i v_i^T \tag{6}$$

siendo σ_i los valores de la matriz diagonal, y u_i y v_i los valores de las respectivas matrices 1D resultantes.

Otra forma que tenemos de comprobarlo es que, si nos fijamos en este tipo de máscaras, el **rango** de la matriz siempre va a ser igual a 1.

Ejercicio 6

Identificar las diferencias y consecuencias desde el punto de vista teórico y de la implementación entre:

(a) **Primero alisar la imagen y después calcular las derivadas sobre la imagen alisada**

(b) **Primero calcular las imágenes derivadas y después alisar dichas imágenes.**

Justificar los argumentos.

Desde un punto de vista teórico dependerá del tipo de filtro que usemos, pero lo lógico sería elegir un filtro de convolución. Como hemos visto en el ejercicio 2, este filtro tiene cumple la propiedad **asociativa**, por lo que tanto la opción *a* como la *b* darían el mismo resultado.

$$(f * g) * h = f * (g * h) \tag{7}$$

En cuanto a la implementación, es más conveniente **alisar primero la imagen y posteriormente calcular las derivadas** sobre la imagen alisada. Si lo hacemos de esta forma, tendremos que hacer solo una convolución y dos derivadas después. De la otra forma, haremos dos derivadas al principio y posteriormente dos alisamientos, uno para la X y otro para la Y.

Ejercicio 7

Ejercicio 8

Ejercicio 9

Ejercicio 10

Ejercicio 11

¿Bajo qué condiciones podemos garantizar una perfecta reconstrucción de una imagen a partir de su pirámide Laplaciana? Dar argumentos y discutir las opciones que considere necesario.

Únicamente con la pirámide Laplaciana no podríamos reconstruir la imagen original, también necesitaríamos la última submuestra de la imagen obtenida al hacer la pirámide Gaussiana.

Si disponemos de ella, se podría reconstruir con el siguiente algoritmo, el cual consiste en sumar al último nivel de la Laplaciana esta última submuestra de la Gaussiana ampliada con una función F:

$$G_k = L_k + F(G_{k+1}) \tag{8}$$

Cuando terminemos (llegemos a G_1) será cuando tengamos la imagen original completamente reconstruida.

Ejercicio 12

¿Cuáles son las contribuciones más relevantes del algoritmo de Canny al cálculo de los contornos sobre una imagen? ¿Existe alguna conexión entre las máscaras de Sobel y el algoritmo de Canny? Justificar la respuesta

Las contribuciones que tiene este algoritmo frente a otros es que este utiliza las siguientes técnicas:

1. **Non-maximum supression.** Esta técnica recorre todos los píxeles de cada imagen y, para cada uno de ellos, comprueba que sus píxeles fronterizos no tienen un valor más alto que el central. En caso de que exista, lo pondrá este

a cero.

$$\begin{bmatrix} 3 & 2 & 3 \\ 6 & 4 & 1 \\ 1 & 3 & 3 \end{bmatrix} \underset{(x,y)>=(0,0)}{\rightarrow} \begin{bmatrix} 0 & 0 & 0 \\ 6 & 0 & 1 \\ 1 & 3 & 3 \end{bmatrix} \qquad \begin{bmatrix} 3 & 2 & 3 \\ 1 & 4 & 1 \\ 1 & 3 & 3 \end{bmatrix} \underset{(x,y)<(0,0)}{\rightarrow} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

2. dskjfjegnjdgbffjb

Ejercicio 13

Ejercicio 14

Ejercicio 15