

Detección de Bordes

Rodrigo Javier Herrera García *

Los detectores de bordes conforman un grupo muy importante de métodos de procesamiento preliminar de imágenes. Se utilizan para detectar cambios abruptos en la función de intensidad que describe una imagen. Entonces, los bordes son aquellos píxeles donde se presentan cambios de intensidad fuertes. Como se sabe, el cálculo permite encontrar los cambios en las funciones continuas a través de las derivadas. Puesto que las imágenes son modeladas como funciones de dos variables, los operadores que describen los bordes se expresan como derivadas parciales.

1. Operadores de Primera Derivada

Un cambio en la intensidad de la imagen puede describirse como un *gradiente* que apunta en la dirección hacia donde el crecimiento del brillo es más grande. Un borde es una propiedad ligada a un píxel particular y se calcula a partir del comportamiento de la función de intensidad de la imagen en un vecindario alrededor del píxel. Este borde es entonces un vector variable que tiene dos componentes: *magnitud* y *dirección*. La magnitud del borde es la magnitud del gradiente mientras que la dirección del borde está rotado -90° con respecto a la dirección del gradiente. En la **Figura 1** se muestra una imagen sintética, la dirección del gradiente y la dirección del borde para un punto particular.

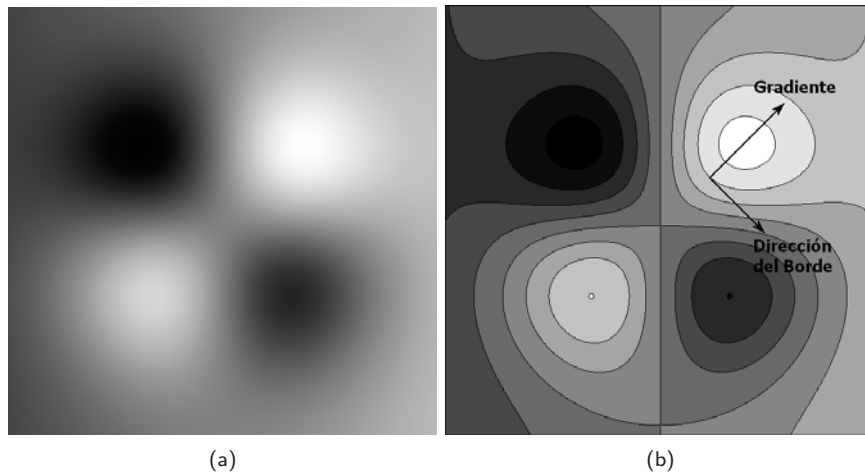


Figura 1: Magnitud y dirección del gradiente de intensidad y dirección del borde en una imagen. (a) Imagen original, (b) contornos de la imagen.

Los métodos de detección de bordes se utilizan con frecuencia en análisis de imágenes para encontrar las fronteras de las regiones. Si se tiene una región con brillo homogéneo, entonces sus fronteras estarán en los píxeles donde la intensidad de la función de imagen varíe. Si la imagen no tiene ruido, los píxeles que determinan las fronteras de la región serán aquellos que presenten una alta magnitud de borde (magnitud del gradiente). Entonces, para una función de intensidad o brillo de una imagen $f(x, y)$, el gradiente en

*Profesor del Proyecto Curricular de Ingeniería Electrónica, Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas, Bogotá, D.C., Colombia.

el punto (x, y) está definido como

$$\nabla \mathbf{f} = \frac{\partial f}{\partial x} \mathbf{i} + \frac{\partial f}{\partial y} \mathbf{j} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (1)$$

La magnitud y dirección del gradiente, en funciones de imagen continuas se calculan como

$$\nabla f = |\nabla \mathbf{f}| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (2)$$

$$\psi = \angle \nabla \mathbf{f} = \arg \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) = \arctan \left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}} \right) \quad (3)$$

donde $\arg(r, s)$ es el ángulo (en radianes) que se forma entre el eje x y la recta que une el punto (r, s) con el origen. Con frecuencia, se está más interesado en las magnitudes de los bordes que en sus direcciones. Para estos casos se utiliza el operador *Laplaciano*, definido como

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \quad (4)$$

La característica del laplaciano es que posee las mismas propiedades en todas las direcciones, por lo tanto es independiente de la rotación de la imagen. El laplaciano se estudiará posteriormente.

Una imagen digital es de naturaleza discreta lo que implica que el cálculo de las derivadas de las ecuaciones (1) a (4) debe aproximarse a ecuaciones de diferencia. La primera aproximación parte de suponer que la variación de la función f es aproximadamente lineal para un pequeño intervalo Δx o Δy de las variables independientes x o y , donde Δx y Δy corresponden al intervalo de muestreo de la función de imagen. Entonces, para una imagen digital f , las derivadas horizontal y vertical pueden aproximarse a las ecuaciones de diferencia de primer orden

$$\frac{\partial f(x, y)}{\partial x} \approx \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} \quad (5)$$

$$\frac{\partial f(x, y)}{\partial y} \approx \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} \quad (6)$$

Sin embargo, en la función de intensidad de una imagen digital, cada píxel está referenciado por índices que corresponden a números enteros. La función de la imagen está definida como $f[m, n]$ con $m = 1, 2, \dots$ y $n = 1, 2, \dots$. Luego $\Delta x = 1$ y $\Delta y = 1$, lo que nos lleva a redefinir las ecuaciones (5) y (6) como

$$\frac{\partial f}{\partial x} \approx f[m + 1, n] - f[m, n] \quad (7)$$

$$\frac{\partial f}{\partial y} \approx f[m, n + 1] - f[m, n] \quad (8)$$

Entonces, para el cálculo aproximado de la magnitud del gradiente, en una imagen digital, se reemplazan las ecuaciones (7) y (8) en la ecuación (2), obteniendo

$$\nabla f \approx \sqrt{(f[m + 1, n] - f[m, n])^2 + (f[m, n + 1] - f[m, n])^2}. \quad (9)$$

Es una práctica habitual obtener la magnitud del gradiente a través de los valores absolutos de las diferencias de intensidad en lugar de la forma mostrada en la ecuación (9). La razón es la complejidad del cálculo de cuadrados y raíces cuadradas, sobretodo en implementaciones en *hardware*. Si la intención es ubicar los píxeles que corresponden a variaciones de intensidad específicas, la aproximación es válida porque importan más las diferencias que se presenten que los valores en sí mismos. Entonces, la magnitud del gradiente estaría dada por

$$\nabla f \approx \left| \frac{\partial f}{\partial x} \right| + \left| \frac{\partial f}{\partial y} \right| \quad (10)$$

Teniendo en cuenta esto último, la magnitud aproximada del gradiente puede obtenerse como

$$\nabla f \approx |f[m+1, n] - f[m, n]| + |f[m, n+1] - f[m, n]| \quad (11)$$

De cualquier manera que se tome la derivada, el cálculo del gradiente permite observar cómo varía la intensidad dentro de la imagen. Tomando variaciones superiores a un umbral se puede encontrar aquellos píxeles que pueden considerarse bordes de regiones u objetos dentro de la escena. En general, la imagen de bordes se construye como una imagen binaria siguiendo un criterio de umbral de la magnitud del gradiente de la siguiente manera

$$g[m, n] = \begin{cases} 0, & \text{si } \nabla f[m, n] < T, \\ 1, & \text{si } \nabla f[m, n] \geq T \end{cases} \quad (12)$$

donde T es el valor de intensidad o brillo umbral y g es la imagen binaria de salida. Los valores de cero y uno corresponden a las intensidades negro y blanco de la imagen de salida. La **Figura 2** muestra la imagen de la microestructura anatómica de una especie maderera, sus derivadas tanto horizontal como vertical, la magnitud (en negativo) y el ángulo del gradiente y la imagen binaria de la magnitud del gradiente siguiendo el criterio de la ecuación (12) con un umbral del 20 % del rango máximo posible en la intensidad de la imagen. Las derivadas se calcularon de acuerdo con las ecuaciones (7) y (8). Sin embargo, las derivadas producen valores negativos. Para obtener las imágenes, los valores de las derivadas fueron trasladadas al rango de 0 (negro) a 1 (blanco) para hacerlas visibles, tal como se muestra. El mismo procedimiento se aplicó para obtener las imágenes de magnitud y ángulo del gradiente. El cálculo del gradiente se realizó a través de la ecuación (10).

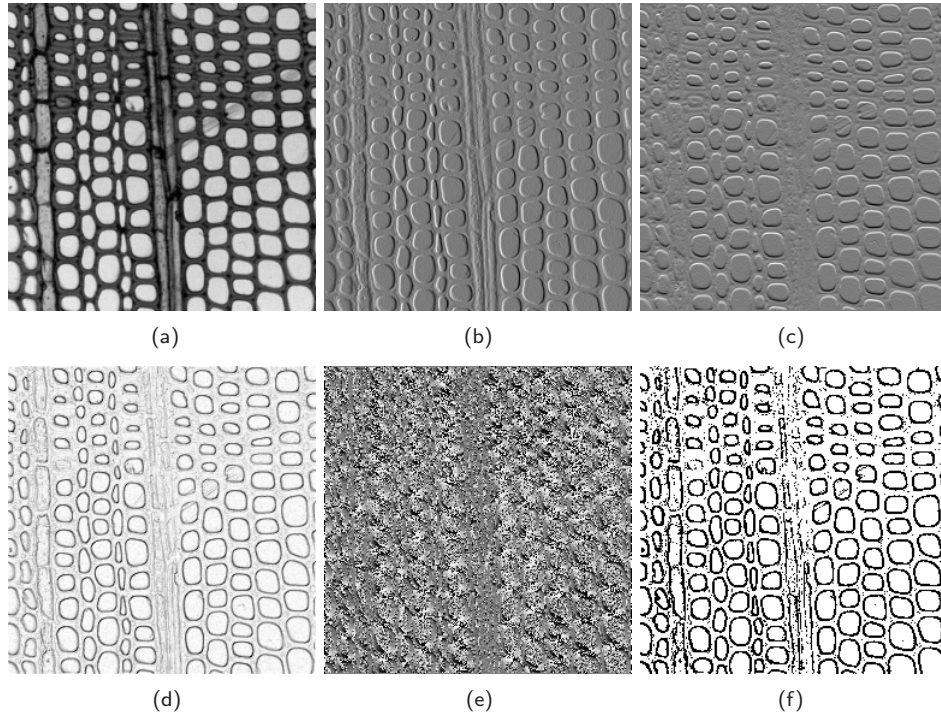


Figura 2: Cálculo del gradiente de intensidad en una imagen. (a) Imagen original, (b) derivada horizontal, (c) derivada vertical, (d) magnitud del gradiente (en negativo), (e) ángulo del gradiente y (f) imagen binaria obtenida de la magnitud del gradiente. (Ver texto para detalles.)

Otra manera de obtener las derivadas es involucrando los píxeles adyacentes a los dos lados del píxel evaluado. Considerando las ecuaciones de diferencia con la aproximación lineal o de primer orden, las derivadas horizontal y vertical de un píxel cualquiera se encuentran como

$$\frac{\partial f(x, y)}{\partial x} \approx \frac{f(x + \Delta x, y) - f(x - \Delta x, y)}{2\Delta x} \quad (13)$$

$$\frac{\partial f(x, y)}{\partial y} \approx \frac{f(x, y + \Delta y) - f(x, y - \Delta y)}{2\Delta y} \quad (14)$$

Bajo las mismas consideraciones que se tuvieron en cuenta para obtener las ecuaciones (7) y (8), las derivadas pueden aproximarse a

$$\frac{\partial f}{\partial x} \approx \frac{f[m+1, n] - f[m-1, n]}{2} \quad (15)$$

$$\frac{\partial f}{\partial y} \approx \frac{f[m, n+1] - f[m, n-1]}{2} \quad (16)$$

Esta manera de obtener las derivadas tiene la desventaja de no considerar el valor de intensidad del píxel evaluado. No obstante, abre la posibilidad de utilizar un número mayor de píxeles como una forma de obtener la derivada dado que su vecindad pasa a ser de 3×3 , a diferencia de la vecindad de 2×2 del método anterior. Un vecindario de este tamaño permite tener el píxel evaluado en el centro, haciendo más fácil y más evidente su cálculo. En efecto, existen operadores de amplio uso en la obtención de las derivadas para extracción de bordes que se estudiarán más adelante.

La operación de convolución bidimensional discreta sobre una imagen digital, para un vecindario rectangular de dimensiones $I \times J$, donde $I = I_2 - I_1 + 1$ y $J = J_2 - J_1 + 1$, está definida como

$$\begin{aligned} y[m, n] &= \sum_{i=I_1}^{I_2} \sum_{j=J_1}^{J_2} x[i, j] h[m-i, n-j] \\ &= \sum_{i=I_1}^{I_2} \sum_{j=J_1}^{J_2} h[i, j] x[m-i, n-j] \end{aligned} \quad (17)$$

donde $x[m, n]$ es la imagen de entrada, $y[m, n]$ es la imagen de salida y $h[i, j]$ se conoce como la *máscara* o *núcleo* y su tamaño corresponde con el vecindario seleccionado. La convolución es entonces una herramienta matemática que permite hallar las derivadas de una imagen. Con una selección del vecindario y las ponderaciones de participación de cada vecino, se determina el núcleo o máscara que se utilizaría en la operación. Se debe tener en cuenta que en la convolución se invierten los índices lo que obliga a hacer previamente una inversión de la máscara. En general, para el cálculo del gradiente, el tamaño de la máscara es pequeño dado que tamaños muy grandes generan mayor tiempo de cómputo de la convolución. Los valores más típicos de tamaño de máscaras son 2×2 y 3×3 . Sin embargo, esta condición no es restrictiva, como se verá más adelante. De manera sencilla, el cálculo de la convolución a través de una máscara h para un vecindario de 3×3 estaría dado por

$$y[m, n] = \sum_{i=-1}^{+1} \sum_{j=-1}^{+1} h[-i, -j] x[m-i, n-j]$$

Se verá a continuación algunas de las máscaras más utilizadas en la detección de bordes. Algunas máscaras no reflejan directamente los operadores descritos para hallar el gradiente pero sus conceptos si se aplican cuando se buscan las variaciones de la intensidad dentro de la imagen. La implementación de los algoritmos a través de las máscaras son sencillas y su valor reside allí.

1.1. Operadores de Sobel

Los operadores de Sobel permiten hallar las derivadas horizontal y vertical de una imagen y las máscaras de 3×3 están definidas de la siguiente manera

$$s_1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad s_2 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (18)$$

Si la imagen x está descrita como una matriz, las derivadas se encuentran entonces como

$$\begin{aligned} \frac{\partial}{\partial x}x[m, n] &= x[m-1, n+1] + 2x[m, n+1] + x[m+1, n+1] \\ &\quad - x[m-1, n-1] - 2x[m, n-1] - x[m+1, n-1] \end{aligned} \quad (19a)$$

$$\begin{aligned} \frac{\partial}{\partial y}x[m, n] &= x[m+1, n-1] + 2x[m+1, n] + x[m+1, n+1] \\ &\quad - x[m-1, n-1] - 2x[m-1, n] - x[m-1, n+1] \end{aligned} \quad (19b)$$

Es decir, la derivada en un píxel se encuentra como el producto de los elementos de la máscara con los elementos (intensidades) de la imagen que están solapados con la máscara. El píxel evaluado es el que está en el centro del vecindario. En la **Figura 3** se puede observar la aplicación de las máscaras de Sobel. A la imagen original mostrada se le encontró las derivadas horizontal y vertical a través de las ecuaciones (19) y se modifican para observar sus resultados en la imagen. La magnitud y el ángulo del gradiente se calculan a través de las ecuaciones (10) y (3). Sus valores se ajustan para observarlos en la imagen de prueba. Finalmente, aplicando el criterio de la ecuación (12) se obtiene la imagen de bordes para un umbral del 20 % del rango. Debido a la mejor presentación visual de la imagen del gradiente, esta se muestra en negativo.

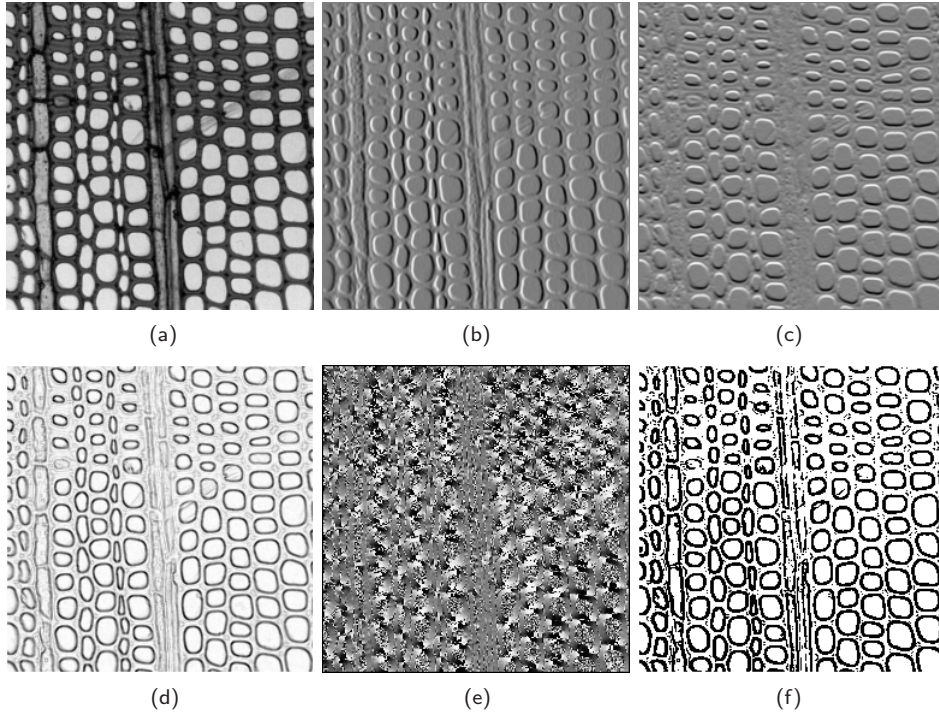


Figura 3: Aplicación de los operadores de Sobel para la detección de bordes. (a) Imagen original, (b) y (c) imágenes producidas por la aplicación de los operadores de Sobel, (d) magnitud del gradiente (en negativo), (e) ángulo del gradiente y (f) imagen binaria obtenida con un umbral del 20 % del rango total de intensidades de la imagen (d).

1.2. Operadores de Prewitt

Los operadores de Prewitt son similares a los de Sobel con la única diferencia que la ponderación de las intensidades de los píxeles cambia. Los máscaras de Prewitt se definen como

$$p_1 = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad p_2 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad (20)$$

De la misma manera como se obtuvo para los operadores de Sobel, la magnitud y el ángulo del gradiente se calculan a través de las ecuaciones (10) y (3) con los operadores de Prewitt. La **Figura 4** muestra el resultado de aplicar las máscaras de Prewitt a la imagen (a) de la **Figura 3**. Para la imagen binaria se utilizó el mismo criterio con el mismo umbral.

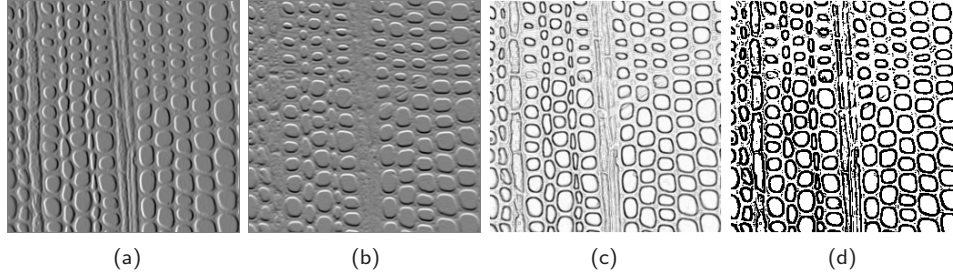


Figura 4: Detección de bordes a través de los operadores de Prewitt. (a) Imagen de la derivada horizontal, (b) imagen de la derivada vertical, (c) magnitud del gradiente (en negativo) y (d) imagen binaria obtenida de la imagen (c) con un umbral del 20 % del rango total de intensidades.

1.3. Operador de Roberts

El operador de Roberts es uno de los más antiguos. A diferencia de los anteriores, toma sólo en cuenta una vecindad de 2×2 píxeles. Las máscaras de convolución están definidas como

$$r_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad r_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad (21)$$

Es un operador muy sencillo. Existen dos formas de cálculo para hallar los bordes con el operador de Roberts

1. la raíz cuadrada de la suma de las diferencias de los vecinos diagonales al cuadrado,

$$\sqrt{|f[m, n] - f[m - 1, n - 1]|^2 + |f[m, n - 1] - f[m - 1, n]|^2} \quad (22)$$

2. la suma de la magnitud de las diferencias de los vecinos diagonales,

$$|f[m, n] - f[m - 1, n - 1]| + |f[m, n - 1] - f[m - 1, n]| \quad (23)$$

La más utilizada de las dos formas es la segunda por su menor complejidad computacional. La **Figura 5** muestra las imágenes obtenidas con la aplicación de las máscaras de Roberts a la imagen (a) de la **Figura 3**. En primera instancia se muestran las imágenes después de calcular la convolución con las máscaras r_1 y r_2 . Luego se muestra la magnitud calculada a través de la segunda forma (ecuación (23)) y finalmente se muestra la imagen binaria considerando el criterio anterior del umbral (20 % de rango de intensidades).

Con el operador de Roberts no se tiene el ángulo del gradiente porque su definición no lo permite. Sólo se puede obtener la magnitud de los puntos de borde sin conocer su orientación. Una de las desventajas más relevantes del operador de Roberts es su alta sensibilidad al ruido debido al poco número de píxeles involucrados en el cálculo del gradiente.

1.4. Máscaras de Kirsch

Las máscaras de Kirsch son ocho y se obtienen considerando una simple y las otras siete rotándolas una posición a la vez. Por esta característica también se conocen como las máscaras de brújula porque la rotación muestra las ocho principales direcciones en un brújula: norte (N), noroeste (NO), oeste (O), suroeste (SO), sur (S), sureste (SE), este (E), noreste (NE). Las máscaras están definidas como

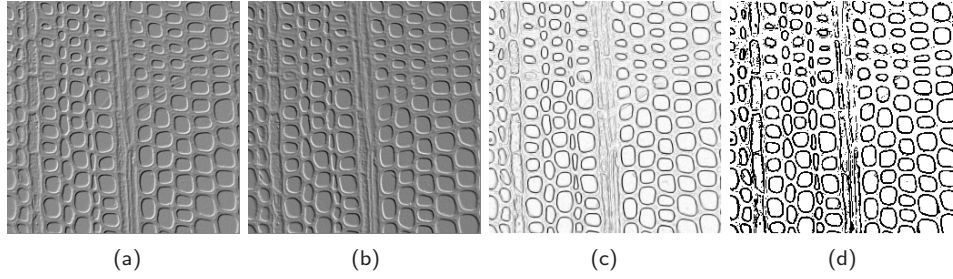


Figura 5: Detección de bordes a partir de los operadores de Roberts. (a) y (b) Imágenes obtenidas después de aplicar las máscaras r_1 y r_2 , respectivamente, (c) magnitud del gradiente (en negativo) y (d) imagen binaria obtenida de la imagen (c) con un umbral del 20 % del rango total de intensidades.

$$\begin{aligned}
 k_0 &= \begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix} & k_1 &= \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix} & k_2 &= \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix} \\
 k_3 &= \begin{bmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix} & k_4 &= \begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix} & & \\
 k_5 &= \begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix} & k_6 &= \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{bmatrix} & k_7 &= \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & 5 \\ -3 & 5 & 5 \end{bmatrix}
 \end{aligned} \tag{24}$$

Cada píxel de una imagen tiene ocho valores obtenidos de la convolución con cada una de las máscaras. La magnitud del gradiente se obtiene entonces con el máximo de estos ocho valores mientras que la dirección queda definida por el ángulo asociado a la máscara que produjo el valor máximo. Los ángulos asociados a las máscaras son

$$\begin{aligned}
 \angle\{k_0\} &= 0^\circ & \angle\{k_1\} &= 45^\circ & \angle\{k_2\} &= 90^\circ & \angle\{k_3\} &= 135^\circ \\
 \angle\{k_4\} &= 180^\circ & \angle\{k_5\} &= 225^\circ & \angle\{k_6\} &= 270^\circ & \angle\{k_7\} &= 315^\circ
 \end{aligned} \tag{25}$$

En la **Figura 6** se muestran cinco imágenes obtenidas de la convolución con las máscaras de Kirsch, la magnitud y el ángulo del gradiente y la imagen binaria de bordes para un umbral del 20 % del rango de intensidades.

1.5. Máscaras de Robinson

Las máscaras de Robinson son similares a las máscaras de Kirsch. Su diferencia radica en la máscara inicial. Las máscaras de Robinson están definidas como

$$\begin{aligned}
 r_0 &= \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} & r_1 &= \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} & r_2 &= \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \\
 r_3 &= \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix} & r_4 &= \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} & r_5 &= \begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix} \\
 r_6 &= \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} & r_7 &= \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}
 \end{aligned} \tag{26}$$

Similarmente, cada máscara está asociada a un ángulo particular. Los ángulos asociados son

$$\begin{aligned}
 \angle\{r_0\} &= 0^\circ & \angle\{r_1\} &= 45^\circ & \angle\{r_2\} &= 90^\circ & \angle\{r_3\} &= 135^\circ \\
 \angle\{r_4\} &= 180^\circ & \angle\{r_5\} &= 225^\circ & \angle\{r_6\} &= 270^\circ & \angle\{r_7\} &= 315^\circ
 \end{aligned} \tag{27}$$

En la **Figura 7** se muestra una imagen resultado de la aplicación de las máscaras de Robinson, la magnitud y el ángulo del gradiente y la imagen binaria de bordes a partir de la magnitud del gradiente para un umbral del 20 % del rango total de intensidades.

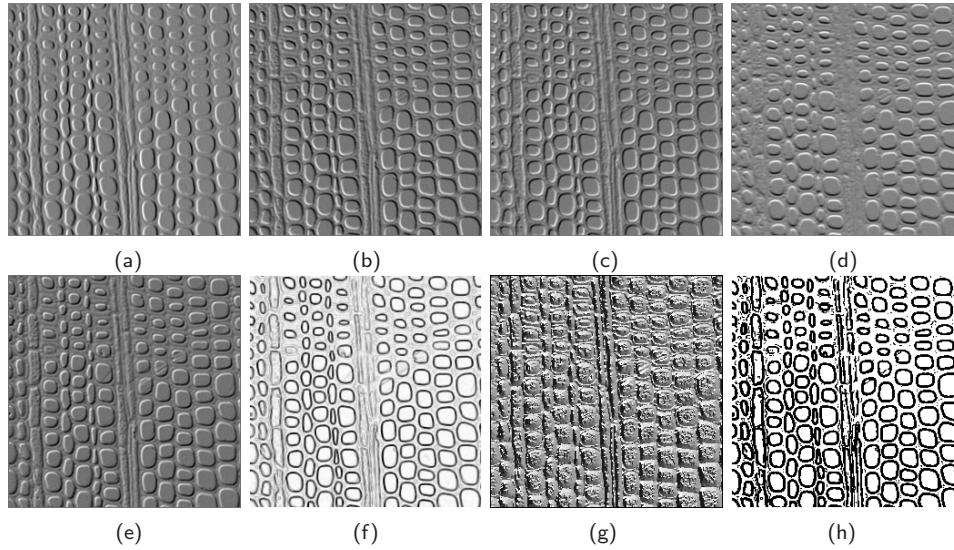


Figura 6: Detección de bordes a través de las máscaras de Kirsch. (a), (b), (c), (d) y (e) Imágenes resultado de la convolución entre la imagen original de la **Figura 3** y las máscaras k_0 , k_1 , k_3 , k_6 y k_7 , respectivamente. (f) Magnitud del gradiente, (g) ángulo del gradiente y (h) imagen binaria obtenida de la imagen (f) para un umbral del 20 % del rango de intensidades.

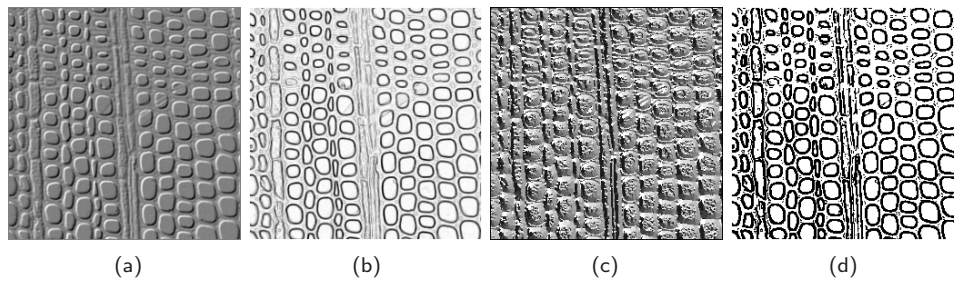


Figura 7: Detección de bordes a través de las máscaras de Robinson. (a) Imagen resultado de la convolución entre la imagen original y la máscara r_7 , (b) magnitud del gradiente (en negativo), (c) ángulo del gradiente y (d) imagen binaria obtenida a partir de la imagen de magnitud del gradiente con un umbral del 20 % del rango total de intensidades.

1.6. Máscaras de Frei-Chen

Las máscaras de Frei-Chen conforman un conjunto completo de vectores base, es decir, las máscaras conforman un conjunto de funciones bases ortogonales o independientes. Entonces, cualquier subimagen de tamaño 3×3 puede representarse como una suma ponderada de las máscaras de Frei-Chen. Los factores de ponderación se encuentran proyectando cada subimagen de 3×3 en cada una de las máscaras. Cada proyección extrae un tipo de comportamiento de la subimagen. Luego, combinando algunos de estos resultados se puede obtener la presencia de bordes. Las máscaras están definidas como

$$\begin{aligned}
 f_1 &= \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -1 & -\sqrt{2} & -1 \end{bmatrix} & f_2 &= \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix} & f_3 &= \frac{1}{\sqrt{8}} \begin{bmatrix} 0 & -1 & \sqrt{2} \\ 1 & 0 & 1 \\ -\sqrt{2} & 1 & 0 \end{bmatrix} \\
 f_4 &= \frac{1}{\sqrt{8}} \begin{bmatrix} \sqrt{2} & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & -\sqrt{2} \end{bmatrix} & f_5 &= \frac{1}{2} \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} & f_6 &= \frac{1}{2} \begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix} \\
 f_7 &= \frac{1}{6} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix} & f_8 &= \frac{1}{6} \begin{bmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{bmatrix} & f_9 &= \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}
 \end{aligned} \tag{28}$$

La aplicación de cada una de las máscaras sobre una imagen produce resultados que resaltan características diferentes. Por ejemplo, en la **Figura 8** se muestra el resultado de convolucionar la imagen (a) de la **Figura 3** con las máscaras f_1 , f_4 , f_6 y f_9 . Se puede observar que las proyecciones destacan características particulares de la imagen. La última prácticamente hace un suavizado, dado que el resultado de la proyección es un promedio.

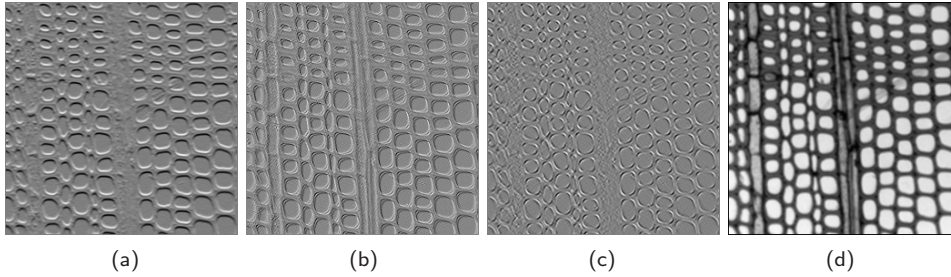


Figura 8: Proyecciones de las subimágenes 1, 4, 6 y 9 de la imagen (a) de la **Figura 3** a través de las máscaras (a) f_1 , (b) f_4 , (c) f_6 y (d) f_9 de Frei-Chen.

La detección de bordes se hace entonces seleccionando un subespacio $\{e_s\}$ y encontrando la proyección relativa de la imagen analizada en dicho subespacio. La proyección se encuentra de la siguiente manera

$$\cos \Theta = \sqrt{\frac{M}{S}} = \sqrt{\frac{\sum_{k \in \{e_s\}} (I_s, f_k)^2}{\sum_{k=1}^9 (I_s, f_k)^2}} \tag{29}$$

donde (I_s, f_k) es la proyección de la subimagen I_s con la máscara f_k . Esta proyección se calcula fácilmente de la siguiente manera: la subimagen y la máscara, ambas de 3×3 , se convierten en vectores columna, es decir

$$I_s = \begin{bmatrix} I_{s1} & I_{s2} & I_{s3} \\ I_{s4} & I_{s5} & I_{s6} \\ I_{s7} & I_{s8} & I_{s9} \end{bmatrix} \xrightarrow{\text{se convierte en}} \mathbf{i}_s = \begin{bmatrix} I_{s1} \\ I_{s2} \\ \vdots \\ I_{s9} \end{bmatrix} \tag{30}$$

y

$$f_k = \begin{bmatrix} f_{k1} & f_{k2} & f_{k3} \\ f_{k4} & f_{k5} & f_{k6} \\ f_{k7} & f_{k8} & f_{k9} \end{bmatrix} \xrightarrow{\text{se convierte en}} \mathbf{f}_k = \begin{bmatrix} f_{k1} \\ f_{k2} \\ \vdots \\ f_{k9} \end{bmatrix} \tag{31}$$

Luego, se obtiene el producto escalar de estos dos vectores, es decir

$$(I_s, f_k) = \mathbf{i}_s^T \mathbf{f}_k = \sum_{n=1}^9 I_{sn} f_{kn} \tag{32}$$

El valor obtenido es un coeficiente que multiplicado con la máscara representa la proyección de la subimagen en la máscara. De hecho, la suma del producto de todas las máscaras por los coeficientes produce la subimagen. Así, los coeficientes representan la proyección en todo el conjunto de vectores base.

En la **Figura 9** se puede observar una muestra de detección de bordes con las máscaras de Frei-Chen, tomando el subespacio $\{e_s\} = \{f_1, f_2, f_3, f_4\}$. En la parte (b) se observa el resultado (en negativo) de M , es decir, la sumatoria del numerador de la fracción dentro del radical de la ecuación (29) que se puede considerar la magnitud de la proyección por su relación con todo el espacio vectorial; en la parte (c) se observa la imagen de Θ para el sub espacio $\{e_s\}$ mencionado arriba; en (d) se observa la imagen binaria de Θ con un umbral del 20% del rango total de intensidades.

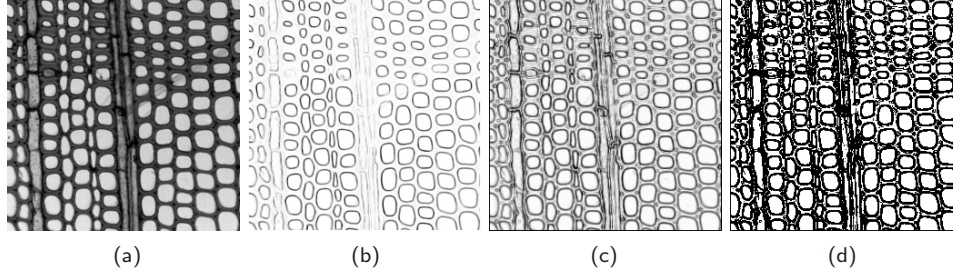


Figura 9: Detección de bordes a través de las máscaras de Frei-Chen. (a) Imagen original, (b) magnitud de la proyección del subespacio $\{e_s\}$ (en negativo), (c) imagen de Θ para el subespacio $\{e_s\}$ y (d) imagen binaria de Θ con un umbral del 20% del rango total de intensidades.

2. Operadores de Segunda Derivada

En la sección anterior, la detección de bordes se hizo a través de operadores que se pueden considerar como de primera derivada. En esencia, los bordes se detectan teniendo en cuenta la variación de la intensidad. Los píxeles donde las variaciones de intensidad son altas se consideran bordes. Estas variaciones se obtienen a través de derivadas. Sin embargo, la segunda derivada suministra información sobre si las variaciones de intensidad son crecientes o decrecientes, a través del signo. La **Figura 10** muestra una imagen sintética donde se tiene un cuadrado blanco en un fondo negro. Los bordes del cuadrado pueden detectarse por medio de la primera derivada. La segunda derivada muestra la dirección de los cambios de intensidad a través de los signos. Esta característica puede utilizarse para encontrar bordes en una imagen cualquiera.

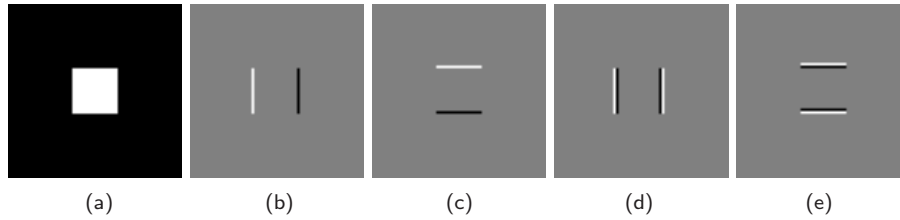


Figura 10: Relación de una imagen y sus derivadas primera y segunda. (a) Imagen original, (b) derivada horizontal, (c) derivada vertical, (d) segunda derivada horizontal y (e) segunda derivada vertical.

Bajo las mismas consideraciones que se tomaron en cuenta para obtener las aproximaciones de las derivadas horizontal y vertical de las ecuaciones (5) y (6), la segunda derivada horizontal se puede obtener aproximadamente a partir del hecho que si

$$g(x, y) = \frac{\partial}{\partial x} [f(x, y)] \approx \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} \quad (33)$$

entonces,

$$\frac{\partial}{\partial x} [g(x, y)] = \frac{\partial^2}{\partial x^2} [f(x, y)] \approx \frac{g(x + \Delta x, y) - g(x, y)}{\Delta x} \quad (34)$$

Luego,

$$\frac{\partial^2}{\partial x^2} [f(x, y)] \approx \frac{f(x + 2\Delta x, y) - 2f(x + \Delta x, y) + f(x, y)}{(\Delta x)^2} \quad (35)$$

De manera similar, para la segunda derivada vertical se tiene,

$$\frac{\partial^2}{\partial y^2} [f(x, y)] \approx \frac{f(x, y + 2\Delta y) - 2f(x, y + \Delta y) + f(x, y)}{(\Delta y)^2} \quad (36)$$

Si, como en la primera derivada, se considera $\Delta x = 1$ y $\Delta y = 1$ los intervalos entre dos píxeles adyacentes, entonces las ecuaciones (35) y (36) se convierten en las ecuaciones de diferencia

$$\frac{\partial^2}{\partial x^2} [f(x, y)] \approx f(x + 2, y) - 2f(x + 1, y) + f(x, y) \quad (37)$$

$$\frac{\partial^2}{\partial y^2} [f(x, y)] \approx f(x, y + 2) - 2f(x, y + 1) + f(x, y) \quad (38)$$

que fueron los operadores que se utilizaron para calcular las derivadas de la **Figura 10**. No obstante los resultados obtenidos, una tendencia es buscar alguna simetría con respecto al píxel que se está evaluando. Bajo las mismas consideraciones que llevaron a deducir las ecuaciones (13) y (14) para la primera derivada, aquí también se pueden encontrar ecuaciones de diferencia que involucren un vecindario alrededor del píxel evaluado. Entonces, si

$$g(x, y) = \frac{\partial}{\partial x} [f(x, y)] \approx \frac{f(x + \Delta x, y) - f(x - \Delta x, y)}{2\Delta x} \quad (39)$$

Entonces, la segunda derivada horizontal estará dada por

$$\frac{\partial^2}{\partial x^2} [f(x, y)] \approx \frac{f(x + 2\Delta x, y) - 2f(x, y) + f(x - 2\Delta x, y)}{(2\Delta x)^2} \quad (40)$$

De manera similar, la segunda derivada vertical estará dada por

$$\frac{\partial^2}{\partial y^2} [f(x, y)] \approx \frac{f(x, y + 2\Delta y) - 2f(x, y) + f(x, y - 2\Delta y)}{(2\Delta y)^2} \quad (41)$$

Con estas definiciones, a continuación se estudian entonces algunos operadores de segunda derivada.

2.1. Operador Laplaciano

La operación laplaciana de una función bidimensional $f(x, y)$ es una operación de segunda derivada definida como

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \quad (42)$$

Como en el caso del gradiente, esta operación se puede realizar de forma digital de varias maneras. Para una imagen digital y si se toman en cuenta las segundas derivadas de las ecuaciones (40) y (41) y se considera que $\Delta x = \frac{1}{2}$ y $\Delta y = \frac{1}{2}$, entonces el operador laplaciano puede aproximarse digitalmente a

$$\begin{aligned} \nabla^2 f[m, n] = & f[m + 1, n] + f[m - 1, n] + f[m, n + 1] \\ & + f[m, n - 1] - 4f[m, n] \end{aligned} \quad (43)$$

La selección de Δx y Δy busca obtener un vecindario de 3×3 pero, adicionalmente, la aproximación de comportamiento lineal de la función de imagen en intervalos pequeños es más válida porque se reduce el intervalo. El operador laplaciano puede considerarse como un núcleo (*kernel*) o máscara de 3×3 , para un vecindario de 4 vecinos, así

$$l_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (44)$$

Obsérvese que la suma de los elementos de la máscara es cero. Esto quiere decir que si la imagen presenta píxeles y sus vecinos con el mismo valor, el operador debe ser cero puesto que es una derivada. No obstante, la definición de l_1 , existen otras máscaras que implementan el operador laplaciano. Por ejemplo, para un vecindario de 8 vecinos, la máscara más utilizada se implementa como

$$l_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (45)$$

También se utilizan operadores laplacianos que acentúan la presencia del píxel central o de sus vecinos. Dos casos se muestran enseguida

$$l_3 = \begin{bmatrix} 2 & -1 & 2 \\ -1 & -4 & -1 \\ 2 & -1 & 2 \end{bmatrix} \quad \text{o} \quad l_4 = \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix} \quad (46)$$

En la **Figura 11** se puede observar la aplicación del operador laplaciano en la detección de bordes. En la Figura se muestran imágenes en negativo (para una mejor presentación) del resultado de la aplicación de los operadores l_1 , l_2 , l_3 y l_4 . Adicionalmente, se presentan las imágenes binarias obtenidas de los resultados para el mismo umbral.

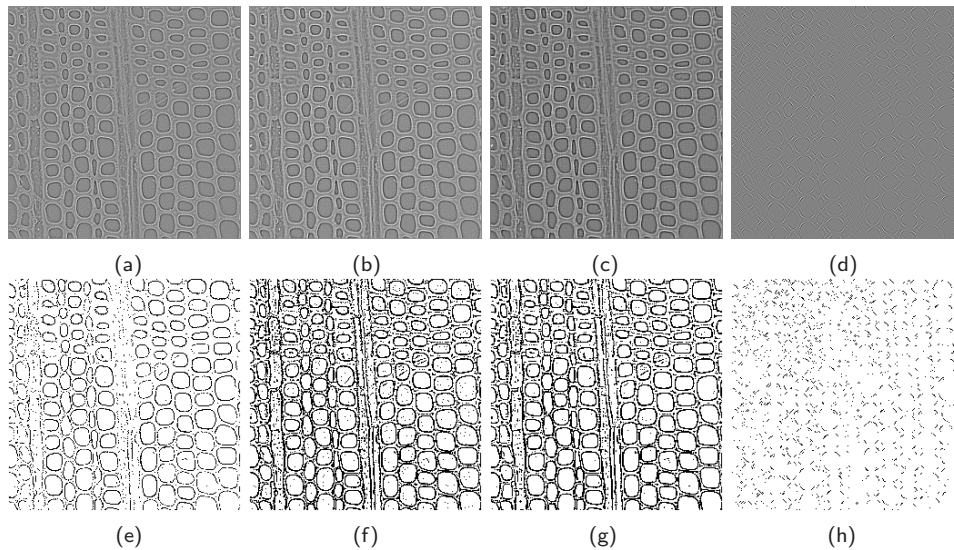


Figura 11: Aplicación del operador laplaciano a la imagen (a) de la **Figura 3**. (a), (b), (c) y (d) Segunda derivada de la imagen a través de los operadores laplacianos l_1 , l_2 , l_3 y l_4 , respectivamente. (e), (f), (g) y (h) Imágenes binarias obtenidas de las imágenes anteriores a través del mismo nivel de umbral.

Debido a que la segunda derivada produce doble borde, como se vio en la introducción, es posible tener los operadores laplacianos con inversión de signo de sus elementos. Esto permitiría seleccionar cuál es más apropiado para la detección de los píxeles que se constituyen como bordes de los objetos. Es conveniente mencionar que el operador laplaciano goza de mucha popularidad entre los procesadores de imágenes. La facilidad de su implementación junto con los resultados obtenidos lo convierten en una herramienta útil en la detección de bordes.

2.2. Operador Laplaciano de la Gaussiana

El operador *Laplaciano de la Gaussiana* o LoG^1 también se conoce como operador de Marr-Hildreth o *Sombrero Mexicano* por la forma que toma la función del vecindario, como se puede ver en la **Figura 12**. Está sustentado en los experimentos neurofisiológicos realizados en la década de los 70, donde se

¹LoG es un acrónimo de las palabras inglesas *Laplacian of Gaussian*.

concluyó que las señales más importantes de la relación entre las intensidades de las imágenes y su interpretación está en los bordes de los objetos de la escena. Estos estudios están planteados en lo que se conoce como la teoría de Marr.

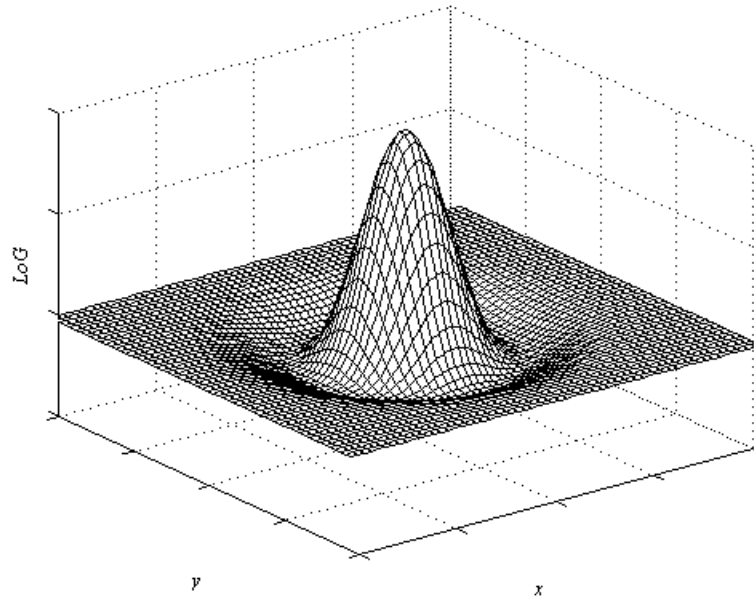


Figura 12: Representación en tres dimensiones de la máscara del operador Laplaciano de la Gaussiana o LoG.

El operador LoG está basado en los cruces por cero de la segunda derivada, así como se consideró en el operador Laplaciano. El hecho es que la primera derivada muestra los cambios de intensidad, característicos de las zonas de bordes, mientras que la segunda derivada produce ceros donde hay extremos de la primera derivada. Los extremos de la primera derivada son los sitios de mayor cambio de intensidad y es exactamente donde la segunda derivada produce los ceros.

Según la teoría de Marr, para llegar a una representación en tres dimensiones, a partir de una en dos dimensiones, se requiere la detección de los cambios de intensidad en la imagen. Por consiguiente, se necesita tanto la medida de los gradientes de intensidad en diferentes escalas como la medida de la localización de estos cambios. Entonces, se propone un filtro de suavizado para seleccionar la información de los cambios de intensidad en diferentes escalas y elegir un filtro que pueda optimizar las dos demandas: los gradientes de intensidad y su posición.

Como se sabe, los gradientes de intensidad representan cambios que tienen relación directa con las frecuencias espaciales contenidas en las imágenes. Luego, el filtro debe ser de suavizado pero limitado en banda de frecuencias para reducir el número de frecuencias que presentan cambios de intensidad en una imagen. No todos los cambios de intensidad interesan en la detección de bordes. Por otro lado, el filtro debe actuar muy cerca de los píxeles donde se presentan los cambios de interés para cumplir con la condición de localización espacial. Estas condiciones de suavizado y localización son conflictivas, sin embargo, se optimizan simultáneamente con la distribución Gaussiana. Entonces, el operador de suavizado gaussiano está dado por

$$g(x, y) = -\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (47)$$

donde x y y son las coordenadas de la imagen y σ es la desviación estándar de la distribución de probabilidad asociada. La desviación estándar está directamente relacionada con el tamaño del vecindario sobre el cual opera el filtro. Los píxeles que están más lejos del centro tienen menor influencia y, de hecho, el efecto producido por los píxeles que están más allá de 4σ se considera despreciable.

La meta es obtener la segunda derivada de una función bidimensional suavizada. Entonces, la aplicación del Laplaciano, que produce la segunda derivada, de una función suavizada se puede representar

como

$$\nabla^2 [g(x, y) * f(x, y)] \quad (48)$$

donde $g(x, y)$ es la gaussiana. Esta operación es la que se conoce como Laplaciana de la Gaussiana o LoG. Puesto que tanto la operación de derivación como de convolución son operaciones lineales, se pueden intercambiar para obtener

$$\nabla^2 [g(x, y)] * f(x, y) \quad (49)$$

Como el primer operando (LoG) de la convolución es independiente del segundo (imagen), el primero se puede calcular previamente, reduciendo la complejidad de la operación planteada en la ecuación. Aplicando el operador Laplaciano sobre la función de distribución Gaussiana se tiene

$$\begin{aligned} \nabla^2 [g(x, y)] &= \frac{\partial^2}{\partial x^2} [g(x, y)] + \frac{\partial^2}{\partial y^2} [g(x, y)] \\ &= K \left[2 - \frac{x^2 + y^2}{\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}} \end{aligned} \quad (50)$$

donde K es una constante de escalado que se introduce para determinar el rango de valores del operador LoG. En general, se busca normalizar la función para que esté en el rango de cero a uno.

Encontrar la segunda derivada de esta manera es muy robusto. Con la función gaussiana se consigue efectivamente el suavizado de la imagen y suprime la influencia de los píxeles que estén más alejados del píxel que se evalúa. Además, el operador $\nabla^2 g$ produce efectivamente una medida de los cambios en la imagen suavizada. Por otro lado, la localización de los cruces por cero no depende de las dimensiones del vecindario. Aunque un aumento de la desviación estándar aumenta la cantidad de píxeles que se toman en cuenta para determinar la segunda derivada, la variación de σ no afecta la localización de los cruces por cero.

Para la implementación digital del operador LoG se pueden seguir los siguientes pasos:

1. Fijar σ y determinar los valores del operador para las diferentes posiciones de filas m y columnas n ; es decir

$$\begin{aligned} \text{LoG}[m, n] &= K \left[2 - \frac{m^2 + n^2}{\sigma^2} \right] e^{-\frac{m^2 + n^2}{2\sigma^2}} \\ &\text{para } m, n = 0, \pm 1, \pm 2, \dots \end{aligned} \quad (51)$$

2. Fijar la escala de los valores mediante la constante K .
3. Redondear los valores obtenidos a través de la ecuación (51) y el factor de escala.
4. Ajustar de forma simétrica los valores anteriores mediante la adición o sustracción de valores pequeños hasta conseguir que la suma de todos sea cero.

El último paso es una condición necesaria para cumplir con el hecho de que si el operador se aplica sobre un píxel que esté dentro de una región con intensidad uniforme, su segunda derivada debe ser cero. Por ejemplo, la **Figura 13** es la máscara del operador LoG obtenida para $\sigma = 2$ y $K = 128$.

A la imagen (a) de la **Figura 3** se le ha aplicado el operador LoG con $\sigma = 2$ y $\sigma = 3$ produciendo máscaras de 17×17 y 25×25 píxeles, respectivamente. La diferencia de tamaño permite un suavizado diferente como se observa en la **Figura 14**. Debido a esta diferencia, la detección de bordes se hace más o menos burda. La Figura muestra las imágenes construidas del resultado de la convolución entre la imagen original y las máscaras. Para hacer más fácil la determinación de los puntos de borde, se obtuvieron las imágenes binarias de la convolución considerando como blanco los valores negativos y como negro los valores positivos. Sin embargo, pueden considerarse diferentes niveles de umbral para obtener estas imágenes. Finalmente se muestran las imágenes de los bordes obtenidas de las imágenes binarias. Este procedimiento se utilizó para realizar el algoritmo más eficientemente debido a que se pueden detectar los cambios de intensidad más fácilmente. Como se puede observar, el nivel de detalle en la detección de los bordes puede hacerse más fino o más burdo a través de la definición de σ . Diferentes resoluciones pueden determinar bordes de objetos de diferente tamaño.

					-1	-1	-1								
				-1	-1	-2	-3	-3	-3	-2	-1	-1			
		-1	-2	-4	-7	-9	-10	-9	-7	-4	-2	-1			
	-1	-3	-6	-12	-18	-22	-24	-22	-18	-12	-6	-3	-1		
-1	-2	-6	-14	-24	-32	-34	-35	-34	-32	-24	-14	-6	-2	-1	
-1	-4	-12	-24	-34	-32	-18	-10	-18	-32	-34	-24	-12	-4	-1	
-2	-7	-18	-32	-32	0	51	78	51	0	-32	-32	-18	-7	-2	
-1	-3	-9	-22	-34	-18	51	150	198	150	51	-18	-34	-22	-9	-3
-1	-3	-10	-24	-35	-10	78	198	256	198	78	-10	-35	-24	-10	-3
-1	-3	-9	-22	-34	-18	51	150	198	150	51	-18	-34	-22	-9	-3
-2	-7	-18	-32	-32	0	51	78	51	0	-32	-32	-18	-7	-2	
-1	-4	-12	-24	-34	-32	-18	-10	-18	-32	-34	-24	-12	-4	-1	
-1	-2	-6	-14	-24	-32	-34	-35	-34	-32	-24	-14	-6	-2	-1	
	-1	-3	-6	-12	-18	-22	-24	-22	-18	-12	-6	-3	-1		
		-1	-2	-4	-7	-9	-10	-9	-7	-4	-2	-1			
			-1	-1	-2	-3	-3	-3	-2	-1	-1				
						-1	-1	-1							

Figura 13: Máscara del operador Laplaciano de Gaussiana o LoG con desviación estándar igual a 2 y factor de escalado de 128.

El operador LoG tiene la desventaja de que introduce un mayor tiempo de procesamiento. Las máscaras pueden llegar a contener algunas decenas de píxeles vecinos involucrados en el cálculo de la segunda derivada. Esto incrementa el tiempo de cómputo de manera significativa frente a los métodos estudiados anteriormente. Sin embargo, cuando este tiempo no es una restricción preponderante, el método del Laplaciano de la Gaussiana es bastante robusto en el cálculo de la segunda derivada y de obtención de los bordes.

Bibliografía

- [1] Gonzalo Pajares Martinsanz y Jesús M. De La Cruz García, *Visión por Computador: Imágenes Digitales y Aplicaciones*, Alfaomega Grupo Editor, S.A. de C.V., México, D.F., 2002.
- [2] John C. Russ, *The Image Processing Handbook*, Fourth Edition, CRC Press LLC, Boca Ratón, Florida, USA, 2002.
- [3] Marcos Faúndez Zanuy, *Tratamiento Digital de Voz e Imagen y Aplicación a la Multimedia*, Alfaomega Grupo Editor, S.A. de C.V., México, D.F., 2001.
- [4] Rafael C. González and Richard E. Woods, *Digital Image Processing*, Second Edition, Prentice-Hall, Inc., New Jersey, USA, 2002.
- [5] Steven W. Smith, *The Scientist and Engineers Guide to Digital Signal Processing*, Second Edition, California Technical Publishing, San Diego, California, 1999.
- [6] William K. Pratt, *Digital Image Processing*, Second Edition, John Wiley & Sons, Inc., New York, USA, 1991.

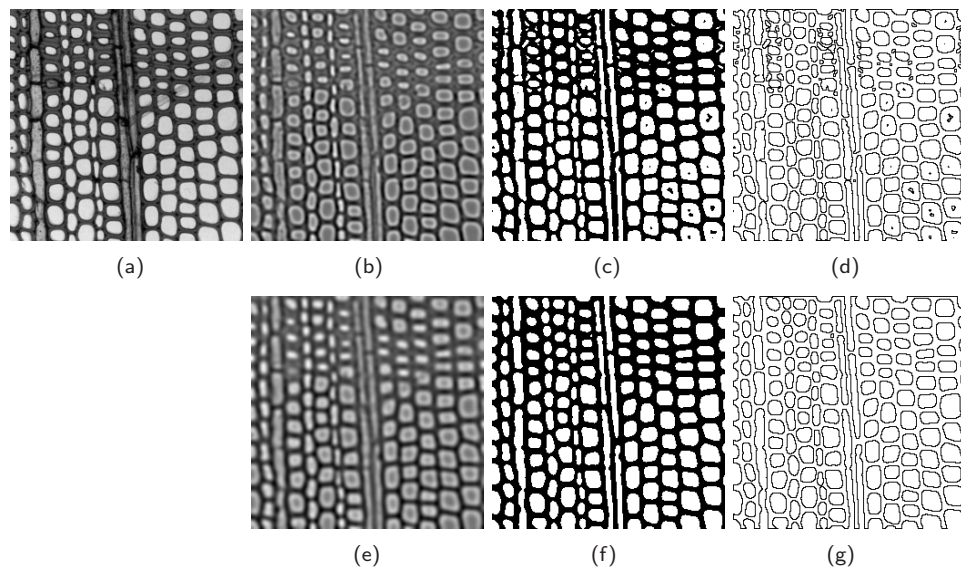


Figura 14: Detección de bordes a través del operador Laplaciano de Gaussiana o LoG. (a) Imagen original. Las imágenes (b) y (e) corresponden al resultado de la convolución entre la imagen original y el operador LoG. (c) y (e) Imágenes binarias de (b) y (e), respectivamente. (d) y (g) Imágenes binarias de los bordes de la imagen original. Se escogió $\sigma = 2$ para las imágenes de arriba y $\sigma = 3$ para las de abajo.