

CAPITULO 3: PASOS REALIZADOS PARA LA DETECCIÓN DE BORDES DE IMÁGENES A COLOR

El algoritmo realizado para la detección de bordes de imágenes a color se basa en el operador Canny. La filosofía del operador Canny consiste en determinar las primeras derivadas parciales con respecto a 'x' e 'y'. Entonces, sobre la base de estos valores, encontrar la magnitud y la dirección del "mejor" borde.

Al ser imágenes a color necesitamos representarlas como una matriz tridimensional $m \times n \times p$ donde $m \times n$ tiene el mismo significado que para el caso de las imágenes en escala de grises mientras que p representa el plano (3.1), para RGB puede ser 1 para el rojo, 2 para el verde y 3 para el azul [10].

$$f(x, y) = \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix} \quad (3.1)$$

A partir de aquí utilizando una imagen como muestra figura 3.1, se describirán una serie de pasos para obtener una detección mediante el operador de Canny.



Figura 3.1

Los pasos realizados en el algoritmo de detección de bordes se enumeran a continuación, los cuales se irán explicando con detalle:

1. Filtrado.
2. Cálculo del gradiente.
3. Cálculo de la magnitud y la dirección.
4. Redondeo a direcciones válidas.

5. Eliminación de no máximos.
6. Umbralización con histéresis.

Para realizarlos de una forma más eficiente procedemos al sub-muestreo de la imagen, que consiste en generar una imagen a partir de la toma de muestras periódicas de la imagen original, de tal forma que ésta quede más pequeña, obteniendo la figura 3.2 como imagen sub-muestreada.



Figura 3.2

3.1 Filtrado:

Un filtrado consiste en aplicar una transformación de forma que se acentúen o disminuyan ciertos aspectos de la imagen. En nuestro caso vamos a realizar un filtrado gaussiano en cada plano RGB para eliminar el ruido y suavizar la imagen.

El filtrado gaussiano se basa en la campana de gauss, la cual podemos ver en la figura 3.3 y en forma de ecuación en (3.2).

La finalidad es obtener una máscara 3x3 para aplicársela a la imagen. Para ello se toma el eje de coordenadas en el píxel central de una ventana 3x3 y se sustituyen los valores de x e y representados en (3.3) en (3.2), obteniendo así la máscara representada en (3.4). Sigma toma el valor por defecto de $\sigma = 0.5$.

$$g(x, y) = e^{-(x^2+y^2)/2\sigma^2} \quad (3.2)$$

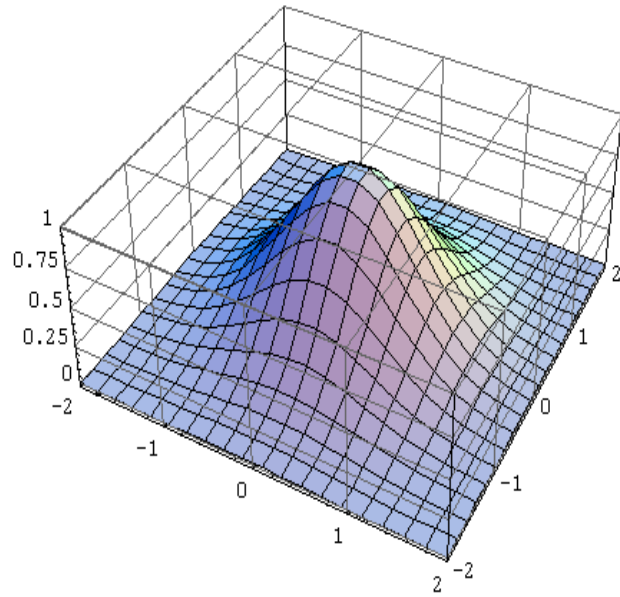


Figura 3.3

$$x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.3)$$

$$g(x, y) = \begin{bmatrix} 0.0183 & 0.1353 & 0.0183 \\ 0.1353 & 1.0000 & 0.1353 \\ 0.0183 & 0.1353 & 0.0183 \end{bmatrix} \quad (3.4)$$

Por último se normaliza sumando todos los elementos de la máscara para obtener en qué medida contribuyen los píxeles vecinos al filtrado del píxel central, obteniendo $sum = 1.6146$. El filtro gaussiano resultante será h (3.5), influyendo el propio píxel central en un 61,93%, los píxeles que se encuentran en las direcciones horizontal y vertical en un 8,38% y los píxeles que se encuentran en las diagonales en un 1,13% [1].

$$h = \frac{g(x, y)}{sum} = \begin{bmatrix} 0.0113 & 0.0838 & 0.0113 \\ 0.0838 & 0.6193 & 0.0838 \\ 0.0113 & 0.0838 & 0.0113 \end{bmatrix} \quad (3.5)$$

Tras calcular el filtro la forma más sencilla de aplicarlo a la imagen es realizar la convolución del filtro con cada plano RGB, como se puede observar en la figura 3.4 tras aplicar el filtro gaussiano se produce un suavizado de la imagen.



Figura 3.4

3.2 Cálculo del gradiente:

Como ya se ha comentado, el operador Canny se basa en el cálculo de las derivadas parciales con respecto a x e y de cada plano RGB (3.6), siendo $f(x, y)$ la imagen y n la dirección definida por el ángulo θ .

$$\frac{\partial f(x,y)}{\partial n} = \begin{bmatrix} \nabla R(x, y) \cdot n \\ \nabla G(x, y) \cdot n \\ \nabla B(x, y) \cdot n \end{bmatrix}; \mathbf{n} = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} \quad (3.6)$$

Obteniendo así las derivadas parciales por separado, a partir de ahora se utilizará notación de la ecuación (3.7) para hacer referencia a las derivadas parciales.

$$Rx = \frac{\partial R}{\partial x} \text{ and } Ry = \frac{\partial R}{\partial y}$$

$$Gx = \frac{\partial G}{\partial x} \text{ and } Gy = \frac{\partial G}{\partial y} \quad (3.7)$$

$$Bx = \frac{\partial B}{\partial x} \text{ and } By = \frac{\partial B}{\partial y}$$

Para realizar el cálculo del gradiente, se toma x_1 como pixel central y a su alrededor sus 8 vecinos, figura 3.5.

\mathbf{x}_6	\mathbf{x}_2	\mathbf{x}_8
\mathbf{x}_4	\mathbf{x}_1	\mathbf{x}_5
\mathbf{x}_9	\mathbf{x}_3	\mathbf{x}_7

Figura 3.5

Las ecuaciones para obtener las derivadas parciales en la dirección horizontal y vertical son aplicadas a cada plano RGB, siendo $F = R, G, B$ según corresponda, (3.8).

$$F_x = \frac{1}{2} * (x_5 - x_4) \quad (3.8)$$

$$F_y = \frac{1}{2} * (x_3 - x_2)$$

3.3 Cálculo de la magnitud y la dirección:

Una vez obtenidas las derivadas con respecto a x y con respecto a y del espacio RGB hay que calcular la magnitud y la dirección del gradiente, para ello utilizamos el operador vector de campo (tensor) explicado en el capítulo 2 apartado 2.2.

El valor absoluto o el máximo autovalor de la matriz $(J^T J)$ es la magnitud del gradiente en la dirección en la que es máximo, (3.9) [9].

$$|m(x, y)| = |\lambda_{max}| \quad (3.9)$$

Donde m representa la magnitud del gradiente y λ los autovalores de la matriz $(J^T J)$.

A continuación se calculan los autovalores de la matriz $(J^T J)$ obteniendo como resultado las ecuaciones (3.10).

$$\lambda_{\pm} = \frac{1}{2} \left(j_{11} + j_{22} \pm \sqrt{(j_{11} - j_{22})^2 + 4j_{12}^2} \right)$$

$$\theta_{+} = \frac{1}{2} \arctan \left\{ \frac{2j_{12}}{j_{11} - j_{22}} \right\} \quad (3.10)$$

Donde:

$$(J^T J) = \begin{bmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \end{bmatrix} = \quad (3.11)$$

$$= \begin{bmatrix} R_x^2 + G_x^2 + B_x^2 & R_x \cdot R_y + G_x \cdot G_y + B_x \cdot B_y \\ R_x \cdot R_y + G_x \cdot G_y + B_x \cdot B_y & R_y^2 + G_y^2 + B_y^2 \end{bmatrix}$$

La dirección en la cual se produce el cambio más grande o la máxima discontinuidad en la imagen cromática, está representado por el autovector de la matriz $(J^T J)$, que corresponde al máximo autovalor. Sustituyendo los valores de la ecuación (3.11) en las ecuaciones (3.10) y utilizando la nomenclatura representada en (3.12), obtenemos las ecuaciones (3.13).

$$C_x = (R_x, G_x, B_x), \quad C_y = (R_y, G_y, B_y) \quad (3.12)$$

$$\tan(2\theta) = \frac{2 \cdot C_x \cdot C_y}{\|C_x\|^2 - \|C_y\|^2} \quad (3.13)$$

$$m^2 = \|C_x\|^2 \cdot \cos(\theta)^2 + 2 \cdot C_x \cdot C_y \cdot \sin(\theta) \cdot \cos(\theta) + \|C_y\|^2 \cdot \sin(\theta)^2$$

Donde θ representa la dirección y m la magnitud de un borde.

Con estos cálculos finalmente obtenemos la magnitud ' m ' y la dirección ' θ ' para cada píxel de la imagen. En la figura 3.6 representamos la magnitud donde ya podemos empezar a observar cómo van destacando los bordes.

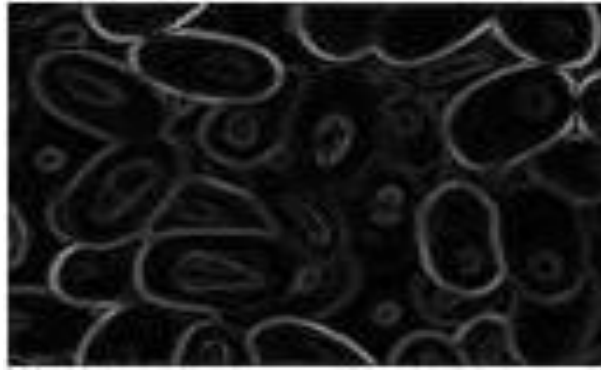


Figura 3.6

3.4 Redondeo a direcciones válidas:

Una vez calculadas la dirección θ para cada píxel se redondean las direcciones para clasificarlas en horizontal, vertical y diagonal. La finalidad de esta función es comparar la magnitud del gradiente de un píxel con sus 2 vecinos según su dirección normal al borde, es decir, si la dirección es vertical tendríamos como pixeles vecinos el de arriba y el de abajo, si la dirección es horizontal los pixeles vecinos estarían a la izquierda y a la derecha y por último si la dirección fuera diagonal los pixeles vecinos serían los que están justamente al lado al trazar la diagonal.

Para conseguir esto primero tenemos que redondear las direcciones, dividimos la circunferencia unidad en sectores, figura 3.7, y según $\theta(\theta)$ esté en un sector u otro se le asigna una dirección [11].

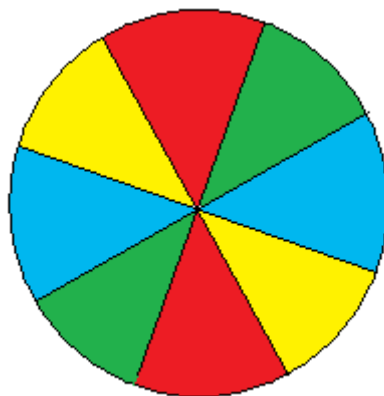


Figura 3.7

- Si θ está en los sectores rojos (entre -22.5° y 22.5° o entre -157.5° y 157.5°) la dirección es vertical y como hemos dicho antes los pixeles vecinos serían el de arriba y el de abajo.
- Si θ está en los sectores azules (entre 67.5° y 112.5° o entre -67.5° y -112.5°) la dirección es horizontal y los pixeles vecinos están a la izquierda y a la derecha.
- Si por el contrario está en los sectores amarillos y verdes (entre 22.5° y 67.5° , entre -157.5° y -112.5° , entre 112.5° y 157.5° o entre -22.5° y -67.5°) la dirección es diagonal y los pixeles vecinos son los que se encuentran en la diagonal.

3.5 Eliminación de no máximos:

La eliminación de no máximos consiste en eliminar aquellos pixeles que comparados con sus vecinos tengan una magnitud menor [11]. Para ello en cada pixel miramos su dirección según lo explicado en el paso anterior, así sabemos cuáles son sus 2 vecinos.

Una vez sabemos que vecinos hay que mirar comparamos el valor de la magnitud del pixel con la magnitud de sus vecinos. El pixel sólo será considerado como borde si su magnitud es mayor que la magnitud de sus vecinos. Con este paso conseguimos adelgazar el borde, el resultado se muestra en la figura 3.8.



Figura 3.8

3.6 Umbralización con Histéresis:

La Umbralización consiste en imponer un umbral, si los píxeles superan ese umbral serán considerados como bordes. Pero aparece un problema si imponemos un umbral muy alto perdemos parte de los bordes, por el contrario si usamos un umbral bajo aparecería ruido, por ello utilizamos la umbralización con histéresis en la que usamos dos umbrales T_h y T_l , siendo T_h mayor que T_l [11].

Recorremos la matriz obtenida de la eliminación de no máximos, si la magnitud del píxel supera el umbral T_h se considera un píxel fuerte, si por el contrario la magnitud del píxel es superior al umbral T_l pero inferior a T_h es considerado un píxel débil.



Figura 3.9

En la figura 3.9 se representan los píxeles débiles y los píxeles fuertes. Como se puede observar en la imagen hay varios píxeles que no deberían ser considerados como borde, para solucionarlo consideramos borde todos los píxeles fuertes más los píxeles débiles que sean vecinos de los píxeles fuertes. Obteniendo una detección mucho más limpia y en la que desaparecen muchos de los píxeles interiores a los bordes, figura 3.10.



Figura 3.10