

Detección de bordes utilizando filtros de primera derivada

Walter Alejandro Moreno Ramírez
Departamento de Estudios Multidisciplinarios
Universidad de Guanajuato
Yuriria, Guanajuato
Correo: wa.morenoramirez@ugto.mx

Abstract—This article describes what are the edges of an image, as they can be detected by obtaining the gradient of a region of the image. Its advantages and applications will also be stated.

Index Terms—Pixel, convolución, filtros, gradiente, primera derivada, función, C++, OpenCV, ventana, máscara, vecindario.

I. INTRODUCCIÓN

Los bordes de una imagen digital se pueden definir como un cambio entre dos regiones de niveles de gris significativamente distintos, normalmente se toman las transiciones entre regiones con nivel de gris 0 a nivel de gris 255. Los bordes nos dan información sobre las fronteras de los objetos y poder reconocerlos.

La derivada de una señal continua proporciona las variaciones locales con respecto a una sola variable, de forma que el valor de la derivada es mayor cuanto más rápidas son los cambios o transiciones de niveles de gris.

En el caso de funciones de dos variables $f(x, y)$, la derivada es un vector que apunta en la dirección de la máxima variación de $f(x, y)$ y cuyo módulo es proporcional a dicha variación. Este vector se denomina gradiente y se define en la Ecuación (1).

$$\nabla f(x, y) = [f_x(x, y)\hat{i}, f_y(x, y)\hat{j}] \quad (1)$$

Cuya magnitud esta definida por la Ecuación (2).

$$\|\nabla f(x, y)\| = \sqrt{(f_x(x, y))^2 + (f_y(x, y))^2} \quad (2)$$

Y su angulo esta dado por la Ecuación (3).

$$\theta = \arctan\left(\frac{f_y(x, y)}{f_x(x, y)}\right) \quad (3)$$

II. METODOLOGÍA

Un imagen digital se puede ver como una función de dos variables, pero al no existir un número infinito de píxeles entre pares de píxeles adyacentes será una función discreta. Para una función bidimensional discreta, las distintas aproximaciones del operador gradiente se basan en diferencias entre los niveles de grises de la imagen. La derivada parcial $f_x(x, y)$ (gradiente de fila $g_f(i, j)$) puede aproximarse por la diferencia de los píxeles adyacentes de la misma fila. Se define por la ecuación (4).

$$f_x(x, y) \approx \nabla_x f(x, y) = f(x, y) - f(x - 1, y) \quad (4)$$

Lo que nos da como resultado un gradiente en forma de matriz fila como se muestra en la Figura 1.

-1	1
----	---

Figura 1: Matriz resultante de discretizar en el eje x.

La discretización del vector gradiente en el eje Y ($G_c(i, j)$), se define en la Ecuación (5).

$$f_y(x, y) \approx \nabla_y f(x, y) = f(x, y) - f(x, y - 1) \quad (5)$$

Lo que nos da como resultado un gradiente en forma de matriz fila como se muestra en la Figura 1.

-1
1

Figura 2: Matriz resultante de discretizar en el eje y.

Al par de matrices resultantes se les llamará máscaras. Para poder obtener los bordes de una imagen, se debe hacer una convolución entre las máscaras y una sección de la imagen de las mismas dimensiones que la máscara. El resultado de dicha convolución será una imagen en la cual se notarán más, dependiendo de la máscara, ya sea los bordes horizontales o verticales.

El presente trabajo se baso en una práctica anterior, ya que se había hecho un filtrado pasa bajas de una imagen utilizando una matriz como filtro y aplicando una convolución entre la imagen de entrada y el filtro. Se utilizaron las mismas funciones pero se modificaron para poder aplicar las distintas máscaras.

Cada máscara tiene 3 filas y 3 columnas. De acuerdo al pseudocódigo mostrado en la Figura 3. se obtiene el vecindario del pixel central donde nos concentramos para cada iteración. El vecindario debe tener las mismas dimensiones que la máscara para poder realizar la convolución.

Una vez que se obtiene el vecindario, se realiza la convolución entre el vecindario y la máscara. Las operaciones se realizan elemento a elemento. Las máscaras tendrán valores de -1, 0 y 1, lo que nos dará como resultado valores menores a cero y mayores a 255, por esto es necesario realizar una etapa de acotamiento, ya que el objeto Mat de las librerías OpenCV almacena sus datos en un puntero de tipo unsigned char, y si tenemos números negativos nos puede dar problemas. Por lo anterior es que se realiza dicha etapa de acotamiento, donde los valores negativos se guardan como 0 y los valores mayores a 255 se guardan en 255, cualquier valor que este dentro del rango será considerado como tal, un valor válido.

```

1 para i=0 hasta rows
2 para j=0 hasta cols
3     sum=0; mi=0;
4     para ni=i-N hasta i+N
5         mj=0
6         para nj=j-N hasta j+N
7             suma += mask(mi,mj)*image(ni, nj)
8             mj++
9         fin para
10        mi++
11    fin para
12
13    si suma < 0
14        imageOut(i,j) = 0
15    sino si sum > 255
16        imageOut(i,j) = 255
17    sino
18        imageOut(i,j) = suma
19    fin para
20 fin para

```

Figura 3: Pseudocódigo que indica el procedimiento para realizar la convolución entre la imagen y la máscara, y la etapa de acotamiento.

Se aplicaron distintos operadores, cada uno con sus respectivas máscaras.

Operador de Roberts

Las máscaras para este operador se muestran en la Figura 4.

0	0	0
0	0	1
0	-1	0

-1	0	0
0	1	0
0	0	0

Figura 4: Gradiente fila y gradiente columna respectivamente.

Con estos gradientes se obtiene una buena respuesta a los bordes diagonales, esto debido a la disposición de sus elementos en las máscaras. Ofrece buenas prestaciones en cuanto a localización.

Operadores de Prewitt, Sobel y Frei-Chen

Los tres operadores pueden formularse de forma conjunta ya que tienen una disposición de sus elementos igual. Se utilizan las máscaras de convolución mostradas en la Figura 5.

$$\frac{1}{2+K} \begin{bmatrix} 1 & 0 & -1 \\ K & 0 & -K \\ 1 & 0 & -1 \end{bmatrix}$$

$$\frac{1}{2+K} \begin{bmatrix} -1 & -K & -1 \\ 0 & 1 & 0 \\ 1 & K & 1 \end{bmatrix}$$

Figura 5: Gradiente fila y gradiente columna respectivamente.

En el operador Prewitt ($k = 1$) se involucran a los vecinos de filas/columnas adyacentes para proporcionar mayor inmunidad al ruido.

El operador Sobel ($k = 2$) en teoría es más sensible a los bordes diagonales que el operador de Prewitt pero en la práctica hay poca diferencia entre ellos.

Frei-Chen ($k = \sqrt{2}$), el gradiente es el mismo para bordes verticales, horizontales y diagonales.

III. RESULTADOS

Para realizar las pruebas se utilizará sólo una imagen, esto debido a la cantidad de operadores.

La Figura 6. muestra la imagen que se utilizará para las pruebas.



Figura 6: Imagen de un juego de SuDoKu con características interesantes para obtener mediante esta técnica.

Al aplicar el gradiente de Prewitt los bordes se obtienen al realizar la convolución y obtener el gradiente utilizando las máscaras vertical y horizontal.

Si se aplica el gradiente de Prewitt con la máscara a 45 grados se obtiene un resultado muy similar, ya que esta máscara es una generalización de las máscaras vertical y horizontal. Esto se puede apreciar mejor en la Figura 7.

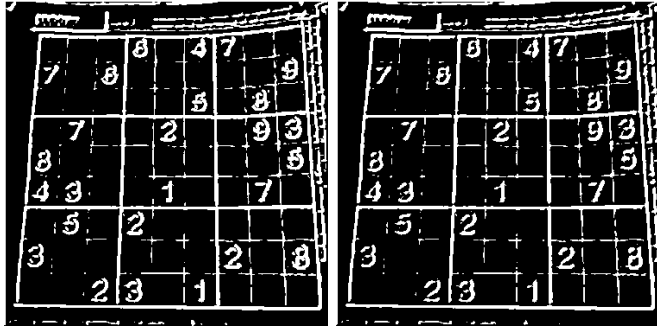


Figura 7: Bordes obtenidos con el gradiente de Prewitt horizontal y vertical, y a 45 grados respectivamente.

Si aplicamos el gradiente de Sobel y, de acuerdo a sus características debería mostrar resultados semejantes al gradiente de Prewitt, aunque reduciendo el ruido ya que concentra los datos en el centro de la máscara.

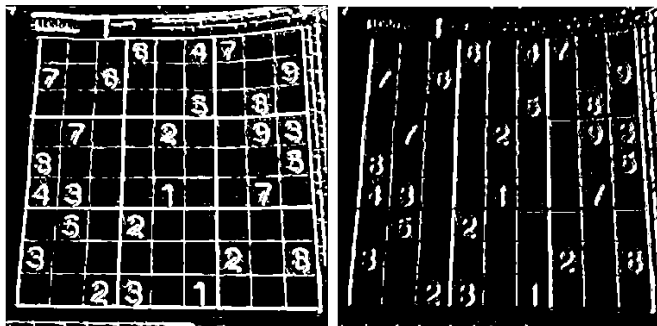


Figura 8: Bordes obtenidos con el gradiente de Sobel horizontal y vertical, y a 45 grados respectivamente.

De acuerdo a la Figura 8, el resultado de aplicar la máscara a 45 grados es muy distinta, ya que parece responder sólo a las lineales verticales y las horizontales las atenúa. En cambio aplicando las máscaras en horizontal y vertical se obtienen mejores resultados que utilizando el gradiente de Prewitt.

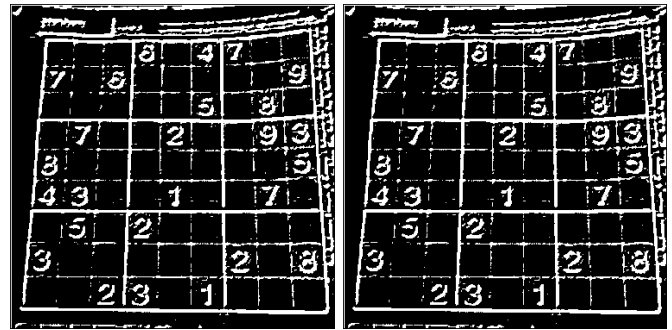


Figura 9: Bordes obtenidos con el gradiente de Roberts horizontal y vertical, y a 45 grados respectivamente.

En la Figura 9, se muestra los resultados de aplicar el gradiente de Roberts. Los resultados son aún mejor que con el gradiente de Sobel. Se obtienen una mejora considerable al detectar la cuadrícula del SuDoKu.

En la Figura 10, se muestra la imagen resultante de aplicar el gradiente de Frin-Chen.

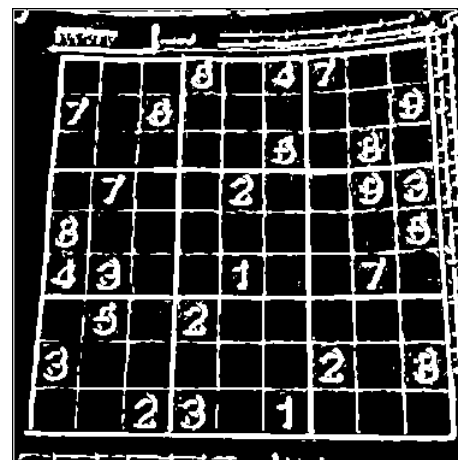


Figura 10: Bordes obtenidos con el gradiente de Frin-Chen horizontal y vertical.

El gradiente de Frin-Chen es el gradiente que generaliza todos los demás, y se puede observar en el resultado, es el gradiente que mejor detecta la cuadrícula del SuDoKu y suprime el ruido mejor que los otros gradientes.

Los resultados de la Figura 7, y Figura 8, se obtuvieron umbralizando la imagen resultantes al obtener la magnitud del gradiente, el umbral utilizado fue 30. En cambio en la Figura 9, se utilizó un umbral de 5. Por último, para la imagen resultado de la Figura 10, se utilizó un umbral de 20.

IV. CONCLUSIONES

En esta práctica solo se mostraron cuatro tipos de gradientes para obtener los bordes de una imagen, en realidad son 8 filtros para cada gradiente, uno para cada 45 grados. Todos los gradientes se pueden utilizar para diferentes tipos de imágenes ya que se puede obtener resultados distintos dependiendo de la imagen y la máscara utilizada.

Como resultado se obtienen diferentes bordes, aunque las diferencias son mínimas, en algunos casos se elimina el ruido de la imagen del resultado, como es el caso cuando se utiliza la máscara de Sobel.

Otro dato importante es que al momento de umbralizar el valor del umbral afecta mucho al resultado de la imagen.

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$