



## CENTRO DE ENSEÑANZA TECNICA INDUSTRIAL PRACTICAS DE LABORATORIO

| INGENIERÍA<br>CARRERA                   | PLAN DE<br>ESTUDIO | CLAVE<br>ASIGNATURA | NOMBRE DE LA<br>ASIGNATURA |
|---|--------------------|---------------------|----------------------------|
| INGENIERÍA DE DESARROLLO<br>DE SOFTWARE | 2025-B             | 19SDS32             | Procesamiento de Imágenes  |
| ALUMNO                                  |                    | FECHA               | EVALUACIÓN                 |

| PRÁCTICA<br>No. | LABORATORIO DE<br>COMPUTACIÓN No | NOMBRE DE LA<br>PRÁCTICA | DURACIÓN<br>(HORAS) |
|-----------------|----------------------------------|--------------------------|---------------------|
| 6               | LCS                              | FILTROS DE BORDES        | 2                   |

### INTRODUCCIÓN

En el procesamiento de imágenes digitales, los filtros de detección de bordes son herramientas fundamentales para resaltar las transiciones significativas de intensidad entre píxeles adyacentes. Los bordes representan cambios bruscos en la intensidad de una imagen y suelen indicar los límites de los objetos o regiones dentro de ella. La detección de bordes permite identificar contornos que son útiles en tareas como el reconocimiento de patrones, la visión por computadora, y la segmentación de objetos.

La detección de bordes facilita el análisis de las imágenes al simplificar la representación del contenido. Por ejemplo, al convertir una imagen en una serie de contornos, se hace más fácil reconocer objetos y patrones, permitiendo así la extracción de características importantes sin procesar toda la información de la imagen original.

### OBJETIVO (COMPETENCIA)

El objetivo de esta práctica es implementar y comparar diferentes **filtros de detección de bordes** (Sobel, Prewitt y Roberts) utilizando MATLAB para identificar los contornos de los objetos en una imagen en escala de grises.

### FUNDAMENTO

El **filtro de Sobel** es un operador de detección de bordes que utiliza dos máscaras de tamaño 3x3 para calcular las variaciones de intensidad en las direcciones horizontal y vertical. Además de detectar los bordes, este filtro suaviza ligeramente la imagen, lo que

lo hace más robusto frente al ruido. La magnitud del gradiente, que indica la intensidad del borde, se obtiene combinando los gradientes en ambas direcciones. El filtro de Sobel es especialmente útil cuando se trabaja con imágenes que contienen algo de ruido y se requieren contornos precisos.

El **filtro de Prewitt** es una variante más sencilla del filtro de Sobel, también con máscaras de  $3 \times 3$ . A diferencia del Sobel, las máscaras de Prewitt no aplican tanto suavizado, lo que lo hace menos preciso en situaciones con ruido. Sin embargo, es adecuado para detectar bordes suaves o cambios graduales de intensidad en la imagen. Su implementación es simple, y es una opción útil en aplicaciones donde los contornos no necesitan alta precisión.

El **filtro de Roberts** emplea dos máscaras de tamaño  $2 \times 2$  para detectar bordes en direcciones diagonales. Es un filtro más rápido y eficiente debido al pequeño tamaño de sus máscaras, pero también es más sensible al ruido. Detecta con precisión cambios abruptos de intensidad, lo que lo hace ideal para identificar bordes finos y detalles pequeños en la imagen. Sin embargo, su simplicidad lo hace menos robusto en entornos con ruido significativo.

### METODOLOGÍA (DESARROLLO DE LA PRACTICA)

El código permite aplicar los filtros de Sobel, Prewitt y Roberts a una imagen en escala de grises, comparando los resultados de cada uno en una figura de múltiples subgráficos. Primero, se carga la imagen original usando `imread` y se convierte a escala de grises mediante `rgb2gray`, ya que los filtros de detección de bordes operan únicamente sobre intensidades y no en imágenes en color.

Luego, se crea una ventana para mostrar los resultados de los tres filtros. Cada uno de ellos se implementa en funciones separadas: `sobelFiltro`, `prewittFiltro` y `robertsFiltro`. Cada filtro devuelve una imagen en escala de grises donde los bordes detectados están resaltados. En esta figura, se muestran cuatro subgráficos: la imagen en escala de grises original y las imágenes resultantes de aplicar los tres filtros.

Las funciones `sobelFiltro` y `prewittFiltro` utilizan máscaras de  $3 \times 3$  para detectar bordes en direcciones horizontal y vertical. En estas funciones, se recorre cada píxel (exceptuando los bordes de la imagen) mediante dos bucles anidados, aplicando las máscaras correspondientes para calcular los gradientes en ambas direcciones. La magnitud del borde se obtiene combinando los gradientes y normalizando los valores para que se encuentren en el rango de 0 a 255.

La función `robertsFiltro` utiliza máscaras más pequeñas, de  $2 \times 2$ , para detectar bordes en direcciones diagonales. Es más eficiente, pero también más sensible al ruido. La magnitud del gradiente se calcula de forma similar a los otros filtros, combinando las contribuciones de cada dirección diagonal y normalizando los valores obtenidos. Finalmente, la figura muestra los resultados de los tres filtros para comparar sus capacidades y diferencias en la detección de bordes.

#### Código:

```
clear all;
clc;
A = imread('mujer.png');
gr = rgb2gray(A);
figure('Name', 'Filtro Sobel', 'NumberTitle', 'off');
sobel = sobelFiltro(gr);
prewitt = prewittFiltro(gr);
robert = robertsFiltro(gr);
subplot(2,2,1);
imshow(gr), title('Grises Original');
subplot(2,2,2);
imshow(sobel), title('Sobel');
subplot(2,2,3);
imshow(sobel), title('Prewitt');
subplot(2,2,4);
imshow(robert), title('Robert');
function GtN = sobelFiltro(gr)
    [m, n] = size(gr);
    suave = double(gr);

    Gx = zeros(size(suave));
    Gy = zeros(size(suave));

    for r = 2:m-1
        for c = 2:n-1
            Gx(r, c) = -1 * suave(r-1, c-1) - 2 * suave(r-1, c) - 1 * suave(r-1, c+1) + ...
                suave(r+1, c-1) + 2 * suave(r+1, c) + suave(r+1, c+1);
            Gy(r, c) = -1 * suave(r-1, c-1) - 2 * suave(r, c-1) - 1 * suave(r+1, c-1) + ...
                suave(r-1, c+1) + 2 * suave(r, c+1) + suave(r+1, c+1);
        end
    end

    Gt = sqrt(Gx.^2 + Gy.^2);

    VmaxGt = max(max(Gt)); % Se obtiene el valor máximo del gradiente
    GtN = (Gt / VmaxGt) * 255;
    GtN = uint8(GtN);
```

```
B = GtN > 100;
end
function GtN = prewittFiltro(gr)
    [m, n] = size(gr);
    suave = double(gr);

    Gx = zeros(size(suave));
    Gy = zeros(size(suave));

    for r = 2:m-1
        for c = 2:n-1
            Gx(r, c) = -1 * suave(r-1, c-1) + 1 * suave(r-1, c) - 1 * suave(r-1,
c+1) + ...
                        suave(r+1, c-1) + 1 * suave(r+1, c) + suave(r+1, c+1);
            Gy(r, c) = 1 * suave(r-1, c-1) + 1 * suave(r, c-1) + 1 * suave(r+1,
c-1) + ...
                        -1*suave(r-1, c+1) -1 * suave(r, c+1) -1 *suave(r+1,
c+1);
        end
    end

    Gt = sqrt(Gx.^2 + Gy.^2);

    VmaxGt = max(max(Gt)); % Se obtiene el valor máximo del gradiente
    GtN = (Gt / VmaxGt) * 255;
    GtN = uint8(GtN);
    B = GtN > 100;
end
function GtN = robertsFiltro(gr)
    [m, n] = size(gr);
    suave = double(gr);

    Gx = zeros(size(suave));
    Gy = zeros(size(suave));

    for r = 1:m-1
        for c = 1:n-1
            Gx(r, c) = suave(r, c) - suave(r+1, c+1);
            Gy(r, c) = suave(r+1, c) - suave(r, c+1);
        end
    end

    Gt = sqrt(Gx.^2 + Gy.^2);
    VmaxGt = max(max(Gt));
    GtN = uint8((Gt / VmaxGt) * 255);
end
```



## CENTRO DE ENSEÑANZA TECNICA INDUSTRIAL

### PRACTICAS DE LABORATORIO

## RESULTADOS Y CONCLUSIONES

## REFERENCIAS

1. **MathWorks España.** (s. f.). *Detección de bordes - MATLAB & Simulink*. MathWorks. Recuperado de <https://es.mathworks.com>
2. **UPIITA-IPN.** (2020). *Extracción de Bordes; Operadores Sobel, Prewitt y Roberts*. Boletín UPIITA. Recuperado de <https://boletin.upiita.ipn.mx>
3. **Universidad de Jaén.** (2005-2006). *Detección de bordes en una imagen. Segmentación*. Área de Ingeniería de Sistemas y Automática. Recuperado de <https://www4.ujaen.es>
4. **Universidad Nacional de Córdoba.** (s. f.). *Procesamiento de imágenes con derivadas - Detección de esquinas y bordes*. Recuperado de <https://www.famaf.unc.edu.ar>

| Elavoro                | Observaciones | Evaluacion |
|------------------------|---------------|------------|
| DR. Gerardo García Gil |               |            |