



**Instituto Politécnico Nacional
Escuela Superior de Cómputo**



Práctica 8

Filtros Lineales

**Procesamiento Digital de Imágenes
Grupo 4BV1**

Alumno: Padilla García Andrea Miranda

Objetivo: Implementar la convolución y aplicar los métodos vistos en clase

Introducción

Para esta práctica vamos a ver el concepto de *convolución* esto se refiere a una operación matemática que combina dos funciones para producir una tercera a través de la cual resulta posible **detectar patrones** en la imagen. Estas funciones corresponden a la imagen original y a una máscara de filtrado. La máscara es una matriz de tamaño reducido (por ejemplo 3×3 , 7×7 u 11×11) que se desplaza sobre la imagen píxel por píxel.

Para cada posición del *kernel* sobre la imagen, se realiza el siguiente procedimiento:

1. Se coloca el centro de la máscara sobre el píxel actual de la imagen.
2. Se multiplican los valores de los píxeles vecinos por los coeficientes correspondientes de la máscara.
3. Se suman todos los productos obtenidos.
4. El resultado de esta suma se asigna como el nuevo valor del píxel en la imagen de salida.

De forma matemática, la convolución se expresa como:

$$g(x, y) = \sum_{i=-O/2}^{i=O/2} \sum_{j=-P/2}^{j=P/2} f(i, j) \cdot h(x-i, y-j)$$

Este proceso se repite para todos los píxeles de la imagen, generando una nueva imagen filtrada.

¿Qué es un filtro?

Como su nombre lo indica, el propósito de los filtros es filtrar algunos componentes innecesarios (en otras palabras, el ruido) o extraer algunas estructuras subyacentes a la imagen. Para implementar filtros lineales, es necesario utilizar la convolución para obtener la imagen resultante.

Filtros pasa-altas

El filtro pasa alto busca las frecuencias altas que están en los cambios fuertes de niveles de gris en una imagen, es decir en los *contornos*. Los contornos de una imagen contienen mucha de la información en la imagen, ellos contienen a los objetos, los definen y les da tamaño, y algunas veces textura. Un contorno se puede definir como el cambio de intensidad pasando de un nivel de intensidad bajo a uno alto y viceversa.

En este trabajo se implementaron distintos operadores de detección de bordes, entre ellos Roberts, Prewitt, Sobel y Frei-Chen, así como operadores de gradiente en compás como Prewitt, Kirsch y Robinson. Estos operadores utilizan máscaras direccionales que permiten detectar bordes en diferentes orientaciones.



Filtros pasa-bajas

Los filtros pasa bajas tienen como objetivo **suavizar la imagen**, reduciendo el ruido y eliminando detalles finos. Estos filtros atenúan las altas frecuencias de la imagen, lo que se traduce visualmente en un efecto de desenfoque. Podemos decir que los filtros de suavizado sirven para:

- Emborronar la imagen:
 - Eliminar pequeños detalles antes de la segmentación de un objeto de interés.
 - Rellenar pequeños espacios.
- Eliminar ruido.

En esta práctica se implementaron filtros de suavizado utilizando máscaras de gran tamaño (7×7 , 9×9 y 11×11), tal como se presenta en el material teórico. A medida que el tamaño del kernel aumenta, el efecto de suavizado es más pronunciado y los contornos de los objetos se vuelven menos definidos.

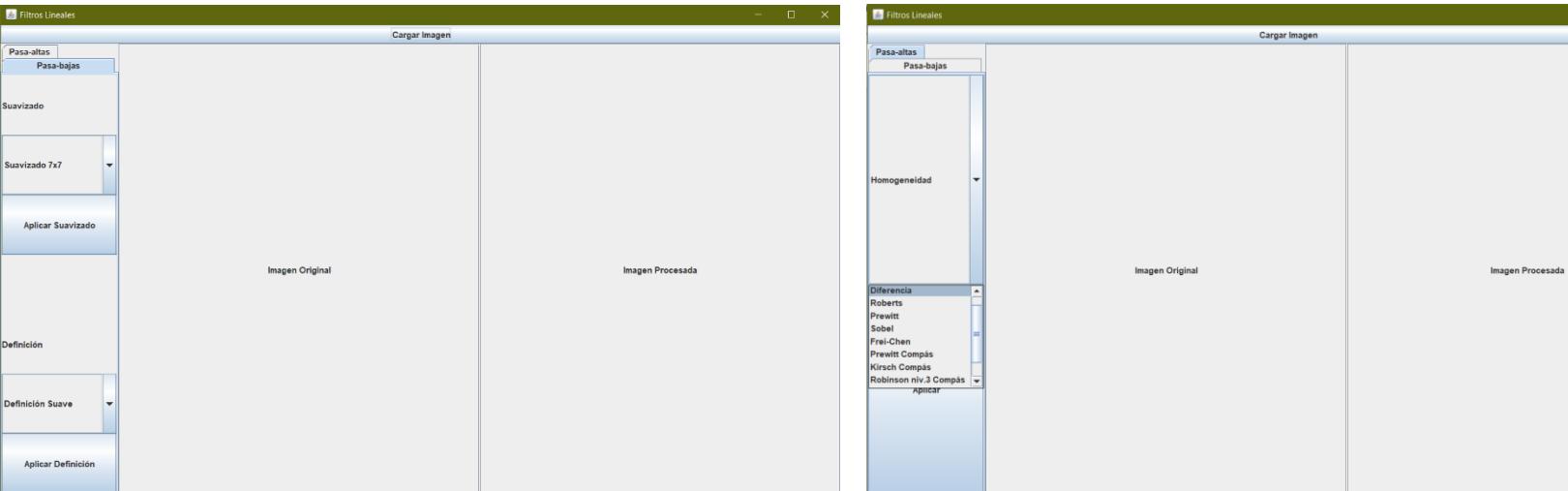


Asimismo, se implementaron filtros de **definición**, los cuales realzan los detalles de la imagen incrementando el contraste entre regiones vecinas. Para ello se emplearon distintas máscaras 3×3 que permiten obtener una definición suave, media o fuerte, dependiendo de la intensidad del realce deseado.



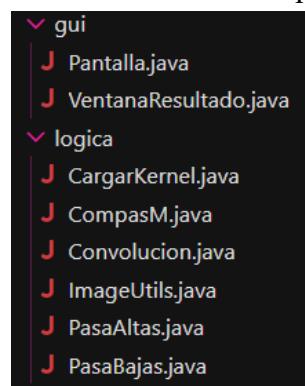
Desarrollo

Para implementar lo requerido en la práctica, se optó por un diseño de la gui de la siguiente forma, separando en distintas pestañas los filtros pasa bajas y pasa altas



La pestaña de pasa bajas es la que se abre al ejecutar el programa. Al seleccionar la pestaña de pasa altas se puede seleccionar de una lista desplegable las posibles optimizaciones a elegir.

Hablando ahora de la estructura del programa, se separa en paquetes la gui y la lógica y dentro de cada uno se tienen clases asociadas a una funcionalidad en particular.



La operación de convolución se implementó como un método que recibe una imagen en escala de grises y una máscara de convolución de tamaño arbitrario. El algoritmo recorre cada píxel de la imagen y calcula el valor resultante a partir de la suma ponderada de los píxeles vecinos definidos por el kernel.

Con el fin de preservar el rango dinámico de la imagen, se incorporó un mecanismo de **normalización automática** de la máscara, el cual divide los coeficientes entre la suma total del kernel cuando esta es distinta de cero. Esta normalización se aplica únicamente a filtros pasa-bajas (particularmente al suavizado), evitando saturación de los valores de intensidad.

Conclusión

A manera de conclusión añado que los filtros de suavizado mostraron claramente la pérdida de detalle conforme aumenta el tamaño de la máscara, mientras que los filtros de definición y detección de bordes facilitaron la identificación de contornos y características importantes de la imagen. Asimismo, la visualización de las respuestas en distintas direcciones y su combinación en una sola imagen me ayudaron a entender mejor el comportamiento de los operadores de gradiente.

Además, al trabajar con diferentes máscaras, pude comprender cómo el tamaño y el tipo de kernel influyen en el resultado final del procesamiento.

Referencias

Awaits. (202d. C., noviembre 9). *Image processing basic: linear filters.* Medium. <https://medium.com/jun94-devpblog/cv-1-image-processing-basic-filter-noise-moving-average-correlation-and-convolution-c026502f6391>

Codificando Bits. (2019, 30 marzo). *La CONVOLUCIÓN en las REDES CONVOLUCIONALES* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=ySbmdeqR0-4>

La Convolución en las Redes Convolucionales | Codificando Bits. (2024, 21 febrero). Codificando Bits. <https://codificandobits.com/blog/convolucion-redes-convolucionales/>

Universidad de Sevilla. (s. f.). *PROCESAMIENTO EN EL DOMINIO ESPACIAL.* https://asignatura.us.es/imagendigital/Tema2-ParteII_Filtros.pdf