**REPORTE DE PRÁCTICA**

**IDENTIFICACIÓN DE LA PRÁCTICA**

| **Práctica** | **7** | **Nombre de la práctica** | | **Filtros lineales** |
| --- | --- | --- | --- | --- |
| **Fecha** | **6/10/2025** | **Nombre del profesor** | | **Alma Nayeli Rodríguez Vázquez** |
| **Nombre del estudiante** | | | **Jesus Alberto Arechiga Carrillo** | |

**OBJETIVO**

| El objetivo de esta práctica consiste en implementar los filtros lineales Box, Gausseano y el de la diferencia. |
| --- |

**PROCEDIMIENTO**

| Realiza la implementación siguiendo estas instrucciones. |
| --- |
| Realiza un programa en Python utilizando OpenCV. Implementa los filtros Box, Gausseano y el de diferencia atendiendo las siguientes indicaciones:   1. Utiliza una imagen a color 2. Implementa un filtro Box de 9x9 utilizando la función cv.blur() de openCV 3. Implementa un filtro Gausseano de 5x5 utilizando la función cv.GaussianBlur() de openCV 4. Implementa el siguiente filtro de diferencia de 5x5 utilizando la función cv.filter2D() de openCV      1. Aplica el filtro de diferencia a la imagen resultante de aplicar el filtro Gausseano 2. Reporta los resultados en los espacios indicados. |

**IMPLEMENTACIÓN**

| Agrega el código de tu implementación aquí. |
| --- |
| **import cv2 as cv**  **import numpy as np**  **import matplotlib.pyplot as plt**  **from google.colab.patches import cv2\_imshow**  **img = cv.imread('lennacolor.png')**  **img\_gray = cv.cvtColor(img, cv.COLOR\_BGR2GRAY)**  **cv2\_imshow(img)**  **cv2\_imshow(img\_gray)**  **# Filtro Box**  **img\_blur = cv.blur(img, ksize=(9,9))**  **cv2\_imshow(img\_blur)**  **# Filtro Gausseano**  **img\_gauss = cv.GaussianBlur(img, ksize=(5,5), sigmaX=0)**  **cv2\_imshow(img\_gauss)**  **# Filtro de Diferencia (Mexican Hat)**  **kernel = np.array([[0, 0, -1, 0, 0],**  **[0, -1, -2, -1, 0],**  **[-1, -2, 16, -2, -1],**  **[0, -1, -2, -1, 0],**  **[0, 0, -1, 0, 0]])**  **img\_hat = cv.filter2D(img, ddepth=-1, kernel=kernel)**  **cv2\_imshow(img\_hat)**  **# Filtro de diferencia a filtro Gausseano**  **img\_diff\_gauss = cv.filter2D(img\_gauss, ddepth=-1, kernel=kernel)**  **cv2\_imshow(img\_diff\_gauss)**  **# Escala de grises con filtro de diferencia 5x5**  **img\_gray\_diff = cv.filter2D(img\_gray, ddepth=-1, kernel=kernel)**  **cv2\_imshow(img\_gray\_diff)**  **# Escala de grises con filtro Gausseano y luego diferencia 5x5**  **img\_gray\_gauss = cv.GaussianBlur(img\_gray, ksize=(5,5), sigmaX=0)**  **img\_gray\_gauss\_diff = cv.filter2D(img\_gray\_gauss, ddepth=-1, kernel=kernel)**  **cv2\_imshow(img\_gray\_gauss\_diff)** |

**RESULTADOS**

Agrega las imágenes indicadas en los espacios correspondientes.

| Imagen RGB | Imagen en escala de grises |
| --- | --- |
|  |  |
| Imagen RGB con filtro Box de 9x9 | Imagen RGB con filtro Gaussiano de 5x5 |
|  |  |
| Imagen RGB con filtro de diferencia de 5x5 | Imagen RGB con filtro Gausseano de 5x5 y después con filtro de diferencia de 5x5 |
|  |  |
| Imagen en escala de grises con filtro de diferencia de 5x5 | Imagen en escala de grises con filtro Gausseano de 5x5 y después con filtro de diferencia de 5x5 |
|  |  |

**CONCLUSIONES**

| Escribe tus observaciones y conclusiones. |
| --- |
| Hay muchos filtros que se le pueden aplicar a las imágenes, pero todos parten de una matriz con valores para hacer el filtro. El filtro Box y Gaussiano son algoritmos ya implementados y el filtro de diferencia es un algoritmo “base” que se implementa con una matriz que debe de ser definida. |