

Guía Laboratorio Practica Final: Procesamiento Digital de Señales

2023-1

Juan Camilo Martinez Ortega

CC 1124855139

Juan.martinez6@udea.edu.co

1. Introducción:

En esta práctica, exploraremos el uso de filtros FIR en imágenes para mejorar su calidad y resaltar aspectos específicos. El objetivo principal es que los estudiantes aprendan a utilizar diferentes tipos de filtros y aplicarlos a imágenes.

1.1 Carga de la imagen y visualización:

Cargue la imagen proporcionada, normalícela y muestre la versión original. Asegúrese de graficar la imagen para su visualización.

2. Análisis espectral:

Realice el análisis espectral de la imagen. Grafique el espectro de amplitud, el espectro de fase y el espectro de magnitud. Explique las características particulares presentes en cada espectro.

3. Diseño de un filtro pasa bajos:

En esta sección, diseñaremos un filtro FIR para resaltar o atenuar ciertas frecuencias en la imagen.

3.1 Diseño del filtro pasa bajos:

Diseñe un filtro pasa bajos y gráfiquelo. El filtro pasa-bajos debe atenuar las altas frecuencias de la imagen y preservar las bajas frecuencias.

3.2 Aplicación del filtro:

Aplique el filtro diseñado a la imagen para obtener la versión filtrada.

3.3 Análisis espectral de la imagen filtrada:

Realice el análisis espectral de la imagen filtrada. Grafique el espectro de amplitud, el espectro de fase y el espectro de magnitud.

Filtros lineales:

Los filtros lineales son aquellos que aplican una operación matemática a cada píxel de la imagen utilizando una máscara o kernel. En esta sección, exploraremos dos filtros lineales comunes: el filtro Gaussiano y el filtro Sobel.

4.1 Carga de la imagen:

Cargue la imagen en Python utilizando una biblioteca como OpenCV o Pillow, y luego muestre la imagen.

4.1.1 Filtro Gaussiano:

Diseñe y aplique un filtro Gaussiano con un kernel de tamaño (5, 5) y desviación estándar de 0. Muestre la imagen resultante y experimente cambiando los valores del kernel para observar las diferencias en las imágenes resultantes.

El filtro Gaussiano se utiliza para suavizar la imagen y reducir el ruido

4.1.2 Filtro Sobel:

Diseñe y aplique un filtro Sobel para detectar bordes en la imagen. Siga los siguientes pasos:

4.1 Carga de la imagen:

Cargue la imagen en Python utilizando una biblioteca como OpenCV o Pillow.

4.2 Aplicación del filtro Sobel:

Aplique el filtro Sobel utilizando una máscara o kernel específico para resaltar los bordes en la imagen.