

Procesamiento Digital de Imágenes

Práctica 5, “Filtrado Espacial”

Prácticas de Laboratorio

1. Objetivo.

Que el alumno aplique las técnicas de filtrado espacial para acentuar o disminuir ciertas características de las imágenes digitales.

2. Introducción

El filtrado espacial permite modificar la contribución de ciertos rangos de frecuencias de la imagen para lograrlo se utiliza la información de los vecinos permitiendo su implementación usando la operación de convolución. La idea básica de la convolución es que una ventana de algún tamaño finito y forma es desplazada a través de la imagen. El valor del pixel de salida es la suma ponderada de los pixeles de entrada dentro de la ventana, donde los pesos son los valores del filtro asignado a cada pixel de la ventana. La ventana con sus pesos es llamado *kernel de convolución, mascara o filtro*.

3. Desarrollo

A) Filtros suavizadores

Parte 1: Filtro promedio

- 1.- Define un filtro promedio tipo bloque (normalizado) de 3x3, 5x5, 7x7 y 11x11 elementos.
- 2.- Agrega ruido del tipo Gaussiano ($M=0$, $\sigma^2=0.02$) a la imagen LENA.BMP (<https://la.mathworks.com/help/images/ref/imnoise.html>)
- 3.- Realiza la operación de convolución con las imágenes sin ruido y con ruido respectivamente usando cada filtro $h_{promedio}$ (usa el comando **conv2** y la opción 'same')
- 4.- Visualiza las imágenes resultantes.

Parte 2: Filtro Gaussiano

- 1.- Define un filtro Gaussiano de 3x3, 5x5, 7x7 y 11x11 elementos. Para ello, elabora un script que obtenga los coeficientes binomiales de orden 0 necesarios para su construcción:

$$f_N^r(x) = \nabla^r \binom{N-r}{x}$$

donde

$$\nabla f(x) = f(x) - f(x-1)$$

para $r=0$

$$f_N(x) = \binom{N}{x} = \frac{N!}{x!(N-x)!}$$

Los filtros binomiales en 2D se obtienen como:

$$[f_N(x)]^T \times [f_N(x)]$$

- 2.- Agrega ruido del tipo Gaussiano ($M=0$, $\sigma^2=0.02$) a la imagen LENA.BMP.
- 3.- Realiza la operación de convolución con las imágenes sin ruido y con ruido respectivamente usando cada filtro h_G (usa el comando **conv2** y la opción 'same')
- 4.- Visualiza las imágenes resultantes.

Compara los resultados obtenidos del filtro promediador tipo bloque vs el filtro Gaussiano. Da tus conclusiones al respecto.

Parte 3: Emborronamiento de imágenes para ocultar contenido en imágenes digitales

- 1.- Tomando como base una imagen cualquiera, con apoyo de un filtro paso bajas y procesamientos de imagen adicionales, diseña un algoritmo que procese las secciones del fondo de una imagen, de tal forma que la imagen final contenga el fondo difuminado, ejemplo:



El tamaño de la ventana del filtro, así como la elección de los procesamientos de imagen utilizados quedan a criterio del estudiante, con el objetivo de lograr un resultado lo más semejante posible al del ejemplo.

B) Mejoramiento de la Nitidez (Mascara “unsharp”)

- 1.- Leer la imagen “boatsc.bmp”
- 2.- Difuminar la imagen del punto anterior usando un filtro pasó bajas (box filter) de ventana 5x5.
- 3.- Obtener 4 imágenes con mejoramiento de nitidez haciendo uso del proceso *unsharp masking* “high-boost” con los siguientes tipos de filtro pasa bajas:

- a) Filtro paso bajas de bloque con tamaño de ventana 3x3 y 7x7.
- b) Filtro paso bajas binomial (orden 0) de ventana 3x3 y 7x7.

Considerando:

$$g_{mask}(x, y) = f(x, y) - \bar{f}(x, y) \quad (1)$$

$$g(x, y) = f(x, y) + k \times g_{mask}(x, y) \quad (2)$$

$\bar{f}(x, y)$ es la versión suavizada de una imagen $f(x, y)$ y k es un peso.

Explicar cuál filtrado proporcione los mejores resultados para mejoramiento de la nitidez.

C) Aclaramiento de imágenes usando el operador Laplaciano

- 1.- Leer la imagen “lenac.bmp”
- 2.- Emborronar la imagen del punto anterior usando un filtro pasa-bajas de ventana 5x5.
- 3.- Obtener 6 imágenes aclaradas usando el procedimiento de aclaramiento de imágenes usando los siguientes operadores Laplacianos:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Considerando:

$$g(x, y) = f(x, y) - c \nabla^2 f(x, y) \quad (3)$$

Donde $c = 0.3, 0.5$ y 0.7 respectivamente.

Explicar cuál operador proporcione los mejores resultados para aclaramiento de imagen.

D) Aplicación del gradiente para aclaración de imágenes

Dada la siguiente definición:

$$\hat{f}(x, y) = f(x, y) + k \times (g_x(x, y) + g_y(x, y)) \quad (4)$$

Donde k es un factor que define el grado de aclaración en la imagen. Cuando $k \geq 1$ se produce una distorsión visible en la imagen.

- 1.- Leer la imagen “airplane16c.bmp”
- 2.- Emborronar la imagen del punto anterior usando un filtro pasa-bajas de ventana 5x5.
- 3.- Utilizar el gradiente normal, el operador Robert y el operador Sobel para obtener las versiones aclaradas, respectivamente.

E) Filtrado No Lineal

Filtros de orden estadístico (no lineales)

El filtrado de orden estadístico es un filtrado espacial no-lineal, cuya respuesta está basada en el ordenamiento de los píxeles dentro de la ventana. El resultado basado del ordenamiento se reemplaza con el valor del centro de la ventana. Existen varios filtros que pertenecen a este grupo de filtros. El ejemplo más conocido es el **filtro de mediana**, que reemplaza el pixel central por la mediana de los pixeles contenidos en la vecindad del filtro. Los filtros de mediana eliminan cierto tipo de ruido de manera muy eficiente, el llamado **ruido impulso o sal y pimienta**.

A. Filtro de Mediana (Median Filter).- Este filtro reemplaza el valor del centro con el valor mediano de los píxeles dentro la ventana.

$$f(x, y) = \underset{(s,t) \in V}{median} \{f(s, t)\}$$

- 1.- Leer la imagen “airplane16.bmp” y contaminar su contenido con ruido del tipo “sal-pimienta” con una densidad de 0.1, <https://la.mathworks.com/help/images/ref/imnoise.html>
- 2.- Aplicar un filtro de mediana de ventanas 3x3, 5x5 y 7x7 a la imagen contaminada

3.- Desplegar cada imagen filtrada

Explica cual tamaño de ventana proporciona el mejor resultado.

4. Resultados

5. Código

En esta sección deberán presentar el código fuente del programa en MATLAB (o en la herramienta que hayan utilizado en su defecto).

6. Conclusiones

Referencias

[1]

[2]