

Universidad Nacional Autónoma de México

FACULTAD DE CIENCIAS

PROCESO DIGITAL DE IMÁGENES

*Práctica 05*  
*Restauración de la imagen*

AUTOR:

*Delgado Díaz Hermes Alberto*  
*319258613*



19 de noviembre de 2025

## 1. Objetivos

- Implementar el filtro promedio aritmético, geométrico y adaptativo. Comparar su desempeño en presencia de ruido.
- Implementar el filtro mediana adaptativo y comparar su desempeño comparado con el filtro mediana simple, en presencia de ruido.
- De acuerdo al modelo de degradación que sufre una serie de imágenes, encontrar y aplicar el filtro de Wiener adecuado para la restauración óptima de cada imagen.

## 2. Introducción

Como en el realce de imágenes, la meta final de las técnicas de restauración es mejorar la imagen en un sentido predeterminado. A pesar de que existen áreas de sobrelape, el realce de una imagen es un proceso altamente subjetivo, mientras que la restauración de una imagen es parte de un proceso objetivo.

La restauración intenta reconstruir o recobrar una imagen que ha sido degradada utilizando conocimiento a priori del modelo de degradación y aplicando el proceso inverso a éste para poder recobrar así la imagen original.

Si  $H$  es lineal y es un proceso invariante a la posición, entonces la imagen degradada está dada, en el dominio espacial por:

$$g(x, y) = h(x, y) * f(x, y) + \eta(x, y)$$

donde  $h(x, y)$  es la representación espacial de la función de degradación, el símbolo " \* " indica convolución y  $\eta(x, y)$  es ruido aditivo.

- Restauración en presencia de ruido (filtros espaciales).

Las fuentes principales de ruido en las imágenes digitales son durante la adquisición (digitalización) y/o durante la transmisión. El desempeño de los sensores de imágenes es afectado por una variedad de factores, como son las condiciones ambientales durante la adquisición de la imagen y por la calidad de los elementos de sensor.

Cuando la única fuente de degradación en una imagen es ruido, la ecuación anterior se reescribe:

$$g(x, y) = f(x, y) + \eta(x, y)$$

Existen diversos filtros útiles para eliminar el ruido:

1. El **filtro promedio aritmético** calcula el valor promedio de la imagen corrupta  $g(x, y)$  en el área  $S_{xy}$ . El valor de la imagen restaurada en el punto  $(x, y)$  es simplemente el

promedio aritmético calculado en esa vecindad:

$$\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s, t)$$

2. Una imagen restaurada utilizando un **filtro promedio geométrico** está dada por la expresión:

$$\hat{f}(x, y) = \left[ \prod_{(s,t) \in S_{xy}} g(s, t) \right]^{\frac{1}{mn}}$$

En este caso, cada pixel restaurado está dado por el producto de los pixeles en la subimagen (ventana), elevado a la potencia  $\frac{1}{mn}$ .

3. **Filtro adaptativo** cuyo comportamiento cambia según las características de la imagen dentro de la región del filtro definida por una ventana rectangular  $S_{xy}$  de tamaño  $m \times n$ . Una expresión adaptativa para obtener  $\hat{f}(x, y)$  basada en los supuestos anteriores, puede escribirse como:

$$\hat{f}(x, y) = g(x, y) - \frac{\sigma_\eta^2}{\sigma_L^2} [g(x, y) - m_L]$$

donde  $\sigma_\eta^2$  es la varianza del ruido que corrompe a  $f(x, y)$  para formar  $g(x, y)$ , y  $m_L$  y  $\sigma_L^2$  son la media y la varianza locales de los pixeles en la vecindad  $S_{xy}$ , respectivamente.

La única cantidad que se necesita conocer o estimar es la varianza del ruido general,  $\sigma_\eta^2$ . Los otros parámetros se calculan de los pixeles en  $S_{xy}$  en cada posición  $(x, y)$  en donde el filtro está centrado.

- El **filtro mediana adaptativo** también trabaja en una vecindad  $S_{xy}$ . Sin embargo, a diferencia de los demás filtros, el filtro mediana adaptativo cambia (incrementa) el tamaño de  $S_{xy}$  durante su operación, dependiendo en ciertas condiciones que veremos más adelante. Recuerde que la salida del filtro es un sólo valor que se utiliza para reemplazar el valor del pixel en la posición  $(x, y)$ , del punto central particular de la vecindad  $S_{xy}$ . El algoritmo del filtro mediana adaptativo trabaja en dos niveles, denominados nivel A y nivel B como sigue:

**Nivel A:**  $A1 = z_{med} - z_{min}$

$$A2 = z_{med} - z_{max}$$

Si  $A1 > 0$  AND  $A2 < 0$ , ve al nivel B

si no incrementa el tamaño de la ventana

Si el tamaño de la ventana  $\leq S_{max}$  repita el nivel A

si no salida =  $z_{xy}$

**Nivel B:**  $B1 = z_{xy} - z_{min}$

$$B2 = z_{xy} - z_{max}$$

Si  $B1 > 0$  AND  $B2 < 0$ , salida =  $z_{xy}$

si no salida =  $z_{med}$

donde  $z_{min}$ ,  $z_{max}$  y  $z_{med}$  es valor mínimo, máximo y mediana de los niveles de gris de  $S_{xy}$ , respectivamente.  $z_{xy}$  es el nivel de gris en las coordenadas  $(x, y)$  y  $S_{max}$  es el valor de tamaño máximo permitido para la ventana  $S_{xy}$ .

- **Filtro Wiener.** Un método que incorpora ambos, la función de degradación y las características estadísticas del ruido, en el proceso de restauración es el llamado filtro Wiener. El método consiste en considerar imagen y ruido como un proceso aleatorio, y el objetivo es encontrar un estimador  $\hat{f}$  de la imagen no-corrupta  $f$  de tal manera que el error promedio al cuadrado entre ellas sea mínimo. Este error está dado por:

$$e^2 = E\{(f - \hat{f})^2\}$$

donde  $E\{\cdot\}$  es el valor esperado (la esperanza) del argumento.

Se asume que el ruido y la imagen no están correlacionados, que una o la otra tienen media igual a cero, y que los niveles de gris en la estimación son una función lineal de los niveles de la imagen degradada. Basados en estas condiciones, la función de error mínima está dada en el dominio de la frecuencia por:

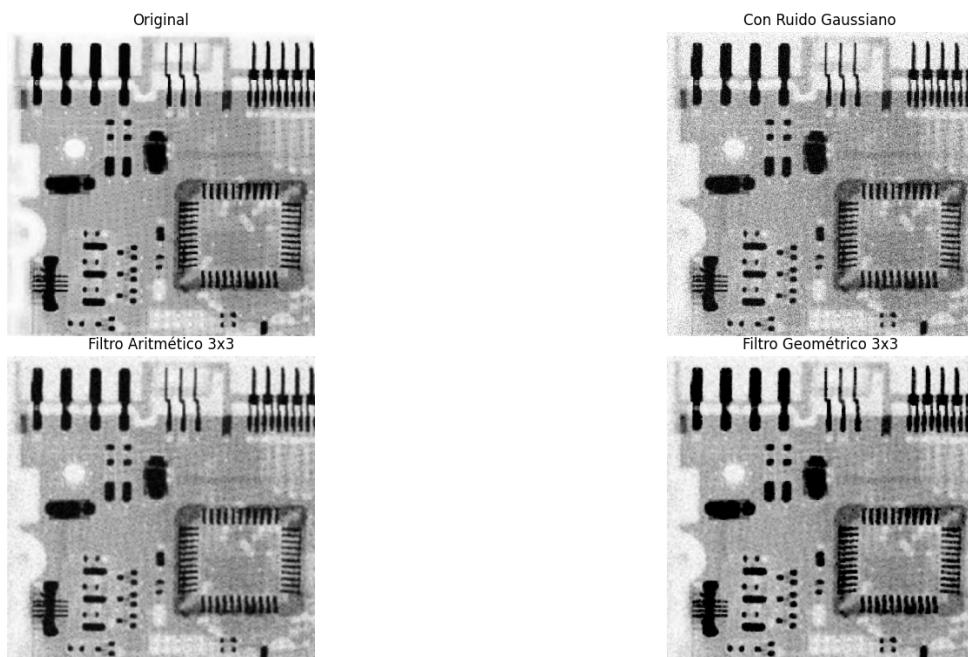
$$\hat{F}(u, v) = \left[ \frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + S_\eta(u, v)/S_f(u, v)} \right] G(u, v)$$

donde  $H(u, v)$  = función de degradación,  $H * (u, v)$  = conjugado complejo de  $H(u, v)$ .  $|H(u, v)|^2 = H * (u, v)H(u, v)$ .  $S_\eta(u, v) = |N(u, v)|^2$  = espectro de potencia del ruido.  $S_f(u, v) = |F(u, v)|^2$  = espectro de potencia de la imagen no degradada.

El filtro que consiste en los términos dentro de los corchetes, también se conoce como **filtro de error promedio mínimo al cuadrado**, o **filtro de error de mínimos cuadrados**.

### 3. Desarrollo

- Utiliza la imagen del **circuito**, genera ruido gaussiano aditivo con media cero y desviación estándar de 0,04. Filtra la imagen primero con un filtro promedio aritmético de tamaño  $3 \times 3$ , y luego filtrala con un filtro promedio geométrico del mismo tamaño. Comparalos.

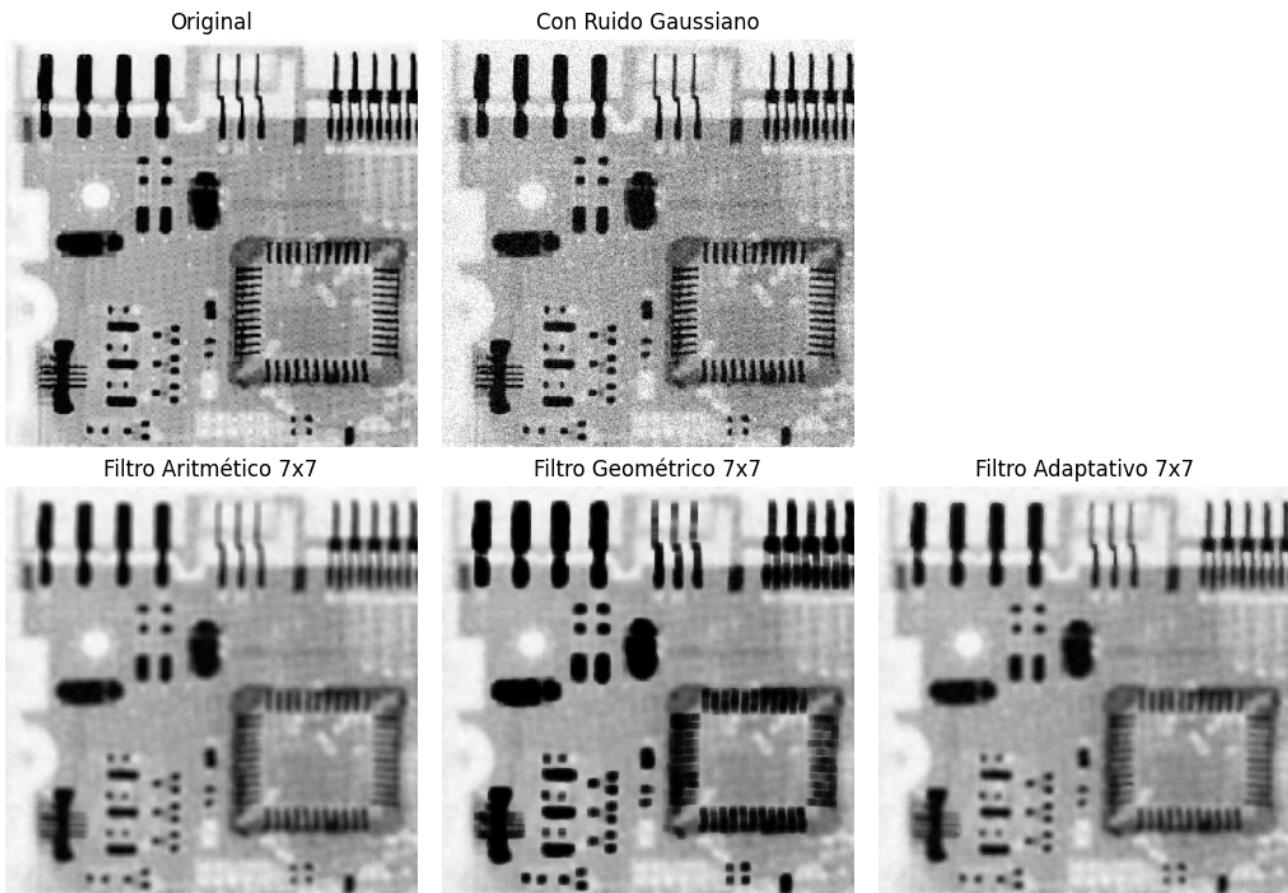


Se genera ruido Gaussiano aditivo (media 0, desviación 0,04) a la imagen de circuito. Se aplicaron ambos filtros de  $3 \times 3$ .

#### Análisis:

- Ambos filtros logran reducir el ruido Gaussiano.
- El **Filtro Aritmético** produce un resultado con un poco de presencia del ruido.
- El Filtro Geométrico preserva mejor los detalles y bordes de la imagen en comparación con el aritmético, ofreciendo un suavizado comparable pero con menor pérdida de detalle.

2. Utiliza nuevamente la imagen **circuito**, genera ruido gaussiano aditivo de media cero y desviación estandar de 0,04. Filtrala primero con un filtro promedio aritmético de tamaño  $7 \times 7$ . Filtrala ahora con un filtro geométrico del mismo tamaño. Finalmente filtrala con un filtro adaptativo para reducción de ruido del mismo tamaño que los anteriores. Comparalos.

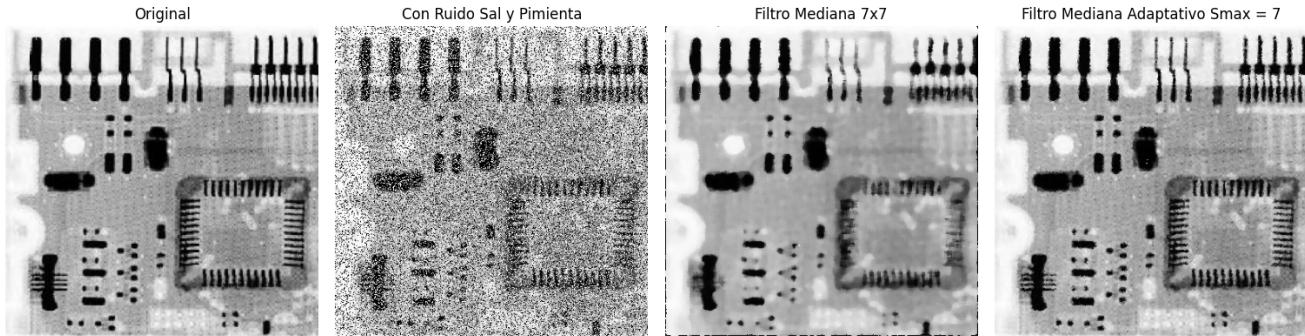


Se repitió el proceso con ruido Gaussiano, pero usando un tamaño de ventana de  $7 \times 7$  y añadiendo el filtro adaptativo de reducción local de ruido.

#### Análisis:

- Al aumentar el tamaño de la ventana a  $7 \times 7$ , el emborronamiento en los filtros aritmético y geométrico es mucho más severo.
- El Filtro Adaptativo 9999 muestra un desempeño superior. Como su comportamiento cambia según las características locales (media y varianza) 10101010, logra preservar los bordes (donde la varianza local  $\sigma_L^2$  es alta 1111) y suaviza el ruido en regiones planas (donde  $\sigma_L^2$  es similar a  $\sigma_\eta^2$  12).

3. Utiliza la imagen **circuito**, generale un ruido sal y pimienta aditivo con probabilidades  $P_a = P_b = 0,25$ . Filtrala primero con un filtro mediana de tamaño  $7 \times 7$ . Filtrala ahora con un filtro mediana adaptativo con  $S_{max} = 7$ . Comparalos.



Se utilizó la imagen del circuito con ruido impulsivo "sal y pimienta" de alta densidad ( $P_a = P_b = 0,25$ ). Se comparó el filtro mediana simple de  $7 \times 7$  con el filtro mediana adaptativo con  $S_{max} = 7$

### Análisis

- El Filtro Mediana simple es efectivo para ruido impulsivo, pero con una densidad tan alta (50 % de pixeles corruptos), no logra eliminar todo el ruido y, a la vez, introduce un emborronamiento considerable en los bordes.
- El Filtro Mediana Adaptativo es notablemente superior. Su algoritmo de dos niveles (A y B) le permite:
  1. Determinar si el valor de la mediana ( $z_{med}$ ) es un impulso. Si lo es, incrementa el tamaño de la ventana (Nivel A).
  2. Una vez que encuentra una mediana no-impulso, revisa si el pixel central ( $z_{xy}$ ) es un impulso (Nivel B).
  3. **Resultado:** Solo reemplaza los pixeles que identifica como ruido impulsivo, preservando el detalle en los pixeles no corruptos. Esto evita la distorsión y el embotellamiento excesivo que sufre el filtro mediana simple, tal como se describe en la teoría.

4. Para la imagen **lena** con ruido aditivo de tipo gaussiano, encontrar el filtro de Wiener y restaurar la imagen.



Caso 1. Filtro Wiener a imagen con ruido

Se aplicó ruido Gaussiano aditivo ( $\sigma = 25$ ) a la imagen original y se restauró utilizando el Filtro de Wiener correspondiente al Caso 1, utilizando el espectro de potencia de la imagen original y del ruido estimado.

#### Análisis:

- El filtro de Wiener logra una restauración superior a los filtros espaciales simples (como el aritmético) porque minimiza el error cuadrático medio globalmente.
- Se observa que el "grano" del ruido se ha suavizado considerablemente en la imagen restaurada, recuperando texturas de la piel y el sombrero de Lena que el ruido había ocultado.
- A diferencia de un filtro paso bajas simple, el filtro de Wiener preserva mejor los bordes, ya que pondera las frecuencias altas basándose en la relación Señal-Ruido ( $S_{ff}/(S_{ff} + S_{nn})$ ).

5. Encontrar el filtro de Wiener y restaurar una imagen **lena** que ha sido sometida a un proceso de pérdida de nitidez. La imagen con pérdida de nitidez se obtiene filtrando una imagen nítida y libre de ruido con un filtro paso bajas de tamaño  $9 \times 9$  normalizado (filtro promedio ponderado).



Caso 2. Filtro Wiener a imagen con pérdida de nitidez

La imagen fue degradada únicamente por un filtro promedio ponderado de  $9 \times 9$  (desenfoque) sin ruido. Se aplicó el filtro inverso ( $\approx 1/H$ ).

#### Análisis:

- Al no existir ruido aditivo, la restauración es casi perfecta. El filtro inverso logró deshacer matemáticamente la convolución del filtro promedio ponderado.
- Los bordes del sombrero y el cabello, que estaban difusos en la imagen degradada, recuperan su nitidez original.
- Este es un caso ideal; la presencia del factor  $\epsilon$  en el código evitó la división por cero en las frecuencias donde  $H(u, v)$  se anula, permitiendo la recuperación de la imagen sin generar artefactos numéricos graves.

6. Para una imagen **lena** a la que se le ha agregado ruido tipo gaussiano y posteriormente ha perdido nitidez, encontrar el filtro de Wiener y restaurarla. Para obtener esta imagen degradada, primero se agrega el ruido tipo gaussiano a la imagen original y luego se filtra con el mismo filtro pasos bajas descrito en el punto 5.



Caso 3. Filtro Wiener a imagen con ruido + pérdida de nitidez

Se añadió ruido Gaussiano a la imagen y posteriormente se aplicó el desenfoque. El ruido también fue "borroneado" por el filtro.

**Análisis:**

- Este escenario presenta un reto particular porque el ruido cambia sus propiedades estadísticas al pasar por el filtro de desenfoque (se vuelve ruido correlacionado).
- La imagen restaurada muestra una recuperación exitosa de la estructura general. Aunque es difícil recuperar las frecuencias más altas (detalles finos) debido a que el ruido y la señal se mezclaron antes del desenfoque, el filtro logró estabilizar la imagen mejor que si se hubieran tratado los defectos por separado.

7. Encontrar el filtro de Wiener y restaurar una imagen **lena** que ha sido degradada por pérdida de nitidez y posteriormente se le ha agregado ruido de tipo gaussiano. Para obtener esta imagen degradada se utiliza el filtro pasos bajas descrito en el punto 5 y posteriormente se le agrega ruido.



Caso 4. Filtro Wiener a imagen con pérdida de nitidez + ruido

Este es el modelo más realista de degradación: la imagen sufrió pérdida de nitidez (desenfoque) y luego se le añadió ruido en el sensor. Se usó la fórmula completa de Wiener con el conjugado  $H^*$ .

#### Análisis:

- La imagen degradada es la más difícil de interpretar visualmente, ya que combina bordes borrosos con grano de ruido.
- El filtro de Wiener demostró su robustez aquí. La fórmula balancea dos objetivos contradictorios: invertir el desenfoque (que requiere amplificar frecuencias altas) y suprimir el ruido (que requiere atenuar frecuencias altas).
- El resultado es una imagen con bordes definidos y un nivel de ruido controlado, demostrando por qué este es el filtro estándar para restauración óptima lineal.

## 4. Conclusión

En esta práctica se implementaron y compararon diversos filtros de restauración. En el dominio espacial, se evidenció que para el ruido impulsivo, los filtros lineales convencionales son insuficientes, siendo el filtro de mediana adaptativo la solución superior al lograr eliminar el ruido sin degradar los bordes. Para el ruido Gaussiano, el filtro geométrico y el adaptativo local mostraron un mejor balance entre suavizado y preservación de detalles que el promedio aritmético.

En el dominio de la frecuencia, se comprobó que el Filtro de Wiener es una herramienta potente para revertir degradaciones complejas, como la combinación de desenfoque y ruido (Casos 3 y 4), siempre que se eviten las singularidades matemáticas mediante términos de estabilización en el denominador.

Finalmente, es crucial notar que en estos ejercicios obtuvimos restauraciones casi perfectas porque utilizamos la imagen original para calcular matemáticamente la función de degradación exacta ( $H = G/F$ ). En un escenario real, carecemos de la imagen original, lo que convierte el proceso en un problema de "deconvolución ciega". Sin la imagen de referencia, debemos estimar o adivinar la función de degradación y la relación señal-ruido; si estas estimaciones no son precisas, el filtro no podrá distinguir correctamente entre detalle y ruido, generando artefactos visuales o amplificando la degradación en lugar de corregirla.

## Referencias

- [1] Abhishek Jain, "Salt and pepper noise and how to remove it in machine learning", <https://medium.com/@abhishekjainindore24/salt-and-pepper-noise-and-how-to-remove-it-in-machine-learning-032d76b138a5>
- [2] Ms Somanna, "Guide to adding noise to your data using Python and Numpy", [https://medium.com/@ms\\_somanna/guide-to-adding-noise-to-your-data-using-python-and-numpy-c8be815df524](https://medium.com/@ms_somanna/guide-to-adding-noise-to-your-data-using-python-and-numpy-c8be815df524)
- [3] eastWillow, "Fourier Transform", [https://opencv24-python-tutorials.readthedocs.io/en/latest/py\\_tutorials/py\\_imgproc/py\\_transforms/py\\_fourier\\_transform/py\\_fourier\\_transform.html](https://opencv24-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_transforms/py_fourier_transform/py_fourier_transform.html)
- [4] Stephen Gruppetta, "How to Create Any Image Using Only Sine Functions | 2D Fourier Transform in Python", <https://thepythontutorialbook.com/2021/08/30/2d-fourier-transform-in-python-and-fourier-synthesis-of-images/>