

## CC3182 – Visión por Computadora

# Laboratorio 2

### Instrucciones

- Esta es una actividad en grupos de no más de 3 integrantes.
  - Recuerden **unirse al grupo de canvas**
- No se permitirá ni se aceptará cualquier indicio de copia. De presentarse, se procederá según el reglamento correspondiente.
- Tendrán hasta el día indicado en Canvas.
  - No se confíen, aprovechen el tiempo en clase para entender todos los ejercicios y avanzar lo más posible.
- **NOTA:** Limiten el uso de IA generativa. Intenten primero buscar en fuentes de internet y si en verdad necesitan usarla, asegúrense de colcar el prompt que utilizar para cada task donde corresponda, así como una explicación de por qué ese prompt funcionó.

### Consideraciones Generales

- El lenguaje sugerido para este laboratorio es Python, con las librerías de OpenCV y Numpy
- Se debe entregar tanto Jupyter Notebook (.ipynb) como la versión PDF del mismo, en este debe contener el código, las visualizaciones y principalmente el análisis escrito justificando cada decisión
- Usted es responsable de importar las librerías, leer las imágenes y estructurar su solución
- Utilice el de de imágenes provisto.

### Task 1

Usted trabaja para una empresa de imágenes satelitales. Una de las cámaras en órbita tiene una interferencia electrónica que genera un ruido sinusoidal (patrón de rayas diagonales) sobre las fotografías de la superficie terrestre. Los filtros espaciales tradicionales (Gaussian Blur) destruyen los detalles geográficos necesarios. Por ello usted ha decidido idear una solución tomando como base fotografías que tiene a mano con el mismo problema previo a implementar la solución real. Con esto en mente, realice:

1. Cargue la imagen `periodic_noise.jpg` en escala de grises.
2. Calcule la Transformada Discreta de Fourier (DFT) y desplace el componente de frecuencia cero al centro.
3. Muestre el Espectro de Magnitud en escala logarítmica.
4. Identifique visualmente los "picos" de energía que no corresponden a la información natural de la imagen (puntos brillantes fuera del centro).
5. Cree una máscara (Notch Filter) que bloquee específicamente esas frecuencias parásitas (haciéndolas cero), pero preserve el resto del espectro, incluyendo el componente DC.
6. Aplique la Transformada Inversa (IDFT) para recuperar la imagen espacial.

Para esta parte se espera que entregue:

- Código funcional y visualización correcta del espectro con los picos de ruidos señalados
- Imagen restaurada exitosamente (sin rayas y con detalles nítidos)
- **Explique** por qué un filtro de promedio (average filter) de 5x5 en el dominio espacial hubiera sido una mala solución para este problema específico.

## Task 2

Está desarrollando un sistema biométrico de seguridad. El sensor de huellas dactilares está sucio y produce imágenes binarias con dos tipos de defectos:

1. Pequeños puntos blancos en los valles negros de la huella (Ruido Sal)
2. Las “crestas” de la huella tienen pequeñas roturas que impiden que el algoritmo de matching funcione (grietas).

Con esto en mente realice:

1. Cargue la imagen fingerprint\_noisy.png. Asegúrese de que sea binaria.
2. Aplique una operación morfológica para eliminar el ruido blanco sin destruir las crestas de la huella. (Seleccione entre Erosión, Dilatación, Apertura o Cierre).
3. Aplique una segunda operación secuencial para conectar las grietas en las crestas de la huella.
4. Muestre la imagen original, la imagen tras el paso 2, y la imagen final.

Para esta parte se espera que su entregable muestre:

- Selección correcta de los Elementos Estructurantes (Forma y Tamaño) para cada paso.
- Calidad visual de la imagen final (sin ruido y continua)
- **Responda:** ¿El orden de los factores altera el producto? Explique qué hubiera pasado si hubiera aplicado las operaciones en orden inverso y demuéstrela con un ejemplo visual en el notebook.


## Task 3

Una fábrica textil necesita detectar rasgaduras en telas de mezclilla (denim) automáticamente. El problema es que la tela tiene una textura natural fuerte (patrón repetitivo) que confunde a los detectores de bordes simples (Canny), detectando el tejido como si fuera un defecto. Por ello se le pide que usted diseñe un pipeline híbrido que combine Fourier y Morfología para aislar solamente la rasgadura. Para ello comienza por probar su solución en una imagen que tiene a mano. Con esto en mente, realice:

1. Utilice Fourier para analizar la textura repetitiva de la tela. Diseñe un filtro que elimine las frecuencias altas/repetitivas del tejido, dejando una imagen "suavizada" donde solo resalte la anomalía (la rasgadura) y la iluminación global. (Supresión de Textura)
  - a. Hint: ¿Qué pasa si eliminamos las frecuencias altas periféricas o específicas?
2. Aplique un umbralizado (thresholding) a la imagen resultante del paso 1 para obtener una máscara binaria preliminar. (Segmentación)
3. La máscara seguramente tendrá ruido residual. Utilice operaciones morfológicas para limpiar la máscara y dejar únicamente la silueta de la rasgadura. (Refinamiento)

Para esta parte se espera que su entregable muestre

- Implementación del Pipeline completo (FFT -> Filtro -> IFFT -> Threshold -> Morfología).
- Resultado final: Una máscara binaria donde el fondo sea negro y la rasgadura sea blanca, sin falsos positivos de la textura de la tela.
- **Escriba** un párrafo (aprox. 100 palabras) describiendo los trade-offs de su solución. ¿Qué pasa si la rasgadura es muy pequeña? ¿Qué pasa si cambiamos el tipo de tela? ¿Es su solución robusta o específica para esta imagen?



## Entregas en Canvas

1. Documento PDF con las respuestas a cada task
2. Archivo .ipynb, o link a repositorio de GitHub (No se acepta entregas en otros medios)

## Evaluación

1. [1.5 pt] Task 1
2. [1.5 pt] Task 2
3. [2.0 pt] Task 3

Total 5 pts