

Tarea 1

Instrucciones generales

- La tarea se realiza en grupos de máximo 3 personas. Cada grupo debe escribir el nombre de los integrantes del grupo en la siguiente dirección electrónica:

https://docs.google.com/spreadsheets/d/1e-biLZ4gG0G5v_TioklGjgjlnoKavsA2

El número del grupo está indicado en la primera columna del documento.

- Todos los archivos de esta tarea se encuentran en la carpeta de *One Drive* del curso.
- Los archivos computacionales implementados en GNU Octave y Python deben estar correctamente comentados. Por cada archivo que no este documentado correctamente, se restaran 5 puntos de la nota final. **Si alguna función o archivo computacional está incompleto o genera error al momento de compilar, entonces pierde el 75% del puntaje de la pregunta asignada.**
- El documento escrito para cada parte de la tarea debe tener su respectiva bibliografía. **En caso de no presentar la bibliografía, se le restará 5 puntos por cada documento que no contenga referencias bibliográficas.**
- Los archivos que dan solución a la tarea deben estar en una carpeta principal con nombre **Tarea 1 - Grupo #**, donde # es el número de cada grupo. Dentro de esta carpeta debe existir tres carpetas con nombres **Parte 1**, **Parte 2** y **Parte 3**. En cada una de estas carpetas estarán todos los archivos necesarios para el desarrollo de las preguntas mencionadas anteriormente.
- La solución de la tarea que se encuentra en la carpeta **Tarea 1 - Grupo #** debe comprimirse en un archivo **.zip** y subirlo al formulario que se encuentra en el siguiente enlace:

<https://forms.gle/ufnZ6dGWXCniv5T78>

Observaciones:

- Se necesita tener una cuenta de **gmail** para llenar el formulario.

- **Fecha y hora máxima de entrega: Domingo 25 de Agosto del 2023, a las 11:59 pm**
- Las entregas tardías se realizarán al correo penalizarán con una reducción de la nota obtenida con un 10% por cada hora de atraso. A las tareas que excedan el plazo de entrega en 10 horas o más después de la hora límite, se les asignará la nota de 0. **SOLAMENTE las entregas tardías se realizarán al correo jusoto@tec.ac.cr.**

Parte 1: Filtro de la Mediana

Descripción General

- Esta parte de la tarea consiste en el estudio de un método para eliminar el ruido de imágenes y videos, llamado *filtro de la mediana*.
- La implementación computacional se debe realizar en Python.
- La parte escrita debe estar en un documento con nombre **T1.P1.pdf**. Todos los resultados numéricos e imágenes deben aparecer en el documento escrito, y explicado con detalle y claridad.

Preguntas

- **Pregunta 1:** Investigue e implemente computacionalmente el **filtro de la mediana** para eliminar el ruido de una imagen.
 - a) [Valor 10 puntos]: La parte escrita debe incluir una breve explicación del filtro de la mediana, su formulación matemática y su respectivo pseudocódigo.
 - b) [Valor 15 puntos]: La sintaxis de la función implementada debe ser **$Y = \text{mediana}(I)$** , donde **I** es la imagen con ruido y **Y** es la imagen con el ruido eliminado. Realice la prueba de dicha función con la imagen **imagen1.jpg** en un archivo con nombre **parte1.p1.py**.
- **Pregunta 2** [Valor 10 puntos]: Implemente la función **$Y = \text{rotación}(I, \text{ang})$** , el cual realiza la rotación de una imagen. La función recibe como parámetros una imagen **I** y un ángulo de rotación **ang** (en GRADOS). Para implementar dicha función, utilice el método estudiado en clases (transformaciones afines). Después de realizar la rotación utilizando la transformación afin, utilice la idea del filtro de la mediana para eliminar los pixeles negros obtenidos de la rotación. **Solo se debe aplicar el filtro de la mediana en los pixeles negros, no en los demás pixeles.** Finalmente, aplique dicha función a la imagen **barbara.jpg**, utilizando un ángulo de 45. El nombre del archivo debe ser **parte1.p2.py**.



Figura 1: Ejemplo de rotación de una imagen, utilizando el filtro de la mediana.

Parte 2: Eliminación de Ruido en un Video

Descripción General

- Esta parte de la tarea se basa en eliminar el ruido de un video, usando dos tipos de filtros que se encuentran en el archivo `fast_filter.pdf`.
- La implementación computacional se debe realizar en GNU Octave.
- La parte escrita debe estar en un documento con nombre `T1.P2.pdf`. Todos los resultados numéricos y ejemplos imágenes del video deben aparecer en el documento, y explicado con detalle y claridad.

Preguntas

- **Pregunta 1 [Valor 5 puntos]:** Explique en que consisten los Algoritmos *Fast Median Filter Approximation* (ver Algoritmo 1) y IAMFA-I (verl Algoritmo 2). Para eso escriba el pseudocódigo de cada algoritmo. Para explicar dichos algoritmos, pueden utilizar un dibujo como se muestran en las Figuras 2 y 3.
- **Pregunta 2 [Valor 5 puntos]:** Crear un video donde aparezcan todos los integrantes del grupo. Este video no debe durar menos de 5 segundos ni más de 10 segundos. El nombre del archivo debe ser `con_ruido.mp4`. Este video debe contener ruido del tipo *salt and pepper*. Para crear dicho ruido, deben usar el comando `imnoise` del paquete `image` de GNU Octave. Información del comando `imnoise` se puede encontrar en el siguiente enlace:

<https://octave.sourceforge.io/image/function/imnoise.html>

- **Pregunta 3 [Valor 20 puntos]:** Implemente computacionalmente los Algoritmos *Fast Median Filter Approximation* y IAMFA-I, para eliminar el ruido del video `con_ruido.mp4`. El resultado final debe ser un video sin ruido con los nombres `sin_ruido_alg1.mp4` y `sin_ruido_alg2.mp4`, según corresponda a cada algoritmo implementado. El nombre del archivo donde se encuentran los algoritmos debe ser `filtros_mediana_modificada.m`.
- **Pregunta 4 [Valor 5 puntos]:** El índice de similitud estructural (SSIM de sus siglas en inglés) es un método para determinar la similitud entre dos imágenes. El índice SSIM resultante entre dos imágenes es un valor decimal en el intervalo $[0, 1]$, y el valor 1 solo es accesible en el caso de dos imágenes idénticas, por lo tanto, indica una similitud estructural perfecta. Un valor de 0 indica que no hay similitud estructural entre las imágenes.
 - Utilice la función `ssim` para determinar cual de los dos métodos utilizados en la Pregunta 3 realiza una mejor limpieza del video `con_ruido.mp4`. Justifique su respuesta. Para eso, deben utilizar el video original sin ruido. Aplique el comando `ssim` a cada uno de los *frames* del video. El nombre del archivo de la implementación computacional debe ser `resultado_ssim.m`.

Parte 3: Índice de Calidad de una Imagen Usando Histogramas

Descripción General

- Esta parte de la tarea consiste en implementar un índice de calidad de imágenes utilizando el concepto de histograma, el cual se encuentra en el archivo `histogram_iqi.pdf`.
- La implementación computacional se debe realizar en GNU Octave o en Python.
- La parte escrita debe estar en un documento con nombre `T1_P3.pdf`. Todos los resultados numéricos y ejemplos imágenes deben aparecer en el documento, explicado con detalle y claridad.

Preguntas

- **Pregunta 1 [Valor 5 puntos]:** Investigue en que consiste el concepto de *Índice de Calidad de una Imagen* (en inglés, *Image Quality Index*). Además, realice una breve explicación de los siguientes índices: mean squared error (MSE), peak signal to noise ratio (PSNR), structural similarity index (SSIM) y signal-to-noise ratio (SNR). Adicionalmente, investigue los comandos computacionales para calcular dichos índices (dependiendo del lenguaje de programación seleccionado para resolver esta parte de la tarea).
- **Pregunta 2:** Explique en que consisten el índice de calidad de una imagen usando histogramas (HQI) que se presenta el documento `histogram_iqi.pdf`.
 - a) **[Valor 5 puntos]:** Para eso, realice una breve explicación del índice HQI, explicando la formulación matemática, y presentando un algoritmo que indique los pasos para obtener el índice que se encuentra en la ecuación (1) del documento `histogram_iqi.pdf`.
 - a) **[Valor 10 puntos]:** La sintaxis de la función implementada debe ser `c=hqi(A,B)`, donde A, B son imágenes a escala de grises, y c es el resultado del índice.
- **Pregunta 3 [Valor 10 puntos]:** Luego, utilice los índices MSE, PSNR, SSIM, SNR y HQI para comparar la calidad de las imágenes `imMod1.jpg`, `imMod2.jpg`, `imMod3.jpg`, `imMod4.jpg`, `imMod5.jpg`, `imMod6.jpg` y `imMod7.jpg`, con la imagen `imOrig.jpg`. El nombre del archivo de la implementación computacional debe ser `indice.imagenes`. Adicionalmente, presente en el documento escrito todos los resultados en una tabla como la siguiente:

Imagen	MSE	PSNR	SSIM	SNR	HQI
<code>imMod1.jpg</code>					
<code>imMod2.jpg</code>					
<code>imMod3.jpg</code>					
<code>imMod4.jpg</code>					
<code>imMod5.jpg</code>					
<code>imMod6.jpg</code>					
<code>imMod7.jpg</code>					

Observación: En el documento escrito deben estar las imágenes utilizadas para el desarrollo de este ejemplo, además de una breve explicación del ejemplo presentado.