MAP (Mini Aplicación)
Serie de Fourier Truncada y FFT

1. Objetivo

Hacer uso de las series de fourier en distintas aplicaciones de ingeniería, como encontrar la aproximación la serie de Fourier truncada calculando la cantidad de terminos con la integral cuadarada del error e incluyendo la eliminación de ruido en imágenes utilizando filtros pasivos. Se utilizarán las series de Fourier para descomponer señales y analizar cómo los filtros afectan su comportamiento.

2. Marco Teórico

Las *Series de Fourier* describen señales periódicas como una combinación de señales armónicas. Uno de sus objetivo es mostrar que una función periódica puede descomponerse como *suma de funciones trigonométricas*, como senos y cosenos. Donde las frecuencias son múltiplos enteros de la frecuencia fundamental. La *serie de Fourier Trigonométrica (SFT)* está dada por:

$$f(t) = a_0 + \sum_{n=1}^{\infty} [a_n \cos(n\omega t) + b_n \sin(n\omega t)] \quad , \quad \omega = \frac{2\pi}{T}$$

donde

$$a_0 = \frac{1}{T}\int_0^T f(t)dt \quad , \quad a_n = \frac{2}{T}\int_0^T f(t)\cos(n\omega t)dt \quad , \quad b_n = \frac{2}{T}\int_0^T f(t)\sin(n\omega t)dt$$

Serie de fourier truncada a $2N+1$ términos: al querer aproximar una función periódica $f(t)$ la cual es continua por partees y posee infinitos armónicos, tendremos que truncar la función hasta el armónico N. La forma de calcular la energía de la señal:

$$E_f = \int_0^T f^2(t)dt$$

Integral cuadrada del error (ICE): el cálculo de ICE se utiliza para encontrar el valor N que luego será utilizado para obtenr una representación de la SFT.

$$ICE(N) = E_f - \left\{ a_0^2 T + \frac{T}{2}\sum_{n=1}^{N}(a_n^2 + b_n^2) \right\}$$

$$ICE(N) \leqslant 0.02E_f$$

Transformada de Fourier Discreta (DFT): es una técnica matemática utilizada para analizar las componentes de frecuencia de una señal discreta o una secuencia. Dada una secuencia $\mathbf{x}[n]$ de longitud N, la DFT $\mathbf{X}[k]$ es calculada utilizando la siguiente fórmula:

$$\mathbf{X}[k] = \sum_{n=0}^{N-1} \mathbf{x}[n] \cdot e^{\frac{-2\pi k\, n\, i}{N}}$$

donde:

- $\mathbf{X}[k]$ es el $k-$ésimo elemento de la secuencia de salida de la DFT.
- $\mathbf{x}[n]$ es el $n-$ésimo elemento de la secuencia de entrada.
- i es la unidad imaginaria, $i^2 = -1$.
- k es el índice de la secuencia de salida, que va de 0 a $N - 1$.
- N es el número total de muestras en la secuencia de entrada.

Transformada Rápida de Fourier (FFT) es un algoritmo eficiente para calcular la Transformada de Fourier Discreta. Reduce la complejidad computacional de la DFT de $O(N^2)$ a $O(N \log N)$ al aprovechar las propiedades de periodicidad de los exponenciales complejos. Uno de los algoritmos de FFT ampliamente utilizados es el *algoritmo FFT de Cooley-Tukey*, que divide de forma recursiva el cálculo de la DFT en subproblemas más pequeños. Los principales pasos del algoritmo FFT de Cooley-Tukey incluyen:

1. Dividir la secuencia de entrada en índices pares e impares para crear dos subproblemas más pequeños.
2. Calcular recursivamente la DFT de los subproblemas pares e impares.
3. Combinar los resultados de la DFT de los subproblemas pares e impares para calcular el resultado final de la DFT.

Al aplicar estos pasos de forma recursiva, el algoritmo FFT reduce significativamente la complejidad temporal del cálculo de la DFT.

3. Guía Práctica

En este proyecto, se dividirá en dos partes: i) series de Fourier para encontrar forma truncada de una serie a $2N + 1$ términos, encontar la energía de una función $f(t)$ y ii) aplicar los filtros generados por fourier a imágenes.

Parte I – Series de Fourier

1. Para esta parte usted podrá hacer uso de cualquier lenguaje de programación en donde su interfaz cumpla con los siguientes requerimientos mínimos:
   - Lo único dado de ingreso por el usuario debe ser la función a trabajar con los intervalos que correspondan.
   - Los datos desplegados por su interfaz al usuario deben ser:
     - Período de la función ingresada.
     - Coeficientes de la serie de Fourier para la función ingresada.
     - Gráfica aproximada de la serie de Fourier trigonométrica truncada a $2N+1$ términos.
     - Valor de N
     - Ecuación de la serie de Fourier trigonométrica.

2. Considere las funciones:

$$i)\ f_1(t) = \begin{cases} 0, & -2 < t < -1 \\ 1, & -1 < t < 1 \\ 0, & 1 < t < 2 \end{cases} \qquad ii)\ f_2(t) = \begin{cases} 0, & -2 < t < -1 \\ t+2, & -1 < t < 1 \\ 0, & 1 < t < 2 \end{cases}$$

junto con su interfaz generada en la parte 1, realice lo siguiente:

   a) Documente paso a paso, es decir colocando todos los datos generados por su interfaz, para los valores $N = 3,\ 5,\ 10$ para cada función.

   b) Responda a las preguntas: ¿En cuál de las dos funciones converge más rapido la serie de Fourier Trigonométrica? ¿Es útil el cálculo de ICE para ambas funciones? ¿Con cuantas iteraciones podemos obtener una mejor aproximación para las funciones dadas?

Parte II – FFT

A continuación se presenta el algoritmo de FFT, el cual deberá buscar una implementación del mismo o bien implementarlo en *cualquier* lenguaje de programación.

Cooley-Tukey Fast Fourier Transform (FFT) Algorithm

---

function FFT(x, N)
  if $N = 1$ **then**
    return x
  end if
  $x_{\text{even}} \leftarrow \text{FFT}([x_0, x_2, \ldots, x_{N-2}], N/2)$ ▷ Even-indexed sub-sequence
  $x_{\text{odd}} \leftarrow \text{FFT}([x_1, x_3, \ldots, x_{N-1}], N/2)$ ▷ Odd-indexed sub-sequence
  for $k \leftarrow 0$ **to** $N/2 - 1$ **do**
    $T \leftarrow x_{\text{odd}}[k] \times e^{-2\pi k\, i/N}$ ▷ i is the imaginary unit
    $X[k] \leftarrow x_{\text{even}}[k] + T$
    $X[k + N/2] \leftarrow x_{\text{even}}[k] - T$
  end for
  return X
end function

Considere las imágenes adjuntas en el proyecto. Si usted desea utilizar sus propias imágenes puede hacerlo. En base algoritmo anterior, realice los incisos que se detallan a continuación.

1. Aplique el algoritmo a las imágenes proporcionadas utilizando un *filtro pasa bajas* con las siguientes frecuencias de corte:

   a) $f_{\text{cut off}} = 0.2$,

   b) $f_{\text{cut off}} = 0.1$,

   c) $f_{\text{cut off}} = 0.9$,

¿Cuál es la diferencia entre las imágenes originales y las generadas? Justifique su respuesta analizando el filto pasa bajas.

2. Aplique el algoritmo a las imágenes proporcionadas utilizando un *filtro pasa altas* con las siguientes frecuencias de corte:

   a) $f_{\text{cut off}} = 0.2$,
   b) $f_{\text{cut off}} = 0.1$,
   c) $f_{\text{cut off}} = 0.9$,

   ¿Cuál es la diferencia entre las imágenes originales y las generadas? Justifique su respuesta analizando el filto pasa altas.

3. **Extras:** modifique su implementación del algoritmo o la implementación utilizada para que pueda realizar *Blur Detection*. Luego, aplique su implentación con Blur Detection a las imágenes proporcionadas.

4. Instrucciones Generales

1. Este proyecto se realizará en grupos de un máximo de *tres* integrantes.
2. Deben enviar un correo o un mensaje en Telegram a su asistente de cátedra *antes del miércoles 9 de Octubre de 2024* para la creación de los grupos de trabajo en el GES. Dicho correo o mensaje debe incluir el nombre y carné de cada uno de los integrantes del grupo.
3. Su reporte deberá realizarlo usando LaTeX.
4. Deberá realizar un pequeño reporte que incluya la respuesta a cada uno de los inciso que se detallan en la sección 3. Recuerde que respuesta no justificada no recibirá calificación, por ello debe dejar constancia de todo razonamiento y procedimiento realizado.
5. **Fecha de Entrega: Domingo 3 de Noviembre de 2024.** Su solución debe subirla en un archivo ZIP enviado por GES y debe contener:
   - Archivo editable de LaTeX (.tex) de su reporte.
   - Imágenes utilizadas en su archivo de LaTeX.
   - Su reporte en formato PDF generado por su archivo de LaTeX.
   - Los archivos correspiendientes de la interfaz generada en la parte I de la guía práctica.

   de lo contrario su trabajo *no* será tomado en cuenta.

5. Referencias

Acerca de Series de Fourier.

a) Notas de clase, Portal de Matemática VI en GES.

b) Dyke, P.P.G: An Introduction to Laplace Transforms and Fourier Series, Springer Undergraduate Mathematics Series. Springer, 2a. edición, 2014.

c) Zill, D., W. Wright y M. Cullen: Matemáticas Avanzadas para Ingeniería, McGraw-Hill, 4a. edición, 2012.

Herramientas de Graficación.

a) Página oficial de SageMath - `http://www.sagemath.org/`

b) Página oficial de Cocalc - `https://cocalc.com/`

Acerca de LaTeX

a) Página oficial de MikTex - `https://miktex.org/`

b) Página oficial de TeXstudio - `https://www.texstudio.org//`

c) Página oficial de TeXmaker - `https://www.xm1math.net/texmaker/`

d) Página oficial de Overleaf - `https://www.overleaf.com/`