



Tarea 1: Parte 2
Ingeniería en Computadores

CE 5201: Procesamiento y Análisis de Imágenes Digitales

Profesor: Juan Pablo Soto Quiros

Nasser Santiago Brown Aparicio 2019043776

Kenichi Hayakawa Bolaños 2020044884

Jose Antonio Retana Corrales 2020144743

Explique en qué consisten los Algoritmos Fast Median Filter Approximation (ver Algoritmo 1) y IAMFA-I (ver Algoritmo 2). Para eso escriba el pseudocódigo de cada algoritmo. Para explicar dichos algoritmos

- **Algoritmo 1: Fast Median Filter Approximation:** Este algoritmo busca aproximar el valor de la mediana en una ventana de tamaño 3x3 para acelerar el proceso de filtrado. El proceso divide la ventana en 3 columnas, calcula la mediana de cada columna y luego obtiene la mediana de esas medianas. Este método reduce el tiempo de ejecución evitando el ordenamiento completo de la ventana.

```

1 # Fast Median Filter Approximation
2
3 # Entrada: Imagen I de tamaño H x W
4 # Salida: Imagen filtrada I
5
6
7 Para i desde 2 hasta H-1:
8     # Calcular la mediana para la primera columna de la ventana 3x3
9     col1 <- mediana(I(i-1, 1), I(i, 1), I(i+1, 1))
10
11     # Calcular la mediana para la segunda columna de la ventana 3x3
12     col2 <- mediana(I(i-1, 2), I(i, 2), I(i+1, 2))
13
14     Para j desde 3 hasta W-1:
15         # Calcular la mediana para la tercera columna de la ventana
16         # 3x3
17         col3 <- mediana(I(i-1, j), I(i, j), I(i+1, j))
18
19         # Calcular la mediana entre las tres columnas y reemplazar
20         # el valor en la imagen filtrada
21         I(i, j) <- mediana([col1, col2, col3])
22
23         # Desplazar las columnas para la siguiente iteración
24         col1 <- col2
25         col2 <- col3
26     Fin del bucle interno
27 Fin del bucle externo

```

- **Algoritmo 2: IAMFA-I:** El IAMFA-I introduce una técnica llamada "Mid-Value-Decision-Median" para seleccionar un valor más robusto que evita la selección de valores corruptos (impulsos de ruido) como la mediana.

```

1 # IAMFA-I Improved Approximated Algorithm
2
3 # Entrada: Imagen I de tamaño H x W
4 # Salida: Imagen filtrada I
5
6
7 Para i desde 2 hasta H-1:
8     # Calcular la mediana usando la técnica Mid-Value-Decision
9     # para la primera columna
10     col1 <- Mid-Value-Decision Median(I(i-1, 1), I(i, 1), I(i+1, 1))
11
12     # Calcular la mediana usando la técnica Mid-Value-Decision
13     # para la segunda columna
14     col2 <- Mid-Value-Decision Median(I(i-1, 2), I(i, 2), I(i+1, 2))
15
16     Para j desde 3 hasta W-1:
17         # Calcular la mediana usando la técnica Mid-Value-Decision
18         # para la tercera columna
19         col3 <- Mid-Value-Decision Median(I(i-1, j), I(i, j), I(i+1, j))
20
21         # Calcular la mediana final usando la técnica Mid-Value-Decision
22         # para las tres columnas
23         I(i, j) <- Mid-Value-Decision Median([col1, col2, col3])
24
25         # Desplazar las columnas para la siguiente iteración
26         col1 <- col2
27         col2 <- col3
28     Fin del bucle interno
29 Fin del bucle externo

```

Resultados:

Se realizó el cálculo del SSIM de cada frame en ambos videos con respecto al original y luego se promediaron.

Los resultados obtenidos señalan al algoritmo 1 (Fast Median Filter Approximation) ligeramente con mayor afinidad al video original. El mejor desempeño del algoritmo 1 se justifica en la simplicidad de su aproximación y su enfoque en preservar la estructura de la imagen en situaciones de ruido moderado. IAMFA-I podría sobresalir en escenarios con ruido más intenso, pero su complejidad adicional no siempre es beneficiosa en todos los contextos.

```
El SSIM promedio para el algoritmo 1 es: 0.963452
El SSIM promedio para el algoritmo 2 es: 0.940956
```

Bibliografía:

Marques, O. (2011). *Practical image and video processing using MATLAB*. Wiley.

González, R., & Woods, R. (2018). *Digital image processing* (4th ed.). Prentice-Hall.