# Pregunta 1

El *Fast* *Median Filter Approximation* o DP al contrario del filtro regular no utiliza una ventana cuadrada para recorrer toda la imagen celda por celda, sino utiliza la mediana de tres valores de dos columnas (val1 y val2) y analiza la mediana de ellos con un valor la tercera columna hasta recorrer todas utilizando los valores de las dos columnas anteriores recursivamente. Luego se desplaza una columna y se repite hasta completar la imagen. El pseudo código correspondiente luce así:

Sea I la imagen a filtrar, I’ la imagen filtrada

Sea H la altura de la imagen

Sea W el ancho de la imagen

for i = 2 to H-1 // recorrer cada columna

val1 = mediana(I(i-1, 1), I(i, 1), I(i+1, 1)); // valor de columna 1

val2 = mediana(I(i-1, 2), I(i, 2), I(i+1, 2)); // valor de columna 2

for j = 3 to W-1 // recorrer cada fila

val3 = mediana(I(i-1, j), I(i, j), I(i+1, j)); // valor de columna 3

I’(i, j) = mediana(val1, val2, val3); // valor final de celda recuperada

val1 = val2; // cambio de valor

val2 = val3; // cambio de valor

endfor

endfor

Un diagrama que ilustre el movimiento es el siguiente:

j++

Celda i,j

Celda i,j

Celda i,j+1

Mediana de cada mediana

IAMFA-I se comporta diferente. Este tiene por objetivo evadir pixeles absolutamente negros o blancos (0 o 255 en valores enteros de 8 bits). Realiza el mismo recorrido que el algoritmo DP excepto que no calcula una mediana, si no que ordena los valores de los pixeles de mayor a menor (sea P1, P2 ,P3 valores de celdas y el orden P1 ≤ P2 ≤ P3) y selecciona el valor final de celda según el valor del valor medio (P2) o *Mid Value Decision Median* (MVDM):

El pseudo código correspondiente es:

Sea I la imagen a filtrar; I’ la imagen filtrada;

Sea H la altura de la imagen;

Sea W el ancho de la imagen;

for i = 2 to H-1 // recorrer cada columna

val1 = MVDM(I(i-1, 1), I(i, 1), I(i+1, 1)); // valor de columna 1

val2 = MVDM(I(i-1, 2), I(i, 2), I(i+1, 2)); // valor de columna 2

for j = 3 to W-1 // recorrer cada fila

val3 = MVDM(I(i-1, j), I(i, j), I(i+1, j)); // valor de columna 3

I’(i, j) = MVDM(val1, val2, val3); // valor final de celda recuperada

val1 = val2; // cambio de valor

val2 = val3; // cambio de valor

endfor

endfor

# Pregunta 2

El video\_con\_ruido.mp4 ubicado en la carpeta “videos” fue creada utilizando 3 videos separados: jung.mp4, daniel.mp4 y dago.mp4. Estos videos tienen una resolución de 160 x 160 pixeles. El video\_con\_ruido.mp4 esta constituido por 255 cuadros y se despliega a una velocidad de 30 cuadros por segundo dando una longitud de 7 minutos con 30 segundos de video. El ruido implementado el video es un ruido sal y pimienta con un valor de 0.01. Por efectos de la tarea que se van a ver a continuación se creó una versión del video sin ruido llamado video\_sin\_ruido.mp4. El resultado de video\_con\_ruido.mp4 se puede observar en la Figura 1 y el de video\_sin\_ruido.mp4 en la Figura 2. El código utilizado para realizar el video\_con\_ruido.mp4 se encuentra bajo el nombre noisy\_join\_videos.m.

  
Figura 1. Cuadro del video **video\_con\_ruido.mp4**

  
Figura 2. Cuadro del video **video\_sin \_ruido.mp4**

# Pregunta 3

En las Figuras 3 y 4 se puede observar el resultado de la aplicación del algoritmo *Fast Median Filter Aproximation* (Figura 3) y IAMFA-I (Figura 4) al cuadro del video **video\_con\_ruido.mp4** observable en la Figura 1. El desarrollo del algoritmo *Fast Median Filter Aproximation* se ubica en el archivo remove\_noise\_FMFA.m y el algoritmo IAMFA-I en el archivo remove\_noise\_IANFAI.m. Los resultados de la aplicación del se encuentran dentro del carpeta “videos” bajo el nombre de video\_sin\_ruido\_alg1.mp4 para el *Fast Median Filter Aproximation* y video\_sin\_ruido\_alg2.mp4 para el IAMFA-I.



Figura 3. Resultado de la aplicación del *Fast Median Filter Aproximation*



Figura 4. Resultado de la aplicación del IAMFA-I

# Pregunta 4

Cuando se aplica el índice de similitud estructural (SSIM) a los videos video\_sin\_ruido\_alg1.mp4 y video\_sin\_ruido\_alg2.mp4 en referencia al video\_sin\_ruido.mp4 se da que el *Fast Median Filter Aproximation* tiene un valor de 0.86111 por mientras que el IAMFA-I tiene un valor de 0.86065. Según el resultado obtenido se tiene que el Fast Median Filter Aproximation realiza una limpieza de la imagen mejor.