

Captura y Procesamiento Digital de Señales e Imágenes

Trabajo de Aplicación

Filtro Adaptativo de mediana

Alumnos:

Long Sebastián.

Villanueva Raúl.

Zalazar Gustavo.

Introducción:

En el área de captura y procesamiento digital de imágenes es común encontrarse con imágenes que posean ruido, debido a defectos de los dispositivos de adquisición o a los medios de transmisión.

Estas imágenes pueden estar contaminadas por diferentes tipos de ruido, como ser gaussiano, impulsivo, gamma, exponencial, uniforme, etc., los cuales la afectan de diferentes maneras.

Existen métodos que logran un adecuado desempeño para determinados tipos de ruido y fallan en otros.

En particular, un filtro de mediana trabaja teniendo en cuenta las características estadísticas globales de la imagen.

El filtro adaptativo de mediana aprovecha las propiedades del filtro de mediana con la diferencia que este se adapta a las características locales de la imagen de una zona a otra en donde es analizado.

Este filtro trabaja para eliminar el ruido impulsivo, sin provocar un desenfoque en la imagen.

Este ruido puede presentarse en dos formas

En este informe se verá como es el funcionamiento del filtro adaptativo de mediana.

Desarrollo

Como fue mencionado anteriormente este filtro solo trabaja con ruido impulsivo. Este ruido tiene la característica de tomar los valores mínimos y máximos del rango de grises (0 - 1).

El filtro de mediana trabaja creando una ventana fija en la cual trata de eliminar el ruido impulsivo calculando la media de los valores de los píxeles contenidos dentro de la ventana, y colocando ese valor en el píxel evaluado. Este proceso se repite desplazando la ventana por toda la imagen.

Uno de los defectos que encontramos en este filtro es que produce un leve desenfoque en la imagen, entendiéndose por esto a la pérdida de detalle en la imagen.

La otra gran deficiencia que encontramos es su mal funcionamiento cuando la densidad de ruido supera el valor 0.2, debido a que la ventana evaluada contendrá en su mayoría valores correspondientes a ruido y no a los propios de la imagen. Cuando este calcule la media es más probable encontrar valores de ruido en la imagen.

El filtro de mediana adaptativo es un filtro no lineal.

Este también trabaja creando una ventana, la cual a diferencia del filtro de mediana, esta no es fija, es decir, cambia su tamaño durante la operación de filtrado dependiendo de la cantidad de ruido que posee la ventana evaluada.

Implementación del Algoritmo

Las variables a utilizar son:

Zmin = mínimo valor del nivel de gris en la ventana.

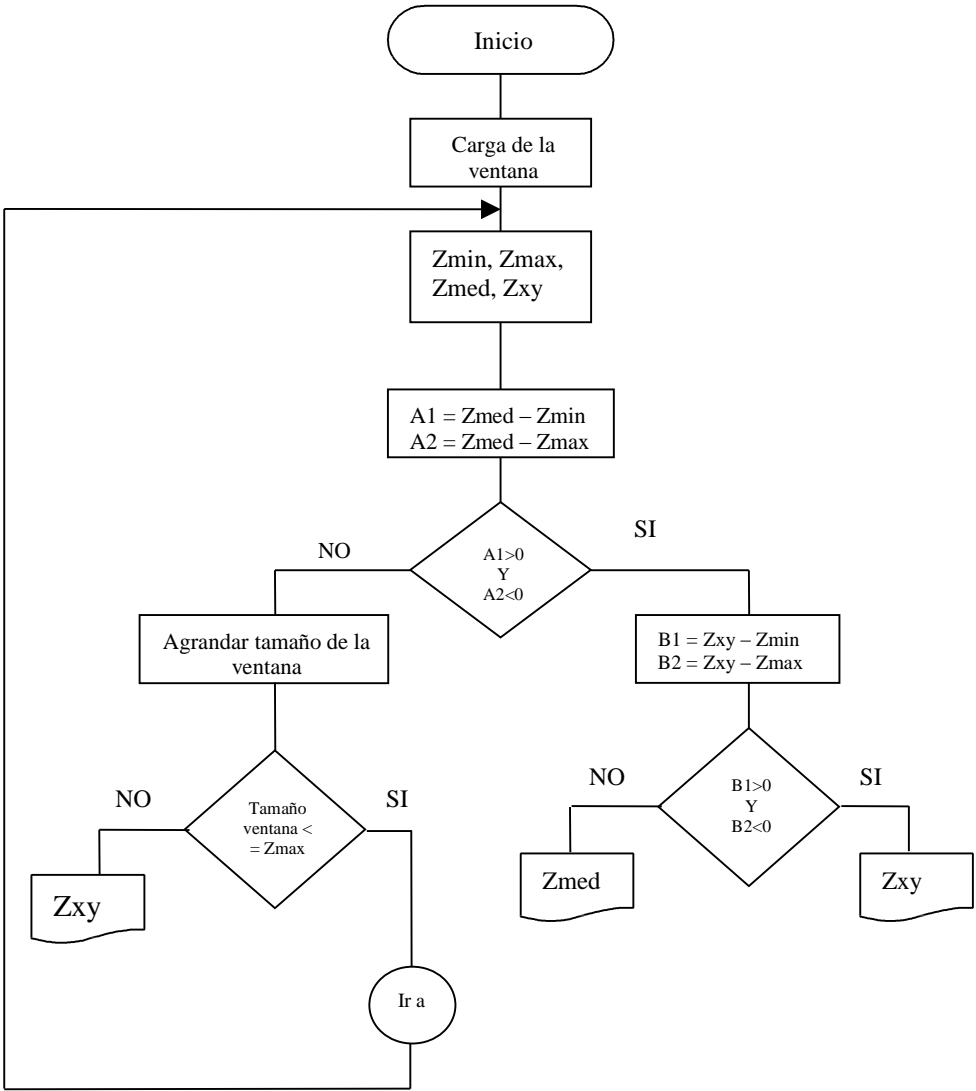
Zmax = máximo valor del nivel de gris en la ventana.

Zmed = mediana del nivel de gris en la ventana.

Smax = máximo tamaño en que se puede agrandar la ventana.

Zxy = nivel de gris de la coordenada xy.

Este diagrama de flujo muestra como opera el algoritmo del filtro adaptativo de mediana. El cual para cada pixel de la imagen hace lo siguiente:



Primero se carga la ventana dándole un tamaño inicial, para nuestro análisis fue hecha con una ventana cuadrada de 3x3.

Luego se calcula en la misma Z_{max} , Z_{min} y Z_{med} . Con estos valores formamos A_1 y A_2 con el fin de averiguar si Z_{med} es ruido o no. Dado que si Z_{med} no es ruido el valor de A_1 siempre va a ser mayor a cero, y A_2 siempre va a ser menor que cero. Con esto verifico que el valor de Z_{med} no es ruido.

Una vez comprobado que Z_{med} no es ruido nos queda por determinar si el píxel evaluado es ruido o no para realizar esto calculamos los valores de $B1$ y $B2$. Dado que si Z_{xy} no es ruido el valor de $B1$ siempre va a ser mayor a cero, y $B2$ siempre va a ser menor que cero. Con esto verifico que el valor de Z_{xy} no es ruido entonces asigno el valor de Z_{xy} a la imagen que queda como salida. En caso contrario inserto en la imagen de salida el valor de Z_{med} el cual ya esta verificado que no es ruido por lo anterior mencionado.

Si se comprueba que Z_{med} es un valor máximo o mínimo, es decir, $A1=0$ o $A2=0$ de la ventana analizada se procede a agrandar la ventana de análisis para evitar considerar que la ventana contenga mas ruido que imagen. Una vez hecho esto volvemos a calcular los valores correspondientes al nuevo tamaño de la ventana. Esto se repetirá hasta alcanzar el valor S_{max} (tamaño máximo de la ventana a considerar). Al llegar al S_{max} el algoritmo considera que el píxel evaluado corresponde a una zona homogénea (todo blanco o todo negro) de la imagen dejando como salida Z_{xy} .

Análisis Comparativo

Para este análisis utilizamos la imagen circuito.bmp que posee una dimensión de 288x278, con 8 bit por nivel de gris. Para la evolución la imagen fue ensuciada con ruido sal (con valor 0) y pimienta (con valor 1) con una densidad que va desde 0.1 a 0.7, con paso 0.1.

El error en la comparación se midió utilizando la media de error absoluta (MDEA).

$$MDEA = \frac{1}{MN} \sum_{i,j} |r_{i,j} - x_{i,j}|$$

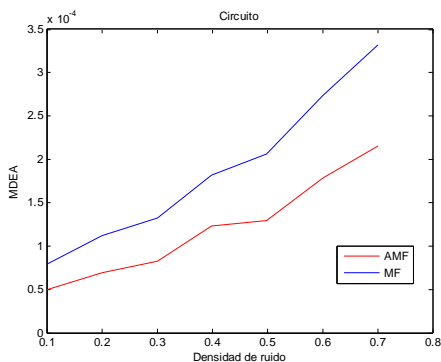
Donde r es la imagen limpia y x es la imagen filtrada.

Para el algoritmo de la mediana (medfilt2) fuimos modificando el tamaño de la ventana, de 3x3, 5x5 y 7x7 (una grafica para cada una). Mientras que para la media adaptativa es tamaño máximo de ventana es de 7x7, con borde fijo (intensidad 0).

Medfilt2 con Ventana 3x3

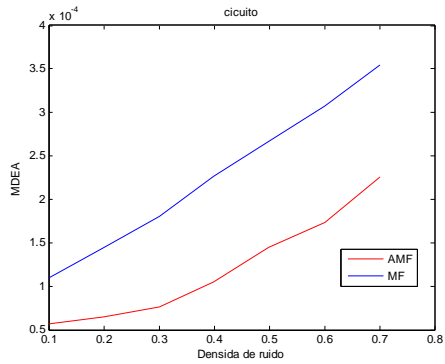
Figura 1

a-



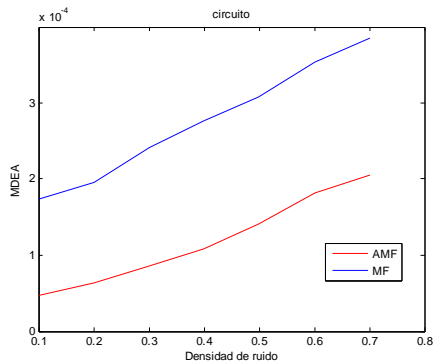
Medfilt2 con ventana 5x5

b-



Medfilt2 con Ventana 7x7

c-



Nosotros resumimos la performance de los dos métodos en las tres figuras anteriores (a,b,c). Desde las graficas vemos que los métodos tienen similar estructura, las dos son monótonas crecientes.

Cuando la densidad de ruido es baja la diferencia del error entre ellas es la mas baja, pero cuando esta crece las diferencias aumentan. Cuando modificamos el tamaño de la ventana en el medfilt2 aumenta el MDEA en densidades bajas y manteniéndose en altas. Este comportamiento se debe a que a baja densidad de ruido (hasta 0.2) el filtro de mediana trabaja bastante bien, es decir, su desempeño se asemeja a el filtro adaptativo de mediana. Pero cuando el nivel de densidad del ruido crece, el filtro de mediana empieza a fallar.

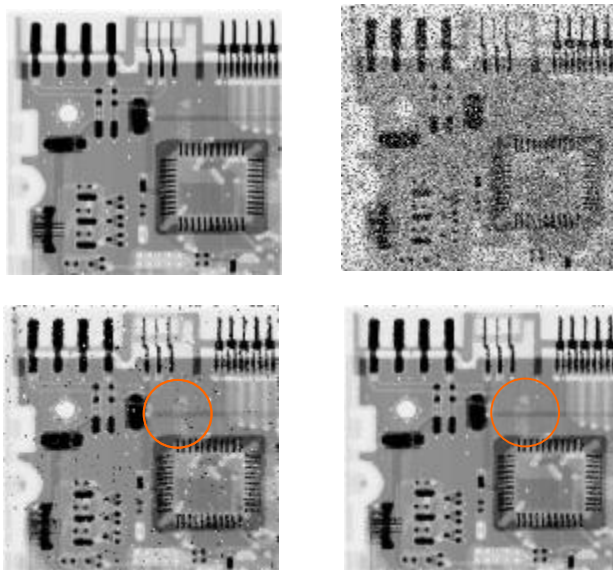


Figura 2
a-b
c -d

En la figura 2 nosotros presentamos la restauración de un circuito (fig N°2a) ensuciada con ruido sal y pimienta con densidad 0.3 (fig N°2b). Como era de esperar el filtro de mediana adaptativo dio un mejor resultado (fig N°2d), que el filtro de mediana (fig N°2e). Esto se puede notar en la figura N°2c todavía existe ruido, mientras que en la figura N°2d el ruido fue eliminado totalmente.

Como mencionamos anteriormente el filtro de mediana pierde detalle mientras que el adaptativo no, un ejemplo de esto se ve en los círculos rojos en la figura N°2c y d. Aquí se puede apreciar claramente que en una se conserva la marca del circuito y en la otra no.

Los puntos negros que se observan en los contornos de la imagen de la figura N°2d, se debe a la elección de un borde fijo con intensidad cero.

Criticas

El único inconveniente que encontramos en este filtro es que solo funciona para restaurar imágenes contaminada con ruido sal y pimienta, con otros tipos de ruido se comporta deficientemente.

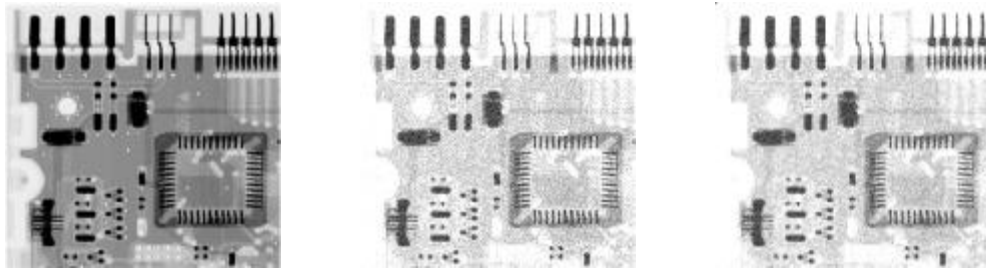


Figura 3
a-b-c

En la figura N°3a tenemos la imagen original, a esta la contaminamos con ruido gaussiano figura N°3b y luego le aplicamos el filtro adaptativo de mediana (figura N°3c) y podemos observar que no produce cambios, es decir, no elimina el ruido.

El costo computacional de implementación de este algoritmo es mayor que el de mediana.

Conclusión

En este trabajo aplicativo, analizamos e implementamos el filtro adaptativo de mediana. Este filtro logro buenas prestaciones para la eliminación del ruido impulsivo tipo sal y pimienta.

Los resultados experimentales muestran que este método logra mucha mejor resultado que el filtro de mediana preservando los detalles, texturas y bordes. Esto es mas notorio cuando el nivel de ruido es alto.