



Revista INGENIERÍA UC

ISSN: 1316-6832

revistaing@uc.edu.ve

Universidad de Carabobo

Venezuela

Latouche, Hyxia; Solarte, Kelvin; Ordoñez, Joan; Sánchez, Luis
Filtros no lineales para reducir ruido en imágenes a color
Revista INGENIERÍA UC, vol. 24, núm. 2, agosto, 2017, pp. 185-195
Universidad de Carabobo
Carabobo, Venezuela

Disponible en: <http://www.redalyc.org/articulo.oa?id=70753474006>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica

Red de Revistas Científicas de América Latina, el Caribe, España y Portugal

Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Nonlinear Filters to denoising color images

Hyxia Latouche^a, Kelvin Solarte^b, Joan Ordoñez^{*,c}, Luis Sánchez^c

^aDepartamento de Matemática, Facultad de Ingeniería, Universidad de Carabobo, Valencia, Venezuela.

^bDepartamento de Matemática y Física, Facultad de Ciencias de la Educación, Universidad de Carabobo, Venezuela.

^cInstituto de Matemática y Cálculo Aplicado, Facultad de Ingeniería, Universidad de Carabobo, Valencia, Venezuela.

Abstract.-

The present investigation consisted in the implementation Montecarlo Sequential methods to restore, in real time, color images altered by Gaussian noise. Were specifically implemented the algorithms of Divided Difference Filter, Adaptive Central Difference and Cubature Kalman Filter. The method of filtering only the luminance of the image by transforming RGB to YIQ is used in order to reduce the computation time. For the restoration of the images, a filtering process is proposed by making two paths, one horizontal and one vertical in the matrix of the Y layer. The resulting filtered layers were averaged in order to obtain a better estimation of the values for each pixel after filtration. The results showed that the implemented filters perform a significant noise reduction compared to the original image. The efficiency of the filters was evaluated in terms of the mean absolute error (EAM) and the peak signal-to-noise ratio (PSNR) of the estimated states, showing little variation among the calculated errors. Finally, related with the execution time of algorithms, a significant difference was found between the filters, being the Divided Difference filter the most efficient.

Keywords: nonlinear filters; images processing; luminance; noise reduction.

Filtros no lineales para reducir ruido en imágenes a color

Resumen.-

La presente investigación consistió en la implementación de métodos Montecarlo Secuenciales para restaurar, en tiempo real imágenes a color alteradas con ruido Gaussiano. Se implementaron específicamente los algoritmos: filtro de diferencia dividida, filtro adaptativo de diferencia centrada y filtro de Kalman Cubatura. Se utilizó un método de filtrado en la luminancia de la imagen establecida por una transformación de RGB a YIQ con el fin de reducir el tiempo de computo. Para restaurar las imágenes, se plantea el filtrado haciendo dos recorridos, uno horizontal y otro vertical en la matriz de la capa Y. Las capas filtradas resultantes se promediaron para obtener una mejor estimación de los valores de cada pixel tras el filtrado. Los resultados evidenciaron que los filtros implementados realizan una notable reducción del ruido al compararla con la imagen original. El rendimiento de los filtros se evaluó en términos del error absoluto medio (EAM) y el pico de relación señal ruido (PSNR) de los estados estimados, mostrando poca variación entre los errores calculados. En cuanto a los tiempos de ejecución de los algoritmos, se obtuvo diferencia significativa entre los filtros, siendo el filtro de diferencias divididas el más eficiente.

Palabras clave: filtros no lineales; procesamiento de imágenes; luminancia; reducción de ruido.

Recibido: junio 2017

Aceptado: agosto 2017

1. Introducción

Una imagen digital x es un arreglo rectangular con valores de intensidad $\{x_{i,j}\}_{i,j=1}^{m,n}$. Para imágenes en escala de grises, el valor $x_{i,j}$ es un número simple, mientras que en las imágenes a color cada $x_{i,j}$ es un vector de tres o más valores [1].

*Autor para correspondencia

Correo-e: jordonez@uc.edu.ve (Joan Ordoñez)

Si la imagen es registrada bajo el modelo RGB, cada $x_{i,j}$, es un vector de tres valores

$$x_{i,j} = (r_{i,j}, g_{i,j}, b_{i,j})$$

los cuales denotan la cantidad de rojo, verde y azul en el punto (i, j) .

Las imágenes de colores pueden ser representadas por tres matrices, cada una de las cuales especifica la cantidad de rojo, verde y azul que contiene la imagen. Este sistema de colores es conocido en sus siglas en inglés como RGB (Red, Green and Blue). Los elementos que integran esas matrices son números enteros entre 0 y 255, y determinan la intensidad del pixel con respecto al color de la matriz.

La imágenes naturales son afectadas por ruidos generados por la captura de los instrumentos, los medios de transmisión de datos, cuantificación en las imágenes y fuentes discretas de radiación. En las imágenes el ruido aleatorio puede ser modelado como una distribución Gaussiana [2].

Un modelo para representar el ruido es descrito por las ecuaciones de estado y observación en tiempo discreto, definido como:

$$x_t = \mathcal{F}_t(x_{t-1}) + u_t \quad (1)$$

$$y_t = \mathcal{H}_t(x_t) + v_t \quad (2)$$

La ecuación dada en (1) representa una ecuación de estado, donde $x_t \in \mathbb{R}^n$ denota un arreglo rectangular desconocido con valores de intensidad en un tiempo t , u_t es un error aleatorio de estimación del estado, \mathcal{F}_t es un operador de transición que mapea el espacio estado dentro del mismo espacio estado. La Ecuación dada en (2) representa una ecuación de observación, donde \mathcal{H}_t es un operador que mapea el espacio estado dentro del espacio de las observaciones en el tiempo t , y_t es el vector de observaciones y v_t es un error aleatorio de observación.

La restauración de los píxeles que contribuyen a restaurar las imágenes es construido por la gráfica del vector de estados x_t en el espacio de estado. Es por ello que una tarea importante es estimar los estados desconocidos $x_t = x_{0:t} = (x_0, x_1, \dots, x_t)$, basándose en las medidas obtenidas a partir del

proceso de observación $y_t = y_{1:t} = (y_1, y_2, \dots, y_t)$. La idea principal es estimar recursivamente en el tiempo t la distribución a posteriori $P(x_{0:t}|y_{1:t})$ y sus características asociadas incluyendo la distribución marginal $P(x_t|y_{1:t})$, sus esperanzas, modas y varianzas.

Uno de los aportes de este trabajo consiste en aplicar técnicas de reducción de ruido. Se utilizaron específicamente los métodos Montecarlo secuenciales: el filtro de diferencia dividida, el filtro adaptativo de diferencia centrada y el filtro de Kalman Cubature, empleados para restaurar imágenes. Para una revisión extensa de los métodos Montecarlo secuenciales ver: [3, 4, 5], entre otros.

Asimismo, se plantea el filtrado de imágenes a color transformandola de RBG a YIQ, para luego filtrar la luminancia representada por la capa Y sin procesar las otras dos capas. El proceso de filtrado se realizó haciendo dos recorridos, uno horizontal y otro vertical, en la matriz de la capa Y (luminancia) de las imágenes tratadas.

Para validar los resultados obtenidos se utilizaron dos medidas de bondad de ajuste: el error absoluto medio (EAM) y el pico de relación señal ruido (PSNR) de los estados estimados x_t .

El resto del artículo es como sigue: en la Sección 2 se describe la luminancia y crominancia de una imagen; en la Sección 3 se describe el ruido en imágenes digitales; en la Sección 4 se describe el procesamiento de imágenes a color; en la Sección 5 se muestran los filtros no lineales; en la Sección 6 se muestra las medidas de bondad de ajuste; en la Sección 7 se muestra los procedimientos y resultados, y en la Sección 8 se muestra una discusión de los resultados obtenidos.

2. Luminancia y crominancia de una imagen

Existen modelos relacionados con la manera en que los seres humanos perciben el color basado en tres propiedades como coordenadas del mismo. Estas son: luminancia, matiz y saturación.

La luminancia, constituye la propiedad que describe el brillo de la luz. Puede ser calculada como la suma de los componentes lineales rojo, verde y azul. El matiz, representa la propiedad

que permite clasificar a los colores como amarillo, rojo, violeta.

La saturación, constituye el grado de pureza del color; es decir, la medida en que el color es libre de luz blanca, dilución o distorsiones. Representa la propiedad que permite distinguir el rojo del rosado.

Las propiedades matiz y saturación, antes mencionadas, se emplean en conjunto para hacer referencia a la crominancia del color, la cual puede ser descrita como la propiedad relacionada con el cómo las personas piensan el color de la luz [6].

Entre los modelos de color que incluyen luminancia y crominancia se encuentra el YIQ. En este modelo, Y contiene la luminancia, o intensidad, mientras que I y Q contienen la información del color que está desacoplada de la luminancia. El modelo YIQ fue diseñado tomando en consideración las características del sistema visual humano, en especial la mayor sensibilidad a los cambios en la luminancia que a los de matiz o saturación. Básicamente, YIQ es una recodificación del modelo RGB [7]. La conversión de RGB a YIQ se hace mediante la siguiente transformación lineal

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ 0,596 & -0,275 & -0,321 \\ 0,212 & -0,523 & 0,311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

3. Ruido en imágenes digitales

El ruido representa información no deseada que deteriora la imagen. El ruido se considera un proceso $\sigma_{n_{i,j}}$ que afecta a la imagen adquirida s y no es parte de la escena original. Usando el modelo de ruido aditivo, este proceso se puede escribir como:

$$g_{i,j} = s_{i,j} + \sigma_{n_{i,j}}$$

El ruido de una imagen digital puede provenir de diversas fuentes. El proceso de adquisición de imágenes digitales convierte las señales ópticas en señales eléctricas y luego en señales digitales, y representa uno de los procesos por los que el ruido se introduce en las imágenes digitales.

El ruido $\sigma_{n_{i,j}}$ puede ser modelado, ya sea por un histograma o una función de densidad de probabilidad que se superpone a la función de densidad de probabilidad de la imagen original s . Uno de los tipos más comunes de ruido es el ruido gaussiano. Este tipo de ruido tiene una función de densidad de probabilidad normal:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$$

En el caso del ruido Gaussiano, es causado por fuentes naturales como vibración térmica de los átomos y la radiación de naturaleza discreta de los objetos calientes. También se le denomina ruido electrónico porque surge de detectores y amplificadores [8].

Una forma de calcular el nivel de ruido en una imagen es el método de Estimación Rápido de Ruido [9], en el cual es necesario realizar una convolución de la imagen mediante el operador Laplaciano

$$N = \begin{bmatrix} 1 & -2 & -1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

y luego se calcula la desviación estándar del ruido mediante la fórmula

$$\sigma_{n_{i,j}} = \sqrt{\frac{\pi}{2}} \frac{1}{6(W-2)(H-2)} \sum_x |x_{i,j} * N|$$

Donde W y H son las dimensiones de alto y ancho de la imagen.

4. Procesamiento de imágenes a color

El procesamiento digital de imágenes implica la modificación de datos digitales para la mejora de las calidades de imagen con ayuda del computador. Por medio del mencionado tratamiento se ayuda a maximizar la claridad y la nitidez de la imagen. El procedimiento que sustenta el procesamiento digital de imágenes es simple. La imagen digital se da como datos de entrada a un computador y éste se programa para modificar esos datos con ayuda de una ecuación, o grupo de ecuaciones y luego

almacenar los valores resultantes de los cómputos para cada pixel.

Los valores resultantes conforman una nueva imagen que puede ser mostrada por pantalla (u otro dispositivo de salida) o guardada en el formato deseado. Para mejorar determinadas características de los datos y para eliminar el ruido de la imagen, los datos digitales se someten a diferentes operaciones de tratamiento de imágenes. El procesamiento de imágenes implica cambiar la calidad de una imagen con el fin de: a) Mejorar la información gráfica de una imagen para la interpretación humana, b) Representar la imagen de una forma adecuada para su percepción por parte de una máquina [10].

Es de hacer notar, que el criterio a considerar en el problema de reducción de ruido depende del tipo de ruido que afecta a la imagen. Al respecto existen diversos tipos de algoritmos que hacen posible la reducción del ruido. Entre ellos se pueden destacar los filtros Montecarlo Secuenciales, donde se utiliza para realzar las características de brillo y contraste de una imagen, reducir su contenido de ruido, o agudizar o intensificar detalles presentes en ella [11].

El filtrado de imágenes a color se puede realizar de dos maneras. Un primer método, filtrado de los colores primarios (R, G y B) por separado y, el segundo método, filtrado unicamente de la luminancia de la imagen. El método de interés es el segundo y consiste en transformar la imagen RGB a YIQ para luego filtrar la capa Y (luminancia) sin procesar las otras dos. Posteriormente, la capa Y' filtrada se combina con las capas I y Q para reensamblar la imagen a color [12]. Una representación del método se muestra en la Figura 1.

La ventaja principal del modelo YIQ en el procesamiento de imágenes es que la luminancia (Y) y la información del color (I y Q) están separadas, de modo que la luminancia de una imagen puede procesarse sin afectar el color [7]. Además, filtrar exclusivamente la luminancia de la imagen implica una mayor eficiencia computacional.

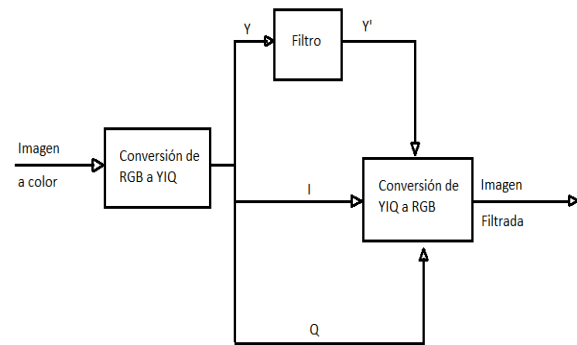


Figura 1: Filtrado unicamente de la luminancia de la imagen

5. Filtros no lineales

5.1. Filtro de diferencia dividida

El filtro de diferencia dividida de primer orden (FDD) se emplea para ecuaciones no lineales de tiempo discreto, considerando que los vectores de ruido son procesos Gaussianos no relacionados. Como punto de partida de la derivación de la diferencia dividida de primer orden se considera como un supuesto la estructura básica del filtro de Kalman [13].

El vector de estado predicho \hat{x}_{k+1} y la covarianza de estado predicho P'_{k+1} son determinados por

$$S_k = chol(P_k) \quad \hat{x}'_{k+1} = f_k(\hat{x}_k) \\ P'_{k+1} = F(\hat{x}'_k, S_k, d)F^T(\hat{x}'_k, S_k, d) + Q$$

Se considera en primer lugar la factorización Cholesky cuadrada

$$S_k = chol(P'_k)$$

donde

$$F(\hat{x}'_k, S_k, d) = \{F_{j,i}(\hat{x}'_k, S_k, d)\} = \\ \frac{\{f_{j,k}(\hat{x}'_k + ds_{x,i}) - f_{j,k}(\hat{x}'_k - ds_{x,i})\}}{2d}$$

Las funciones f_k y h_k se aproximan usando la interpolación polinomial de Stirling de primer orden.

S_k es la descomposición Cholesky de la matriz de la covarianza P_k , d es el tamaño de paso.

Se calcula el vector de observaciones predicho \hat{y}'_k y su covarianza predicha

$$\hat{y}'_k = h_k(\hat{x}'_k)$$

$$P'_{y,k} = H(\hat{x}'_k, S_k, d)H^T(\hat{x}'_k, S_k, d) + R_k$$

$$H(\hat{x}'_k, S_k, d) = \{H_{j,i}(\hat{x}'_k, S_k, d)\} = \frac{\{h_{j,k}(\hat{x}'_k + ds_{x,i}) - h_{j,k}(\hat{x}'_k - ds_{x,i})\}}{2d}$$

La matriz de correlación cruzada se determina por

$$P'_{x,y} = S_k H^T(\hat{x}'_k, S_k, d)$$

Así, la media y la covarianza actualizada es dada por:

$$K_k = P'_{x,y}((P'_{x,y})^{-1})$$

$$\hat{x}_k = \hat{x}'_k + K_k(y_k - \hat{y}'_k)$$

$$P_k = P'_k - K_k P'_{y,k} K_k^T$$

5.2. Filtro Adaptativo de diferencia centrada

El filtro adaptativo de diferencia centrada (FADC) se utiliza para la estimación de estados en sistemas no lineales con un proceso aditivo y la medición del ruido donde el conocimiento previo de la medida estadística del ruido no está disponible [14]. Considérese la ecuación de medición discreta no lineal siguiente

$$y_k = h_d(x_k) + v_k$$

donde, z_k es el vector de medida discreta de orden m y v_k los valores del ruido con covarianza R_k . El algoritmo procede como sigue:

Se determina la factorización Cholesky de P_{k-1} de la siguiente manera $S_{k-1} = chol(P_{k-1})$.

Se determina el vector de estados y se calcula su covarianza predicha

$$\hat{x}_k^- = f_d(\hat{x}_{k-1}^-)$$

$$P_k^- = S_{\hat{x}x} S_{\hat{x}x}^T$$

donde

$$S_{\hat{x}x}(i, j) = (f_{di}(\hat{x}_{k-1} + s_{k-1,j}) - f_{di}(\hat{x}_{k-1} - s_{k-1,j}))/2$$

y $s_{k-1,j}$ es la j -ésima columna de S_{k-1}
Se calcula el factor Cholesky de P_k^- :

$$S_{k-1}^- = chol(P_{k-1}^-)$$

Se determina la observación (medida) como

$$\hat{y}_k^- = h_d(\hat{x}_k^-)$$

Primera fase de corrección: se calcula la covarianza

$$P_{yy1} = (S_{y\hat{x}^-} S_{y\hat{x}^-}^T) + \hat{R}_{(k-1)m}$$

donde

$$S_{z\hat{x}^-}(i, j) = (h_{di}(\hat{x}_k^- + s_{k,j}^-) - h_{di}(\hat{x}_k^- - s_{k,j}^-))/2$$

y $s_{k,j}$ es la j -ésima columna de S_k^-

Se calcula la covarianza cruzada entre el estado predicho y el observado (medido)

$$P_{xy} = S_k^- S_{y\hat{x}^-}^T$$

Se calcula la ganancia de Kalman

$$K_{k1} = P_{xy} P_{yy1}^{-1}$$

Se estima el estado como

$$\hat{x}_{k1} = \hat{x}_k^- + K_{k1}(y_k - \hat{y}_k^-)$$

Se calcula el error de covarianza de los estados estimados

$$P_{k1} = (S_k^- - K_{k1} S_{y\hat{x}^-})(S_k^- - K_{k1} S_{y\hat{x}^-})^T + K_{k1} \hat{R}_{(k-1)m} K_{k1}^T$$

Se determina el Factor Cholesky de P_{k1}

$$S_{k1} = chol(P_{k1})$$

Adaptación de R_k : estima las observaciones como

$$\hat{y}_k^+ = h_d(\hat{x}_{k1})$$

Se encuentran los residuos

$$res_k = y_k - \hat{y}_k^+$$

Se calcula la covarianza de los residuos

$$P_{res} = \frac{1}{k_w} \sum_{i=k_w+1}^k (res_i)(res_i)^T$$

Se estima R_k

$$\hat{R}_k = P_{res} + S_{y\hat{x}} S_{y\hat{x}}^T$$

donde

$$S_{z\hat{x}}(i, j) = (h_{di}(\hat{x}_{k1} + s_{k1,j}) - h_{di}(\hat{x}_{k1} + s_{k1,j}))/2$$

y $s_{k1,j}$ es la j -ésima columna de S_{k1}

Se estima R_k modulada

$$\hat{R}_{km} = (1 - \varsigma)\hat{R}_{(k-1)m} + \varsigma\hat{R}_k$$

donde ς es el factor de modulación. Segunda fase de corrección: se calcula la correlación final de actualización

$$P_{yy} = (S_{y\hat{x}} - S_{y\hat{x}}^T) + \hat{R}_{km}$$

Se calcula la ganancia final de Kalman

$$K_k = P_{yx} P_{yy}^{-1}$$

Se calcula el estado final estimado

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - \hat{y}_k^-)$$

Se calcula la covarianza final de los errores de los estados estimados

$$P_k = (S_k^- - K_k S_{z\hat{x}})(S_k^- - K_k S_{y\hat{x}})^T + K_k \hat{R}_{km} K_k^T$$

Se determina el factor Cholesky de P_k

$$S_k = chol(P_k)$$

El factor de modulación que aquí se presenta proporciona una compensación entre la suavidad y la velocidad de la adaptación. Se puede elegir el valor de ς entre 0 y 1 .

5.3. Filtro de Kalman Cubature

El Filtro de Kalman de Cubature emplea una regla cubature esférica radial de tercer grado para calcular numéricamente las integrales que se encuentran en problemas de filtrado no lineal [15]. La forma aditiva del algoritmo del filtro de Kalman Cubature es como sigue

Etapa de predicción:

Se forman los sigma puntos

$$x_{k-1}^{(i)} = m_{k-1} + \sqrt{P_{k-1}} \xi^{(i)}, i = 1, \dots, 2n$$

donde los sigma puntos se definen como

$$\xi^{(i)} = \begin{cases} \sqrt{n}e_i, & i = 1, \dots, n \\ -\sqrt{n}e_{i-n}, & i = n+1, \dots, 2n. \end{cases}$$

Se propagan los sigma puntos a través del modelo dinámico

$$\hat{x}_k^{(i)} = f(\hat{x}_{k-1}^{(i)}), i = 1, \dots, 2n.$$

Se calcula la media m_k^- y la covarianza P_k^- predichas

$$m_k^- = \frac{1}{2n} \sum_{i=1}^{2n} \hat{x}_k^{(i)}$$

$$P_k^- = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{x}_k^{(i)} - m_k^-)(\hat{x}_k^{(i)} - m_k^-)^T + Q_{k-1}$$

Etapa de actualización: Se forman los sigma puntos

$$x_{k-1}^{-(i)} = m_k + \sqrt{P_k} \xi^{(i)}, i = 1, \dots, 2n$$

Se propagan los sigma puntos a través del modelo de las observaciones

$$\hat{y}_k^{(i)} = h(\hat{x}_k^{-(i)}), i = 1, \dots, 2n.$$

Se calcula la media μ_k y la covarianza S_k de las observaciones, y la covarianza cruzada de los estados y las observaciones C_k

$$\begin{aligned} \mu_k &= \frac{1}{2n} \sum_{i=1}^{2n} \hat{y}_k^{(i)} \\ S_k &= \frac{1}{2n} \sum_{i=1}^{2n} (\hat{y}_k^{(i)} - \mu_k)(\hat{y}_k^{(i)} - \mu_k)^T + R_k \\ C_k &= \frac{1}{2n} \sum_{i=1}^{2n} (\hat{x}_k^{(i)} - m_k^-)(\hat{y}_k^{(i)} - \mu_k)^T \end{aligned}$$

Se calcula la ganancia de Kalman K_k y la media de los estados filtrados m_k y la covarianza P_k , condicionada a las observaciones y_k

$$\begin{aligned} K_k &= C_k S_k^{-1} \\ m_k &= m_k^- + K_k [y_k - \mu_k] \\ P_k &= P_k^- - K_k S_k K_k^T \end{aligned}$$

6. Medidas de bondad de ajuste

Para comparar la imagen original con la imagen filtrada se emplean los siguientes parámetros:

El error absoluto medio (EAM) definida por

$$EAM = \frac{1}{N} \sum_{j=1}^N |x_{i,j}(k) - \hat{x}_{i,j}(k)|$$

donde N es el número de ejecuciones de cada filtro, j indica la j -ésima ejecución de simulación, e i representa la i -ésima componente del vector de estados $x(k)$ y su actual vector estimado $\hat{x}(k)$.

El pico de relación señal ruido (PSNR), constituye la relación entre la potencia máxima posible de una señal y la potencia de distorsión de ruido que afecta a la calidad de su representación.

La representación matemática del PSRN es la siguiente:

$$PSRN = 10 \log_{10} \left(\frac{(MAX_f)^2}{EMC} \right)$$

donde EMC es el error de la media cuadrática y MAX_f es el máximo valor que puede asumir cada pixel en la imagen original. Para el caso de las imágenes en el modelo YIQ, el máximo valor que tomar los pixeles en la capa Y es 255.

7. Procedimientos y Resultados

En este trabajo se presenta la restauración de píxeles provenientes de imágenes con ruido artificial. Se implementaron tres algoritmos: FDD, FADC y FKC, para estimar el vector de estados. Los algoritmos fueron implementados en el ambiente de programación Octave en un Intel Core i7, 2.8 GHz bajo Windows 8 de 64 bits. Se utilizó el modelo definido por [2] para la eliminación de ruido durante el procesamiento de imágenes.

Se realizaron pruebas con dos imágenes (imagen 1 e imagen 2) en formato JPG tomadas de internet con dimensiones 240x210 y 500x366, respectivamente.

Inicialmente se consideró el filtrado recorriendo la matriz de la capa Y (luminancia) solo en una dirección. En la imagen se consideró un ruido artificial con una varianza de $\sigma = 30$, para inicializar los filtros se utilizaron los siguientes valores a priori:

- Para el FDD $\hat{x}_0^+ = 255$, $P_0^+ = 0.1$, $Q_t = 0.15$, $R_t = 0.2$ con un tamaño de paso $h = 0.0001$.
- Para el FADC $\hat{x}_0^+ = 255$, $P_0^+ = 1$, $Q_t = 0.7$, $R_t = 0.7$, $\alpha = 0,01$, $R_{km} = 0,25$.
- Para el FKC $\hat{x}_0^+ = 255$, $P_0^+ = 0,05$, $R_t = 0,1$

Los resultados obtenidos permiten constatar que los filtros hacen un ajuste aceptable si se compara con la imagen original. En las Figuras 2 y 3 se puede observar un buen ajuste, sin embargo se aprecia que existe trazas horizontales y verticales en la imagen filtrada que no son propias de la imagen original.

Filtro	EAM	PSNR	Tiempo (s)
FKC	0,1213	17,0342	117,09
FDD	0,0667	21,8248	4,76
FADC	0,0652	22,1053	8,37

Tabla 1: Resultados para imagen 1. Filtrado horizontal.

Filtro	EAM	PSNR	Tiempo (s)
FKC	0,1216	17,0839	133,01
FDD	0,0665	21,8239	5,39
FADC	0,0654	22,0632	9,43

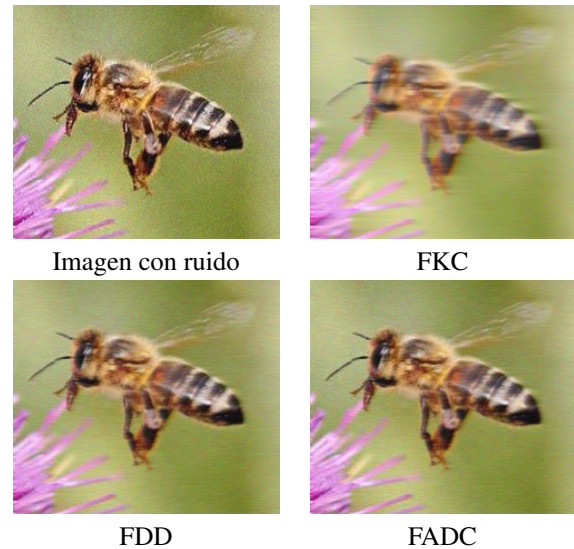
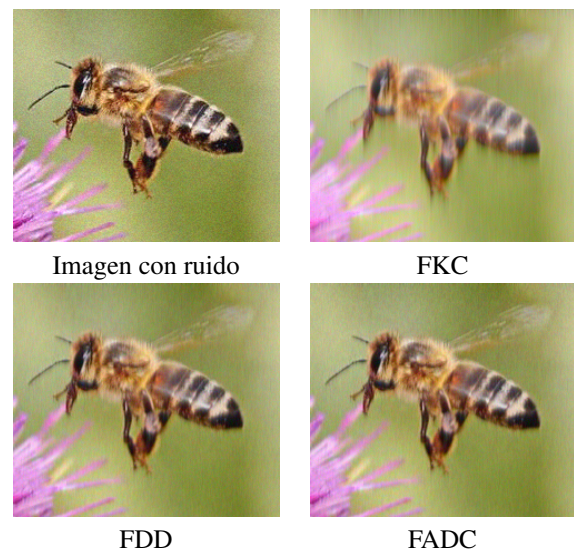
Tabla 2: Resultados para imagen 1. Filtrado vertical.

En los Tablas 1 y 2 se encuentran los valores asociados a los parámetros de rendimiento que se utilizaron a través de los cuales se confirma que los filtros hacen un ajuste aceptable de la imagen. Cabe recordar que el error absoluto medio permite comparar los valores reales de los píxeles de la imagen original con los de la imagen filtrada. El error es la cantidad que indica la diferencia entre los valores de los píxeles entre las imágenes. Por ello, mientras más cercano a cero sea el error se considera que la aproximación es mejor. Por su parte, el PSNR constituye una medida que hace posible comparar los efectos de los filtros en la calidad de la imagen. Al respecto, mientras más alto sea el valor del PSNR, se tiene una mejor reconstrucción de la imagen y por tanto, el filtro tiene un mejor desempeño.

Se realizó una prueba tomando en cuenta un doble recorrido. El proceso de filtrado se realizó haciendo un recorrido horizontal y otro vertical en la matriz de la capa Y (luminancia) de las imágenes tratadas. En vista que se obtienen dos matrices con distintos valores para cada píxel, se supone una buena aproximación al valor real, empleando el promediado para determinar un valor de tendencia más representativo; es decir, las matrices resultantes fueron adicionadas y los valores de los píxeles se promediaron para determinar la matriz Y filtrada final.

El procedimiento de filtrar en dos direcciones constituye un mecanismo al que se recurrió para mejorar los detalles que se presentaron al filtrar en

una sola dirección. Los filtros espaciales realizan operaciones teniendo en cuenta los píxeles vecinos además del píxel de interés, de modo que, cada píxel se sustituye por un promedio de sus píxeles vecinos [16]. Asimismo, la técnica de promediado de imágenes permite obtener una imagen como promedio de un conjunto de imágenes tomadas en las mismas condiciones generando que el nivel de error en los valores de los píxeles decrezca [17].

Figura 2: Resultados del filtrado horizontal con $\sigma = 30$.Figura 3: Resultados del filtrado vertical con $\sigma = 30$.

En la Figura 4 se presenta la imagen con ruido gaussiano artificial y las imágenes resultantes

del tratamiento empleando el procedimiento antes expuesto con cada uno de los filtros. Se puede evidenciar que los filtros implementados realizan un buen ajuste en comparación con la imagen original así como una notable reducción del ruido. Al compararlos visualmente, se puede notar que el FDD y el FADC reducen el ruido preservando la imagen con un bajo nivel de difuminado. Por su parte, el FKC (ver Tabla 3), a pesar de realizar una importante reducción del ruido tiende a difuminar la imagen. Los resultados obtenidos al evaluar los parámetros de rendimiento propuestos permiten evidenciar numéricamente lo advertido por medio de la observación de las imágenes filtradas.

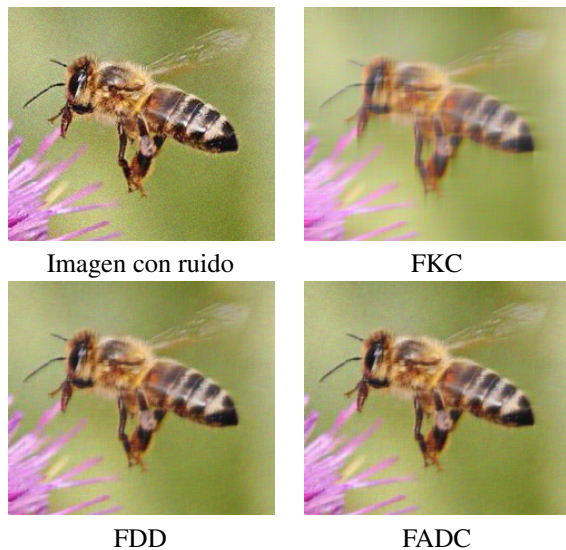


Figura 4: Resultados del filtrado con $\sigma = 30$.

Filtro	EAM	PSNR	Tiempo (s)
FKC	0,1204	17,3158	250,1
FDD	0,0647	22,2322	10,15
FADC	0,0635	22,4780	17,81

Tabla 3: Resultados para imagen 1 con $\sigma = 30$.

En el segundo caso se consideró un ruido artificial con varianza $\sigma = 50$. Además, se tomaron los siguientes valores a priori:

- Para el FDD $\hat{x}_0^+ = 255, P_0^+ = 0.1, Q_t = 0.25, R_t = 0.2$ con un tamaño de paso $h = 0.0001$.

- Para el FADC $\hat{x}_0^+ = 255, P_0^+ = 1, Q_t = 0.7, R_t = 0.9, \alpha = 0,01, R_{km} = 0,25$.
- Para el FKC $\hat{x}_0^+ = 255, P_0^+ = 0,04, R_t = 0,1$

En la Figura 5 se presenta la imagen con ruido gaussiano artificial y las imágenes resultantes del tratamiento con cada uno de los filtros. Se puede evidenciar que los filtros implementados realizan, al igual que la primera prueba, un buen ajuste en comparación con la imagen original así como una notable reducción del ruido. Los filtros FDD y FADC generan menor difuminado de la imagen tras el filtrado. Por otra parte, el FKC (ver Tabla 4) reduce de manera más significativa el nivel de ruido en comparación con los otros dos filtros pero es menos eficiente al ajustar los valores a la imagen original. Estas observaciones se pueden constatar en el Tabla 5 a partir de los valores que arrojaron los parámetros de rendimiento empleados.



Figura 5: Resultados del filtrado con $\sigma = 50$.

Filtro	EMC	PSNR	Tiempo (s)
FKC	0,1489	15,6534	1112,20
FDD	0,1012	18,6400	44,48
FADC	0,1011	18,6458	77,62

Tabla 4: Resultados para imagen 2 con $\sigma = 50$.

Para justificar lo mencionado en los apartes anteriores sobre la eficiencia en cuanto a la reducción del ruido por parte de los filtros, en el

Tabla 5 se presentan los niveles de ruido de la capa Y de las imágenes antes y después del proceso de filtrado.

Los valores en la tabla permiten verificar que el FKC realiza una mayor reducción del nivel de ruido, seguido por el FDD y en último lugar el FADC. No obstante, es importante resaltar que la notable reducción del ruido por parte de los filtros no garantiza que estos tengan un buen desempeño del todo. De allí que, la reducción del nivel de ruido debe ir a la par con un aceptable ajuste de los datos al comparar con la imagen de referencia.

	$\sigma = 30$	$\sigma = 50$
Imagen	2,8250	7,5213
FKC	0,2609	0,6670
FDD	0,6615	1,1287
FADC	0,8176	1,4360

Tabla 5: Niveles de ruido.

8. Conclusiones

A manera de conclusión es importante resaltar que en esta investigación se implementaron los algoritmos mencionados en el desarrollo de la misma, para estimar estados provenientes de imágenes con ruido artificial. Se utilizó la estructura básica del filtro de Kalman como punto de partida de la derivación de la diferencia dividida de primer orden. Las funciones que mapean los estados y las observaciones son aproximadas usando la interpolación polinomial de Stirling de primer orden. Se empleó la regla de curvatura esférica radial de tercer grado para calcular numéricamente las integrales que se encuentran en problemas de filtrado no lineal. Se realizó un doble recorrido (horizontal y vertical) en el filtrado de la matriz de luminancia (capa Y) lo que permitió una mejor aproximación al valor real, ya que se determinó un valor de tendencia más representativo para la matriz final Y filtrada, logrando así subsanar deficiencias propias del filtrado en una sola dirección. Este estudio demuestra que los filtros FDD, FADC y FKC son una alternativa válida para la restauración de imágenes. Se observó que al filtrar en dos direcciones se genera menor difuminado de la

imagen tras el filtrado en comparación con el filtrado de una dirección. Por otra parte, se observó que el FKC reduce de manera más significativa el nivel de ruido en comparación con los otros dos filtros. Adicionalmente, se estimó el error absoluto medio de los estados a posterior como medida de calidad de estimación de los filtros, observándose poca variabilidad entre ellos. Asimismo, se estimó el PSNR para verificar la calidad del ajuste de la imagen, observándose valores más altos en los filtros FDD y FADC implicando un mejor ajuste de las imágenes filtradas con respecto a la imagen original. En cuanto al tiempo de ejecución de los algoritmos se obtuvo diferencia significativa entre los filtros, siendo el DDF más eficiente que los filtros restantes.

Referencias

- [1] Knut Mørken. Numerical algorithms and digital representation, 2013. Department of Mathematics, Centre of Mathematics for Applications, University of Oslo.
- [2] Kelvin Solarte, Luis Sánchez y Joan Ordonez. Reducción de ruido en imágenes satelitales usando métodos montecarlo secuencial. *Revista Ingeniería UC*, 21(2):35–42, 2014.
- [3] Netzer Moriya. *Primer to Kalman Filtering: A Physicist's Perspective*. Nova Science Publishers Inc, New York, 2011.
- [4] Anna Saro Vijendran and Bobby Lukose. Fast and efficient method for image denoising. *International Journal of Engineering and Innovative Technology*, 3(1):200–207, 2013.
- [5] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482, 2000.
- [6] Versha Rani. A brief study of various noise model and filtering techniques. *Journal of global research in computer science*, 4(4):166–171, 2013.
- [7] R. Molina. Introducción al procesamiento y análisis de imágenes digitales, 1998. Universidad de Granada.
- [8] Ajay Boyat and Brijendra Kumar Joshi. Image denoising using wavelet transform and median filtering. In *Engineering (NUI CONE), 2013 Nirma University International Conference on*, pages 1–6. IEEE, 2013.
- [9] John Immerkaer. Fast noise variance estimation. *Computer vision and image understanding*, 64(2):300–302, 1996.
- [10] Douglas A. Kerr. Chromaticity and chrominance in color definition, 2010. Consultado: 15/02/2017.

https://dougkerr.net/Pumpkin/articles/Chromaticity_Chrominance.pdf.

- [11] B. Aldalur and M. Santamaría. Realce de imágenes: filtrado espacial. *Revista de teledetección*, 17:31–42, 2002.
- [12] S. Jamarayan, S. Esakkirajan, and T. Veerakumar. *Digital Image Processing*. Tata McGraw Hill Education Private Limited,, 2009.
- [13] Magnus Nørgaard, Niels K. Poulsen, and Ole Ravn. New developments in state estimation for nonlinear systems. *Automatica*, 36(11):1627 – 1638, 2000.
- [14] Manasi Das, Aritro Dey, Smita Sadhu, and Tapan Kumar Ghoshal. Adaptive central difference filter for nonlinear state estimation. *IET Science, Measurement & Technology*, 9(6):728–733, 2015.
- [15] Simo Särkkä. *Bayesian filtering and smoothing*. Cambridge University Press, 2013.
- [16] R. Gonzalez and R. Woods. *Procesamiento de Imágenes Digitales*. Addison-Wesley, 1998.
- [17] B. Chanda and D. D. Majumder. *Digital Image Processing and Analysis*. Prentice Hall of India, 2006.