

# Filtrado en Tiempo Real Usando el Estimador ABST

F. J. Gallegos-Funes<sup>1</sup>, V. I. Ponomaryov<sup>2</sup>, O. B. Pogrebnyak<sup>3</sup>

<sup>1</sup>Instituto Politécnico Nacional, ESIME, U. P. Zacatenco,  
Av. IPN s/n, Edificio Z, Acceso 3, 3er. Piso, Col. Lindavista, 07738, México D. F.

<sup>2</sup>Instituto Politécnico Nacional, ESIME, U. P. Culhuacan,  
Av. Santa Ana # 1000, Col. San Francisco Culhuacan, 04430, México D. F.

<sup>3</sup>Instituto Politécnico Nacional, Centro de Investigación en Computación  
Av. Juan de Dios Batiz s/n, Col. Lindavista, 07738, México D. F.

Teléfono (Fax): (5255) 5729-6000 Ext. 54622. Email: [fcogf@hotmail.com](mailto:fcogf@hotmail.com)

## RESUMEN

En este artículo, se presenta un filtro robusto basado en el estimador ABST (Ansari-Bradley-Siegel-Tukey) para la supresión de ruido impulsivo en aplicaciones de tiempo real. El filtro diseñado es llamado ABSTM-KNN (ABST M-Type K-Nearest Neighbor). Su esquema de filtrado usa una técnica similar al usado en el filtro KNN para proveer la preservación de contornos y detalles finos, y la combinación del estimador-M con el estimador-ABST provee la supresión de ruido impulsivo. La eficiencia del filtro propuesto fue evaluada por numerosas simulaciones en imágenes y secuencias de vídeo. También se muestra el tiempo de procesamiento del filtro ABSTM-KNN para demostrar que éste provee una solución para mejorar las transmisiones de vídeo.

### Palabras clave:

DSP TMS320C6701, Estimador ABST, Filtro ABSTM-KNN.

## INTRODUCCION

La pérdida de información en sistemas de visión por computadora es la principal razón para la restauración de imágenes y vídeo. La situación se vuelve más complicada debido a la presencia de ruido aleatorio. Generalmente, el ruido es una mezcla compleja de ruido Gaussiano aditivo debido al sensor eléctrico y al de amplificación. El ruido multiplicativo debido a interferencias electromagnéticas. El ruido impulsivo es causado por errores durante la adquisición o la transmisión entre canales [1, 2].

Por otro lado, también hay que considerar que los dispositivos para procesamiento de señales en tiempo real son muy importantes en un gran número de aplicaciones [2]. En aplicaciones de procesamiento de imágenes como son los casos del análisis de imágenes médicas, radar, percepción remota, telecomunicaciones, etc., es importante que los resultados estén en tiempo real para un análisis preciso para cada una de estas aplicaciones [2].

En sistemas de visión por computadora, los robots inteligentes responden a cambios ocurridos en su sensor de visión CCD (*Charge Coupled Device*) cuando ocurre una reacción en su entrada de visión. Por esta razón, las imágenes digitales adquiridas en

sistemas de visión de robots deberán tener buena resolución y estar libres de ruido. Otro caso en el cual es importante la realización en tiempo real es en el procesamiento de vídeo, su transmisión y recepción, en las cuales las secuencias de vídeo deberán ser procesadas en tiempo real mediante algún algoritmo que restaure sus características, por ejemplo, la restauración o supresión de ruido de las secuencias ocasionada durante transmisiones de TV y vídeo [2].

Recientemente, presentamos algoritmos robustos tipo RM, que proveen supresión de ruido impulsivo y una buena preservación de detalles finos [3]. Estos algoritmos usan los filtros MM-KNN y WM-KNN con diferentes funciones de influencia [4]. En este artículo, presentamos el novedoso filtro ABSTM-KNN el cual usa una técnica similar al filtro KNN [1] para proveer la preservación de detalles finos y la combinación de estimadores-M con el estimador-ABST provee la supresión de ruido impulsivo. Se presentan resultados experimentales del filtrado en imágenes y secuencias de vídeo. Diferentes simulaciones demostraron que el filtro propuesto provee mejor supresión de ruido y preservación de detalles en comparación a otros filtros propuestos en la literatura. Finalmente presentamos el tiempo de procesamiento del filtro propuesto para demostrar que éste provee una solución en tiempo real para mejorar las transmisiones de imágenes y vídeo.

## II. ESTIMADORES-R

Los estimadores-R son una clase de estimadores robustos no paramétricos basados en cálculos de rango. Considere dos muestras de test de rangos  $X_1, \dots, X_m$  y  $Y_1, \dots, Y_n$  como dos muestras con distribuciones  $H(x)$  y  $H(x + \Delta)$ , donde  $\Delta$  es corrimiento de localización desconocido. Sea  $R_i$  rango de  $X_i$  en la muestra de tamaño  $N = m + n$ . Un test de rangos de  $\Delta = 0$  para  $\Delta > 0$  está basado en el test estadístico [1, 5]

$$S = \frac{1}{m} \sum_{i=1}^m a_i(R_i) \quad (1)$$

donde los coeficientes  $a_i$  son generados por una función  $J(s)$  de la siguiente manera:

$$a_i = (m+n) \int_{(i-1)/(m+n)}^{i/(m+n)} J(s) ds \quad (2)$$

La función  $J(s)$  es simétrica en el censo de  $J(1-s) = -J(s)$  y satisface  $\int J(s) ds = 0$  y los coeficientes  $a_i$  satisfacen  $\sum_{i=1}^n a_i = 0$ . La selección de la función  $J(s)$  asume que la función de prueba  $f_0(X)$  es simétrica y realizando [5]

$$J(f_0(X)) = -c \frac{f'_0(X)}{f_0(X)}, \quad c \neq 0 \quad (3)$$

El estimador mediano es derivado de una función de distribución Laplaciana  $f_0(X) = \frac{1}{2} e^{-|x|}$  y una función

$$J(t) = \begin{cases} -1 & t < \frac{1}{2} \\ 1 & t > \frac{1}{2} \end{cases} \quad \text{dada por Ec. (3) y coeficientes}$$

$$a_i = \begin{cases} 1 & i = (N+1)/2 \\ 0 & \text{cualquier otro caso} \end{cases} \quad \text{dados por Ec. (2) [1, 5]:}$$

$$\theta_{\text{med}} = \begin{cases} \frac{1}{2} (X_{(n/2)} + X_{(n/2+1)}) & \text{para } n \text{ par} \\ X_{(n/2+1)} & \text{para } n \text{ impar} \end{cases} \quad (4)$$

donde  $X_{(j)}$  es el elemento de rango  $j$ .

El estimador Hodges-Lehmann con una función  $J(t) = \left| t - \frac{1}{2} \right|$  esta relacionado con el test de Wilcoxon

y una función de distribución logística  $f_0(x) = \frac{1}{1+e^{-x}}$  con coeficientes dados por  $a_i = \frac{2i-N-1}{2N}$  y su estimador de rango es el estimador-R Wilcoxon [1, 5]:

$$\theta_{\text{wh}} = \text{med} \left\{ \frac{1}{2} (X_{(i)} + X_{(j)}) \mid i, j = 1, \dots, N \right\} \quad (5)$$

donde  $X_{(i)}$ ,  $X_{(j)}$  son los elementos con rango  $i$  y  $j$  respectivamente.

La función Ansari-Bradley-Siegel-Tukey  $J(t) = \left| t - \frac{1}{2} \right| - \frac{1}{4}$ , tiene coeficientes dados por  $a_i = \frac{2i - \frac{3}{2}N - 1}{2N}$  y su correspondientes estimador-R es el siguiente [5]

$$\theta_{\text{ABST}} = \text{med} \left\{ \begin{array}{ll} X_{(i)} & i \leq \frac{N}{2} \\ \frac{1}{2} (X_{(i)} + X_{(N-i+1)}) & i > \frac{N}{2} \end{array} \right\} \quad (6)$$

el estimador ABST representa la utilización de los estimadores Ec. (4) y (5). Suponga valores  $X_i$  de longitud  $N=1, \dots, 9$ , los primeros cuatro rangos  $X_{(i)}$  están definidos de acuerdo a Ec. (4) y los siguientes cinco rangos están definidos como en el caso del estimador de Wilcoxon Ec. (5).

### III. ESTIMADOR-M

La forma generalizada del máximo de verosimilitud para los estimadores-M fue propuesta por Huber [5]. Su definición es dada por una función  $\rho(X) = \ln(f(X))$  conectada con la función de densidad de probabilidad  $f(X)$  para una muestra de datos  $X_i$ ,  $i = 1, \dots, N$ :

$$\theta = \arg \min_{\theta \in \Theta} \sum_{i=1}^N \rho(X_i - \theta) \quad (7)$$

La estimación para el parámetro de localización  $\theta$  puede ser encontrado mediante el cálculo de la derivada parcial de  $\rho$  (con respecto a  $\theta$ ) introduciendo la función  $\psi(X, \theta) = \frac{\partial}{\partial \theta} \rho(X, \theta)$  [1, 5]:

$$\sum_{i=1}^N \psi(X_i - \theta) = 0 \quad (8)$$

La solución de  $\theta$  para el estimador  $M$  tiene algunas limitantes: la desviación de la función  $\psi(X)$  llamada *winsorización* o de la desviación de  $(X_i - \theta)$  llamada *ensorización* o *trimming*. La técnica usada para el cálculo de la estimación  $M$  esta basada en el método iterativo de Newton [1, 5]:

$$\theta_M = \frac{\sum_{i=1}^N X_i \phi(X_i - \text{MED}\{X\})}{\sum_{i=1}^N 1_{[1-r, r]}(X_i - \text{MED}\{X\})} \quad (9)$$

donde  $\varphi$  es la función normalizada  $\psi$ :  $\psi(X) = X\varphi(X)$ . Es evidente que la fórmula (9) representa la media aritmética de  $\sum_{i=1}^n \psi(X_i - \text{MED}\{X\})$  evaluada en el intervalo  $[-r, r]$ .

La restricción más simple sobre el rango de  $\psi(X)$  es el límite del estimador-M de Huber [5]

$$\varphi_b(X) = \text{MIN}(b, \text{MAX}(X, -b)) = [X]_{-b}^b. \quad (10)$$

Otra vía para derivar la función  $\varphi(X)$  consiste en eliminar los valores lejanos (por peso) de una muestra de datos. Este paso es también llamado mínima estimación-M. Hampel [5] demostró que la función de influencia de corte es la mínima estimación robusta M. También usamos las funciones de influencia simple, Hampel, seno de Andrew, Tukey y Bernoulli [1, 5].

#### IV ESTIMADOR-ABST TIPO-M

Para mejorar las propiedades robustas de los estimadores R y M mediante el uso de la estimación R (6) consiste en un procedimiento similar a la media aritmética [3, 4]. De este modo, obtenemos la estimación combinada RM (ABSTM) siguiente:

$$\begin{aligned} \theta_{\text{ABSTM}} &= \\ &= \text{MED} \begin{cases} X, \varphi(X, -\text{MED}\{X\}) & i \leq \frac{N}{2} \\ \frac{1}{2}[X, \varphi(X, -\text{MED}\{X\}) + X, \varphi(X, -\text{MED}\{X\})] & i > \frac{N}{2} \end{cases} \end{aligned} \quad (11)$$

#### V FILTRO ABSTM-KNN

El filtro propuesto usa un cálculo iterativo derivado del estimador mediano. La estimación inicial usa el valor del pixel central de la ventana de filtrado para proveer la preservación de los detalles. Los estimadores-R y -M combinados [4] usan una función similar al filtro KNN [1] para proveer la supresión de ruido impulsivo [4]. El filtro diseñado es llamado Filtro ABSTM-KNN (Ansari-Bradley-Siegel-Tukey M-type K- Nearest Neighbor):

$$\begin{aligned} \theta_{\text{ABSTM-KNN}}^{(q)}(i, j) &= \\ &= \text{MED} \begin{cases} h^{(q)}(i + m, j + n), & i, j \leq \frac{N}{2} \\ \frac{h^{(q)}(i + m, j + n) + h^{(q)}(i + m_1, j + n_1)}{2}, & i > \frac{N}{2}, i \leq j \end{cases} \end{aligned} \quad (12)$$

donde  $h^{(q)}(i + m, j + n)$  y  $h^{(q)}(i + m_1, j + n_1)$  son conjuntos de  $K_{\text{close}}$  pixeles con peso de acuerdo a la

función de influencia usada [1, 4, 5] y están encerrados por valor en la ventana del filtro con la estimación del paso previo  $\theta_{\text{ABSTM-KNN}}^{(q-1)}(i, j)$ .  $\theta_{\text{ABSTM-KNN}}^{(0)}(i, j) = x(i, j)$  es el estimador inicial.  $x(i, j)$  es la imagen degradada,  $(2L+1)^2$  es el tamaño de la ventana del filtro,  $m, n, m_1, n_1 = -L \dots L$  son los límites de la ventana del filtro,  $q$  es el índice de la iteración actual y  $\theta_{\text{ABSTM-KNN}}^{(q)}(i, j)$  es la estimación en la iteración  $q$ . El algoritmo finaliza cuando  $\theta_{\text{ABSTM-KNN}}^{(q)}(i, j) = \theta_{\text{ABSTM-KNN}}^{(q-1)}(i, j)$ . Usualmente es necesario 2 ó 3 pasos para satisfacer esta condición.  $K_{\text{close}}(i, j)$  refleja la actividad local y la presencia de impulsos [4]

$$K_{\text{close}}(i, j) = K_{\min} + a\Sigma(x(i, j)) \leq K_{\max} \quad (13)$$

donde  $a$  controla la sensibilidad del filtro para una buena detección de detalles.  $K_{\min}$  es el número mínimo de vecinos necesarios para remover el ruido.  $K_{\max}$  es el número máximo de vecinos que restringe los contornos y los detalles suaves.  $\Sigma(x(i, j))$  es el detector de impulsos y es calculado como:

$$\begin{aligned} \Sigma(x(i, j)) &= \frac{\text{med}\{|x(i, j) - x(i + m, j + n)|\}}{\text{MAD}\{x(i, j)\}} + \\ &+ \frac{1}{2} \frac{\text{MAD}\{x(i, j)\}}{\text{med}\{x(i + k, j + l)\}} \end{aligned} \quad (14)$$

donde  $\text{med}\{x(i + k, j + l)\}$  es la mediana de los pixeles contenidos en la ventana del filtro,  $k, l = -L \dots L$  son los límites de la ventana y MAD es la mediana de las desviaciones absolutas de la mediana [1, 2, 5]:

$$\text{MAD}\{x(i, j)\} = \text{med}\{\text{med}\{x(i + k, j + l)\} - x(i + m, j + n)\} \quad (15)$$

#### VI RESULTADOS EXPERIMENTALES

El filtro ABSTM-KNN y otros filtros propuestos como comparativo fueron implementados en el DSP TMS 320C6701 de Texas Instruments [6]. El 'C6701 es un dispositivo de alto desempeño que realiza hasta 1 GFLOPS con una velocidad de 167 MHz. [6]. El filtro ABSTM-KNN con diferentes funciones de influencia fue evaluado y su desempeño fue comparado con diferentes filtros. El criterio usado para comparar el rendimiento de supresión de ruido fue el pico de la relación señal a ruido (PSNR) y para la preservación de los detalles el criterio usado fue el error absoluto medio (MAE) [1, 2]:

$$\text{PSNR} = 10 \cdot \log \left[ \frac{(255)^2}{\text{MSE}} \right] \text{ dB} \quad (16)$$

$$\text{MAE} = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |e(i, j) - \hat{x}(i, j)|, \quad (17)$$

donde:  $e(i, j)$  es la imagen original,  $\hat{x}(i, j)$  es la imagen restaurada,  $M, N$  es el tamaño de la imagen y

$\text{MSE} = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [e(i, j) - \hat{x}(i, j)]^2$  es el error cuadrático medio.

### VI.1 Restauración de Imágenes

Para determinar las propiedades de supresión de ruido impulsivo de diferentes filtros, la imagen "Peppers" (en niveles de gris de 256x256 píxeles), fue degradada con 20% de ruido impulsivo. El rendimiento PSNR y MAE del filtro ABSTM-KNN 3x3 con las funciones de influencia *simple*, *corte*, *Hampel*, *Andrews*, *Tukey* y *Bernoulli* fue comparado con el rendimiento de los filtros *mediano*, WM 3x3 (*weighted median*) [1, 2], FIRMH 5x5 (*FIR median hybrid*) [7], LUM 5x5 (*Lower-upper-middle*) [8], TSM 3x3 (*tri-state median*) [9], ACWM 3x3 (*adaptive center weighted median*) [10], ROM 3x3 (*rank order mean*) [11], MMEM 3x3 (*minimum-maximum exclusive mean*) [12] y LMMSE 3x3 (*Local Linear Minimum Mean Square Error*) [13]. Los filtros usados para comparar nuestro método fueron computados de acuerdo a sus referencias. La razón de haber escogido estos filtros es que su rendimiento ha sido comparado con diferentes filtros conocidos y sus ventajas han sido demostradas. Los valores PSNR y MAE y el Tiempo de Procesamiento en segundos son mostrados en la Tabla 1.

Tabla 1. PSNR en dB, MAE y Tiempo de Procesamiento para la imagen "Peppers".

| Algoritmo           | PSNR dB | MAE   | Tiempo   |
|---------------------|---------|-------|----------|
| Mediano 3x3         | 22.83   | 11.24 | 0.014634 |
| WM 3x3              | 24.68   | 7.81  | 0.020341 |
| FIRMH 5x5           | 21.96   | 11.17 | 0.060991 |
| LUM 5x5             | 23.65   | 11.37 | 0.475757 |
| TSM 3x3             | 22.96   | 11.38 | 0.054759 |
| ACWM 3x3            | 25.18   | 9.21  | 0.229951 |
| ROM 3x3             | 25.04   | 9.62  | 0.075008 |
| MMEM 3x3            | 24.40   | 9.67  | 0.040618 |
| LMMSE 3x3           | 24.75   | 9.70  | 0.075140 |
| ABSTM-KNN Simple    | 25.55   | 7.72  | 0.063415 |
| ABSTM-KNN Corte     | 25.85   | 7.55  | 0.063876 |
| ABSTM-KNN Hampel    | 25.62   | 7.75  | 0.063787 |
| ABSTM-KNN Andrew    | 25.95   | 7.57  | 0.074301 |
| ABSTM-KNN Tukey     | 25.51   | 7.75  | 0.063151 |
| ABSTM-KNN Bernoulli | 25.46   | 7.75  | 0.067383 |

De los resultados PSNR y MAE de la Tabla 1, se observa que el filtro ABSTM-KNN suprime mejor el ruido impulsivo y preserva mejor los contornos y los detalles finos en comparación con los filtros utilizados como comparativo. Con lo relacionado al tiempo de procesamiento el filtro propuesto tiene el mayor tiempo de procesamiento a excepción de los filtros LUM, ACWM, ROM y LMMSE. Todos los tiempos incluyen el tiempo de adquisición, procesamiento y almacenamiento. En el caso del filtro ROM, éste utiliza datos de entrenamiento y el tiempo utilizado para derivar los coeficientes utilizados en su esquema de filtrado es de 0.035 seg. aproximadamente.

El filtro ABSTM-KNN con diferentes funciones de influencia puede procesar de 13 a 15 imágenes de 256x256 píxeles en 1 segundo proporcionando buena supresión de ruido impulsivo y preservación de detalles para aplicaciones en tiempo real.

### VI.2 Restauración de Secuencias de Video

La mayoría de las secuencias de video tienen alta correlación entre imágenes consecutivas, es claro que el filtrado en 3-D es mas eficiente que el filtrado en 2-D en términos de PSNR [14]. El filtrado en 3-D requiere un gran cálculo y tiempo de procesamiento. En algunas aplicaciones como visión por computadora o análisis de imágenes médicas, las imágenes consecutivas de una secuencia de video no tienen ninguna correlación. Por estas razones usamos filtrado en 2-D en el caso de secuencias de video.

Para este propósito, usamos una secuencia de video QCIF (*Quarter Common Intermediate Format*) [15]. Este formato usa 176x144 píxeles por imagen con una velocidad de 15 a 30 imágenes por segundo. Para la codificación y decodificación de este formato usamos el estándar H263 y la biblioteca de "image/video processing" [16, 17]. La secuencia de video de prueba "Carphone" fue degradada con 20% de ruido impulsivo. El rendimiento obtenido con los filtros WM, TSM, ACWM, MMEM y LMMSE fue comparado con el filtro ABSTM-KNN 3x3 con diferentes funciones de influencia. En la Tabla 2 se presenta el rendimiento PSNR, MAE y Tiempo de Procesamiento para diferentes filtros.

De la Tabla 2 se observa que el rendimiento PSNR y MAE del filtro propuesto es superior al de los filtros usados como comparativo. El tiempo de procesamiento del filtro propuesto es de casi 0.03 seg. por imagen, por lo que se puede decir que el filtro ABSTM-KNN puede procesar 33 imágenes QCIF en un segundo. El filtro ROM no fue utilizado debido a que utiliza datos de entrenamiento por lo que no es recomendable su utilización en aplicaciones en tiempo real. El filtro ACWM fue utilizado debido a que es el mejor filtro de comparativo (valores PSNR y MAE) pero su tiempo excede las especificaciones del formato QCIF.

Tabla 2. PSNR en dB, MAE y Tiempo de Procesamiento para una imagen de la secuencia de video "Carphone".

| Algoritmo           | PSNR dB | MAE   | Tiempo   |
|---------------------|---------|-------|----------|
| WM 3x3              | 23.83   | 9.80  | 0.009977 |
| TSM 3x3             | 21.53   | 13.24 | 0.023730 |
| ACWM 3x3            | 24.34   | 9.79  | 0.093865 |
| MMEM 3x3            | 23.60   | 11.11 | 0.018021 |
| LMMSE 3x3           | 24.15   | 10.04 | 0.034716 |
| ABSTM-KNN Simple    | 25.08   | 9.44  | 0.027151 |
| ABSTM-KNN Corte     | 25.07   | 9.44  | 0.027068 |
| ABSTM-KNN Hampel    | 24.79   | 9.66  | 0.027932 |
| ABSTM-KNN Andrew    | 25.08   | 9.48  | 0.029762 |
| ABSTM-KNN Tukey     | 24.78   | 9.62  | 0.026815 |
| ABSTM-KNN Bernoulli | 24.88   | 9.62  | 0.028440 |

En la Fig. 1 se muestran los resultados visuales en una imagen de la secuencia "Carphone". De esta figura se observa que el filtro ABSTM-KNN provee mejores resultados visuales tanto en supresión de ruido como en preservación de detalles en comparación con otros filtros.

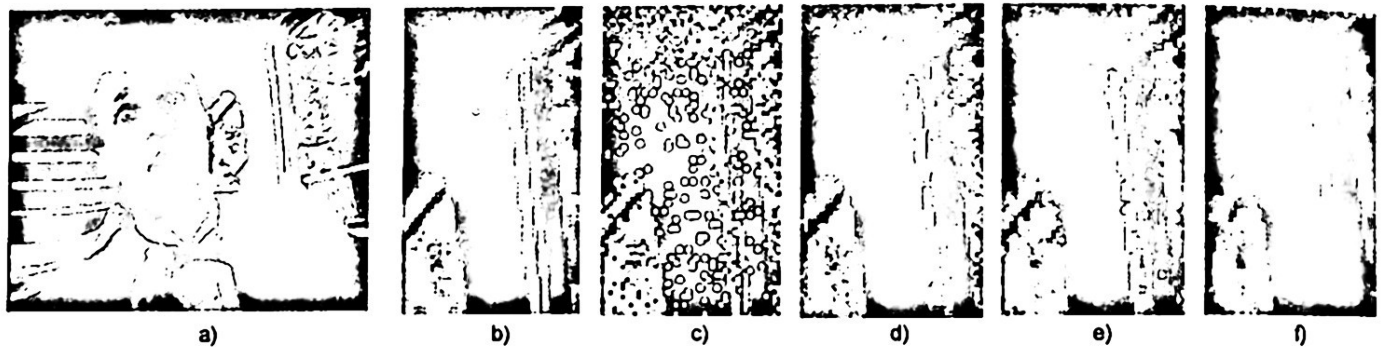


Figura 1. Resultados visuales en una imagen de la secuencia de video "Carphone". a) Imagen de video original, b) Acercamiento de la parte superior derecha de (a), c) Acercamiento de una parte de (a) degradada con 20% de ruido impulsivo, d) Acercamiento de una parte de (a) restaurada con el filtro ACWM, e) Acercamiento de una parte de (a) restaurada con el filtro LMMSE, f) Acercamiento de una parte de (a) restaurada con el filtro ABSTM-KNN (corte).

## REFERENCIAS

- [1] J. Astola, P. Kuosmanen, *Fundamentals of Nonlinear Digital Filtering*, New York, CRC Press, 1997.
- [2] A. Bovik, *Handbook of Image and Video Processing*, San Diego, Academic Press, 2000.
- [3] V. I. Ponomaryov, O. B. Pogrebnyak, "Novel robust RM filters for radar image preliminary processing," *Journal of Electronic Imaging*; vol. 8, no. 4, pp. 467-477, 1999.
- [4] F. J. Gallegos-Funes, V. Ponomaryov, S. Sadovnychiy, L. Nino, "Median M-type K-nearest neighbour (MMKNN) filter to remove impulse noise from corrupted images," *IEE Electronics Letters*; vol. 38, no. 15, pp. 786-787, 2002.
- [5] F. R. Hampel, E. M. Ronchetti, P. Rousseeuw, W. Stahel, *Robust Statistics. The approach based on influence function*, New York, Wiley, 1986.
- [6] Texas Instruments, *TMS320C6201/6701 Evaluation Module User's Guide*, Dallas, Texas Instruments Inc., 1998.
- [7] R. Wichman, J. Astola, P. Heinonen, Y. Neuvo, "FIR-median hybrid filters with excellent transient response in noisy conditions," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, pp. 2108-2117, Dec. 1990.
- [8] R. C. Hardie, C. G. Bonchelet, "LUM filters: a class of rank order based filters for smoothing and sharpening," *IEEE Trans. Signal Process.*, vol. 41, pp. 1061-1076, March 1993.
- [9] T. Chen, K. Ma, L. Chen, "Tri-State Median filter for image denoising," *IEEE Trans. Image Process.* vol. 8, no. 12, pp. 1834-1838, 1999.
- [10] T. Chen, H. R. Wu, "Adaptive impulse detection using center-weighted median filters," *IEEE Signal Process. Lett.*, vol. 8, no. 1, pp. 1-3, 2001.
- [11] E. Abreu, M. Lightstone, S. K. Mitra, K. Arakawa, "A new efficient approach for the removal of impulse noise from highly corrupted images," *IEEE Trans. Image Process.* vol. 5, no. 6, pp. 1012-1025, 1996.
- [12] H. Wei-Yu, L. Ja-Chen, "Minimum-maximum exclusive mean (MMEM) filter to remove impulse noise from highly corrupted images," *IEE Electronics Letters*, vol. 33, no. 2, pp. 124-125, 1997.
- [13] M. K. Özkan, M. I. Sezan, A. Murat, "Adaptive motion-compensated filtering of noisy image sequences," *IEEE Trans. Circuits and Syst. For Video Tech.* vol. 3, no. 4, pp. 277-290, 1993.
- [14] J. S. Kim, H. W. Park, "Adaptive 3-D median filtering for restoration of an image sequence corrupted by impulsive noise," *Signal Processing: Image Communication*, vol. 16, pp. 657-668, 2001.
- [15] P. E. Mattison, *Practical Digital Video with Programming Examples in C*, New York, Wiley, 1994.
- [16] G. Cote, B. Erol, M. Gallant, F. Kossentini, "H.263+: Video Coding at Low Bit Rates," *IEEE Trans. Circuits and Syst. Video Technol.* vol. 8 no. 7, pp. 849-866, 1998.
- [17] Texas Instruments, *TMS320C62x Image/Video Processing Library: Programmer's Reference*, Dallas, Texas Instruments Inc., 2002.

## VII. CONCLUSIONES

Se presenta el filtro robusto ABSTM-KNN para la supresión de ruido impulsivo con preservación de detalles para aplicaciones en tiempo real. Diferentes simulaciones en imágenes y secuencias corrompidas por ruido impulsivo demuestran que el filtro propuesto provee mejor supresión de ruido y preservación de detalles en comparación a otros filtros propuestos en las referencias. El filtro propuesto puede usarse en aplicaciones en tiempo real procesando de 13 a 15 imágenes de 256x256 píxeles en un segundo o secuencias de video QCIF.

A trabajo futuro se pretende la utilización del filtrado propuesto en aplicaciones de 3-D mejorando con ello la habilidad para suprimir ruido impulsivo y preservar los detalles finos.

AGRADECIMIENTOS. Los autores dan las gracias al Instituto Politécnico Nacional por los apoyos brindados.