

# Optimización por PSO de un sistema borroso para filtrado de ruido sal y pimienta

Autores: Manzo Renato José, Vizzarri Cristian  
Responsable de cátedra: Dr. Gerard Matias Fernando

Universidad Nacional del Litoral, Facultad de Ingeniería y Ciencias Hídricas, Santa Fe  
CP3000, AR

manzorenato@gmail.com, cristiaan3003@gmail.com,  
mgerard@santafe-conicet.gov.ar

**Resumen** El filtrado de ruido es una tarea importante en el preprocesamiento de imágenes digitales. Tal es así, que sin ello la tarea de extracción de características se torna difícil. En el presente trabajo se desarrolló un método de eliminación de ruido impulsivo, del tipo sal y pimienta, sobre imágenes en escala de grises. Para la creación de los filtros se utiliza lógica borrosa y optimización por enjambre de partículas. El rendimiento del método fue comparado con el filtro de mediana clásico sobre un conjunto de imágenes de prueba a distintas densidades de ruido. Los resultados experimentales probaron que el método propuesto presenta un rendimiento superior al filtrado de mediana en el rango de ruido 1% – 50% y no introduce desenfoque.

**Keywords:** Sal y pimienta, Procesamiento, Conjunto borroso, Optimización, PSO.

## 1 Introducción

En ciertas ocasiones y debido a factores externos, las imágenes digitales poseen información no deseada que degrada la imagen original. Uno de estos factores de degradación es conocido como ruido y es incorporado a la imagen por dos grandes causas: error en la adquisición de la imagen debido a defectos en los sensores o error en la transmisión de la imagen a través de los canales de comunicación.

La restauración de imágenes con degradación es una disciplina que lleva ya muchos años y resulta de vital importancia en el esquema de procesamiento de una imagen digital. No obstante para poder eliminar el ruido, primero se debe conocer el tipo de ruido que afecta la imagen. En particular, el ruido conocido como sal y pimienta provoca que píxeles de la imagen - en escala de gris - presenten una diferencia de intensidad marcada respecto a la de sus vecinos circundantes [2]. Su presencia en principio conduce a una degradación a nivel visual pero también puede afectar a otras tareas importantes en el esquema de procesamiento de una imagen digital como ser, por ejemplo, la detección de bordes.

Dentro de las técnicas clásicas de procesamiento existen distintos filtros para la reducción de este tipo de ruido. El filtrado de mediana es un filtro lineal que se aplica en el dominio espacial y es frecuentemente el más utilizado para este tipo de ruido debido a su simplicidad y buenos resultados. Al ser un filtrado espacial las operaciones se hacen directamente sobre los píxeles y no se requiere ninguna transformación de la imagen. Inicialmente se establece una ventana (kernel) de un determinado tamaño. Por ejemplo, una ventana con una vecindad de 3x3 píxeles donde en el centro se encuentra el píxel a procesar. Posteriormente se recorre la imagen completa, píxel a píxel, y se calcula la mediana de la vecindad utilizando como referencia la ventana. Así el valor de intensidad del píxel central es reemplazado por dicho valor. Como contraparte, este método introduce otra degradación no deseada a la imagen denominada *desenfoque*.

En los últimos años, en distintas áreas de aplicaciones, se han incrementado el uso de sistemas borrosos para resolver problemas. Desde aplicaciones del rango de uso hogareño como cámaras, grabadoras, lavarropas, microondas y hasta incluso el control de procesos industriales, entre otros. Sin embargo las funciones de membresía se podrían redefinir para producir mejores resultados, adaptándolas a las condiciones del problema [1].

Optimización por enjambre de partículas (PSO, por sus siglas en inglés) es un método de optimización metaheurístico bioinspirado, que permite realizar optimizaciones en problemas multidimensionales a partir de una población de soluciones candidatas (partículas). El proceso es guiado mediante una función objetivo que combina información acerca de la mejor solución encontrada por cada partícula y por todo el enjambre durante toda la búsqueda [6].

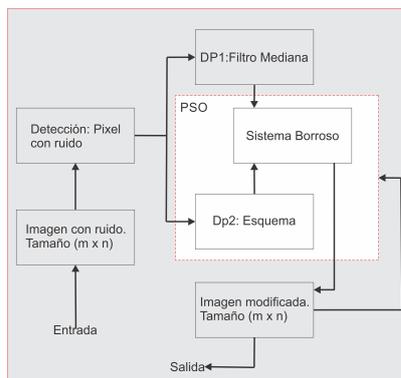
Debido a las características de esta técnica y siguiendo los lineamientos *descriptos* en otro trabajo [5], se propone una estrategia de optimización mediante PSO para las funciones de membresía de un algoritmo basado en lógica difusa empleado en la restauración de las imágenes. La estructura del presente trabajo se define de la siguiente manera: en la Sección 2 se expone la metodología utilizada para el desarrollo del trabajo. En la Sección 3 se aborda el marco experimental. En la Sección 4 se discuten los resultados experimentales. Finalmente, en la Sección 5 se exponen las conclusiones y trabajos futuros.

## 2 Metodología

### 2.1 Esquema de corrección de ruido

El proceso de corrección de ruido esquematizado en la Fig. 1, está compuesto de cuatro etapas que se aplican en el dominio espacial. Como puede apreciarse, el proceso emplea un sistema borroso  $\varphi(\cdot)$  para el cálculo de los nuevos valores de intensidad de aquellos píxeles candidatos <sup>1</sup>, utilizando un conjunto de funciones de membresía predefinido. El recuadro blanco en la Fig. 1 indica en que etapa del proceso se aplica PSO para optimizar las funciones de membresía del sistema borroso (ver Sección 2.3).

<sup>1</sup> No se tienen en cuenta los bordes de la imagen en el procesamiento.



**Fig. 1.** Esquema de corrección de ruido. El recuadro blanco indica la etapa del proceso donde se realiza la optimización con PSO.

La primera etapa consiste en la detección de los píxeles candidatos. Estos son aquellos que en función de su valor de intensidad pueden ser considerados píxeles con ruido de tipo sal o pimienta (0 o 255). De ellos, y en función de la dimensión de la imagen, se guarda un registro de filas y columnas  $(i, j)$ . Solo estos serán procesados y alterados en la imagen final de salida.

En la segunda etapa se establece una ventana alrededor del píxel candidato de  $3 \times 3$  píxeles. Esto se debe a que este método utiliza operadores diferenciales discretos  $[i]$  que tienen en cuenta las direcciones del entorno del píxel a corregir. La Fig. 2 presenta un ejemplo de la ventana seleccionada en torno a un píxel candidato. Los nombres que reciben los píxeles dentro de la ventana se indican en Fig. 2(a). La Fig. 2(b) muestra un ejemplo de un píxel con ruido de sal y los vecinos que abarca dicha ventana.

En la tercera etapa se procede a calcular el nuevo valor del píxel candidato  $(i, j)$ . Para proceder, se utiliza la metodología propuesta [4] donde, en función de la ventana establecida y de las direcciones del entorno, se arman conjuntos para el cálculo del nuevo valor. En primer lugar se definen cuatro conjuntos a partir de los ocho píxeles vecinos al píxel candidato  $(i, j)$ :  $SS1 = [N, S]$ ,  $SS2 = [O, E]$ ,  $SS3 = [NE, SO]$  y  $SS4 = [NO, SE]$ . De acuerdo a la Fig. 2(b) esos conjuntos toman los valores  $SS1 = [175, 165]$ ,  $SS2 = [201, 179]$ ,  $SS3 = [179, 202]$  y  $SS4 = [184, 200]$ . Estos son procesados con el sistema borroso  $\varphi(\cdot)$  de a pares y luego sus salidas utilizadas para formar nuevos conjuntos. Así  $NS1 = [\varphi(SS1), \varphi(SS2)]$  y  $NS2 = [\varphi(SS3), \varphi(SS4)]$ . A continuación, ambos conjuntos son procesados nuevamente con el sistema borroso para obtener  $DP1 = \varphi(NS1, NS2)$ . En paralelo se calcula  $DP2$  aplicando el filtro de mediana clásico a la ventana asociada al píxel candidato  $(i, j)$ . Finalmente se determina el valor corregido del píxel  $(i, j)$  como  $s = \varphi(DP1, DP2)$ .

Finalmente, el valor del píxel candidato  $(i, j)$  se reemplaza por el valor de salida del sistema borroso en la cuarta etapa. Este proceso se realiza iterativamente hasta modificar todos los píxeles candidatos  $(i, j)$ . Cabe aclarar que la aplicación de PSO no forma parte del trabajo original y será explicado en detalle en la Sección 2.3.

4 Manzo R.J., Vizzarri C., Gerard M.F.

NO	N	NE
O	<b>Ruido</b>	E
SO	S	SE

(a)

184	175	179
201	<b>255</b>	179
202	165	200

(b)

**Fig. 2.** Identificación de los píxeles en una ventana de 3x3. a) Vecindad de un píxel ruidoso. b) Ejemplo de un píxel con ruido de tipo sal (centro).

## 2.2 Sistema borroso

El sistema borroso empleado en este trabajo es del tipo Mandami, con dos entradas y una salida. La composición de reglas se realiza mediante minimización (AND) y el proceso de defuzzificación mediante el método del centroide [5]. Como se puede observar en la Fig. 3 las funciones de membresía utilizadas son del tipo gaussiana, proveen una transición más suave y se encuentran definidas en el intervalo  $[0, 1]$ . El conjunto de entrada está definido por las funciones *Dark*, *Avg* y *Light*, tanto para la entrada *P1* como para la entrada *P2*. En cambio, las funciones de membresía del conjunto de salida son *Dark*, *DmA*, *Avg*, *AmL* y *Light*. A partir de estos conjuntos es posible construir un total de 9 reglas. Por ejemplo, si *P1* y *P2* toman valor *Dark*, la decisión será *Dark*. En cambio, si *P2* es *Light* la nueva decisión será *Avg*. Las Fig. 3(a) y 3(b) presentan las funciones de membresía para los conjuntos de entrada y salida, respectivamente.

La función de membresía empleada en el sistema difuso para todos los conjuntos está caracterizada por los parámetros  $(\sigma_1, c_1)$  y  $(\sigma_2, c_2)$ , y se define como:

$$f(\mu, \sigma, c) = e^{-\frac{(\mu-c)^2}{2\sigma^2}}. \quad (1)$$

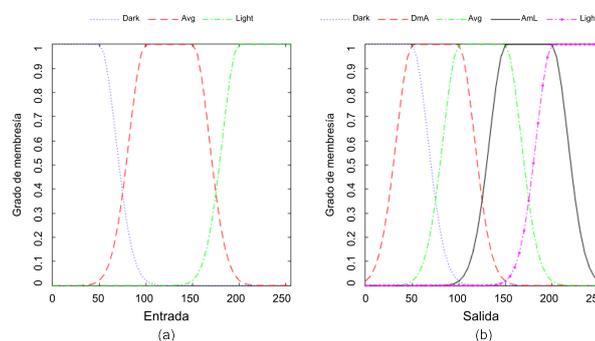
El primer conjunto de parámetros definen la varianza y la media de la mitad izquierda de la gaussiana, mientras que el otro conjunto la mitad derecha de la misma. Cuando  $c_1$  y  $c_2$  son diferentes, los valores intermedios toman valor 1.0, mientras que en el resto de los casos toman valores menores a la unidad. Esta función permite la construcción de funciones gaussianas no simétricas. La Fig. 3 presenta un ejemplo de este tipo de funciones.

## 2.3 Optimización del sistema borroso

Nuestro problema requiere la optimización de los parámetros de las funciones de membresía, se decidió codificar cada partícula como una matriz de cuatro columnas  $\sigma_1, c_1, \sigma_2, c_2$  por un total de  $n$  filas, siendo  $n$  la cantidad de gaussianas a optimizar.

En el proceso iterativo utiliza una función objetivo como indicativo de la dirección de la posible solución y funciones de corte. De este modo, aquella partícula que logre por cada pasada un mejor ajuste es marcada y utilizada en la próxima iteración.

Inicialmente el conjunto de partículas se ubican en una posición aleatoria y luego, a medida que pasan las iteraciones, se van acomodando en el espacio de



**Fig. 3.** Conjuntos difusos del sistema. a) Conjuntos de entrada. b) Conjuntos de salida.

posibles soluciones. Este proceso se repite hasta que se llega a la cota de corte o bien cuando el número de iteraciones se agota. El valor final de la última solución debería ser la mejor posible entre todo el espacio explorado hasta esa instancia.

### 3 Datos y medidas de evaluación

Las imágenes utilizadas en este trabajo pertenecen a la base de datos (SIPI) <sup>2</sup>. Esta contiene 44 imágenes en formato TIFF, 16 en color y 28 monocromáticas, con tamaños promedios entre 256x256 y 512x512 px.

Para comparar el filtro de mediana frente al filtro obtenido con la metodología propuesta se usaron dos medidas objetivas. La primera, denominada *Pico de la señal a ruido* (PSNR, por sus siglas en inglés)[3], se define como:

$$PSNR = 10 \log_{10} \left( \frac{MAX_I^2}{MSE} \right), \quad (2)$$

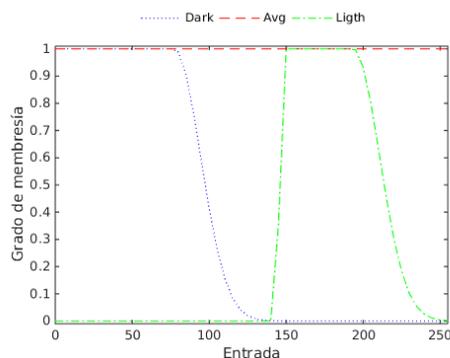
donde  $MAX_I$  denota el máximo valor que puede tomar un píxel en la imagen, y  $MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |I(i, j) - K(i, j)|^2$  corresponde al error cuadrático medio calculado para la ventana, siendo  $I(i, j)$  el valor del píxel de la imagen contaminada con ruido en la posición  $(i, j)$ , y  $K(i, j)$  el valor del píxel de la imagen original sin contaminar para la posición  $(i, j)$ . Esta medida se expresa en decibelios (db); valores elevados indican un mejor filtrado del ruido en la imagen. La otra medida empleada es la *Similaridad Estructural* (SSIM, por sus siglas en inglés)[7], definida como:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (3)$$

donde  $\mu_x$  representa la media de  $x$ ,  $\mu_y$  la media de  $y$ ,  $\sigma_x^2$  la varianza de  $x$ ,  $\sigma_y^2$  la varianza de  $y$ , y  $\sigma_{xy}$  la covarianza entre  $x$  e  $y$ .  $c_1 = (k_1L)^2$  y  $c_2 = (k_2L)^2$  corresponden a coeficientes de estabilización, siendo  $L$  el rango dinámico del píxel y  $k_1, k_2$  dos constantes a especificar.

<sup>2</sup> <http://sipi.usc.edu/database/database.php?volume=misc>

6 Manzo R.J., Vizzarri C., Gerard M.F.



**Fig. 4.** Ejemplo de un conjunto de entrada del sistema borroso optimizado mediante PSO.

## 4 Resultados experimentales

En esta sección se presentan los resultados de los experimentos realizados con la estrategia propuesta. En primer lugar se realizó la optimización del sistema difuso  $\varphi(\cdot)^*$  para diferentes imágenes y se comparó los resultados frente al filtro de mediana y el sistema difuso sin optimizar  $\varphi(\cdot)$ . Luego se estudió el grado de generalización del método frente a otras imágenes en la etapa de prueba. Finalmente, se propuso una modificación sobre la estrategia de entrenamiento para lograr una mejora en la corrección de ruido.

### 4.1 Análisis de resultados

Inicialmente, se realizaron pruebas preliminares y se observó que los mejores resultados se lograban optimizando sólo los conjuntos de entrada del sistema borroso. Para lograr esto, durante la etapa de entrenamiento se establecieron los siguientes parámetros experimentales: 30 partículas, 200 iteraciones, vecindad cuadrada de  $3 \times 3$ , constantes  $c_1 = 2.0$  y  $c_2 = 3.5$ , y un criterio de corte basado en un número máximo de iteraciones y un valor deseado de  $\text{PSNR} \geq 50$  db. La Fig. 4 presenta un ejemplo de los conjuntos optimizados obtenidos con esta propuesta.

Para evaluar la calidad de la restauración que se obtuvo con la optimización de las funciones de membresía, se contaminaron 4 imágenes con ruido de sal y pimienta al 1% y se procedió a su posterior restauración. Debido a que PSO es una técnica que incorpora componentes estocásticas, se realizaron 10 restauraciones independientes por cada imagen y los resultados se promediaron.

La Tabla 1 presenta los resultados obtenidos para cada imagen y los 3 métodos analizados. Las 3 primeras columnas corresponden al cálculo de la PSNR y las 3 restantes a SSIM. La columna Referencia corresponde a la relación entre la PSNR de la imagen original y la misma imagen contaminada con ruido. Conocer este resultado es importante para evaluar la calidad de la restauración, siendo deseable obtener valores superiores a éste luego de aplicar un método de

**Tabla 1.** Resultados comparativos de PSNR y SSIM del método propuesto en distintas imágenes con 1% de ruido. Los valores corresponden a promedios sobre 10 filtros borrosos. \* indica el sistema optimizado con PSO. Los filtros que producen mejores resultados se indican en negrita.

	PSNR			SSIM		
	Referencia	Mediana	Fuzzy	Ganancia	Mediana	Fuzzy
Lena	25,258	35,939	<b>38,803</b>	13,544	0,935	<b>0,984</b>
Cameraman	25,758	26,751	<b>39,483</b>	13,725	0,887	<b>0,982</b>
Mandrill	25,454	20,829	<b>39,183</b>	13,730	0,488	<b>0,993</b>
Blonde	25,247	24,790	<b>35,328</b>	10,080	0,790	<b>0,983</b>

restauración. La columna mediana presenta la relación entre la PSNR o SSIM de la imagen restaurada aplicando el filtro de medianas y la misma contaminada con ruido. De la misma manera, la columna Fuzzy presenta la relación que existe entre las medidas calculadas para la imagen restaurada con el filtro del método propuesto optimizado y la imagen contaminada con ruido. La columna Ganancia es la diferencia, en dbs, entre la columna Fuzzy y la Referencia.

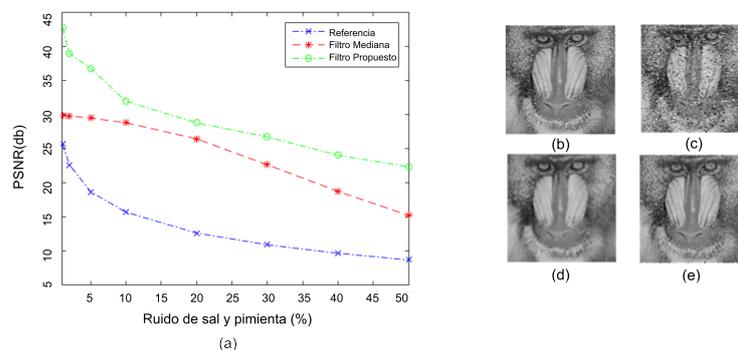
En la Tabla 1 observamos como los valores de los filtros propuestos optimizados (Fuzzy) son superiores al filtro de mediana en todas las imágenes. La ganancia promedio de todas las imágenes está en el orden de los 13 dbs, lo cual resulta en un valor más que satisfactorio. A su vez, el valor de SSIM del método propuesto optimizado es más alto que el del filtro de mediana para todas las imágenes.

Dado que el ruido que se empleó en estos experimentos es bajo, se decidió evaluar la calidad de la restauración para valores de ruido más elevados. Para ello, y por una simplicidad de la prueba, se seleccionó la imagen *Mandrill* y se la sometió a valores crecientes de ruido. Se utilizó esta imagen en particular dado que posee un nivel de detalle elevado, lo que permite apreciar a simple vista los resultados obtenidos.

La Fig. 5(a) presenta los resultados de la restauración de la imagen en términos de PSNR para un amplio rango de valores de ruido, tanto para el filtro de mediana (asteriscos rojos), el sistema difuso optimizado (círculos verdes) y la Referencia (estrellas azules). Al igual que en la prueba anterior, si tenemos en cuenta la Referencia como el valor a superar, vemos claramente como el método propuesto optimizado supera ampliamente al filtro de mediana. Por otra parte, la Fig. 5(d) y 5(e) muestran, respectivamente, los resultados obtenidos luego de aplicar el filtro de mediana y del filtro optimizado. Claramente, como se observa en la Fig. 5(e) el filtro optimizado logra reducir el ruido eficazmente preservando los detalles de la imagen y sin introducir desenfoque, a diferencia de lo que ocurre en la Fig. 5(d) con el filtro de medianas.

## 4.2 Generalización del filtro propuesto

En la etapa anterior, se planteó una metodología de optimización del filtro que hizo muy dependiente al sistema borroso de la imagen para el que fue entrenado,



**Fig. 5.** Resultados obtenidos luego de aplicar el sistema optimizado. a) Valores de PSNR para distintas densidades de ruido para la imagen *Mandrill*, obtenidos por el filtro de mediana (asterisco) y el filtro optimizado (circulo). Referencia indica los valores de PSNR de la imagen sin filtrar. b) Imagen original. c) Imagen con 10% de ruido. d) Imagen restaurada aplicando el filtro de mediana. e) Imagen restaurada aplicando el filtro optimizado.

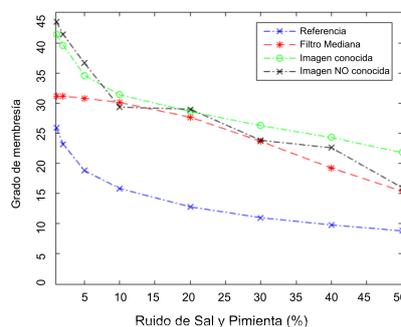
**Tabla 2.** Análisis comparativo entre filtros optimizados para imágenes con 1% de ruido y aplicados sobre otra imágenes distintas. Los valores corresponden a promedios de PSNR sobre 10 filtros borrosos. Los filtros que producen mejores resultados se indican en negrita.

	FL-Lena(db)	FL-Cameraman(db)	FL-Mandrill(db)	FL-Blonde(db)
Lena	<b>38.803</b>	37,932	33,671	37,401
Cameraman	38,623	<b>39.483</b>	33,883	26,751
Mandrill	38,708	38,348	<b>39.183</b>	36,657
Blonde	33,199	33,744	29,203	<b>35.328</b>

lo que llevo a pensar de que filtro tenia una baja capacidad de generalización. Para verificar esto, se realizó un análisis comparativo entre los filtros optimizados. Así se optimizaron 4 filtros, cada uno para una imagen diferente contaminada con ruido de 1% db, y luego se aplicó cada uno sobre las demás imágenes buscando evaluar la calidad de la restauración

La Tabla 2 presenta los resultados comparativos entre los 4 filtros optimizados. Las filas indican las imágenes de entrenamiento usadas, mientras que las columnas los filtros optimizados con cada imagen. De este modo, al aplicar el filtro FL-Lena (optimizado utilizando la imagen Lena) sobre la imagen Lena contaminada se logra un valor promedio de PSNR de 38.803 db. Si ahora se aplican sobre la misma imagen contaminada los filtros FL-Cameraman, FL-Mandrill y FL-Blonde se obtienen valores de PSNR promedio de 37.932 db, 33.671 db, 37.401 db, respectivamente. Claramente, los resultados muestran que los filtros producen mejores resultados cuando se aplican sobre la imagen usada durante la optimización.

Dado que los resultados expuestos no eran satisfactorios, se planteo la optimización del sistema teniendo en cuenta imágenes con distintas características



**Fig. 6.** Valores de PSNR calculados del filtro de mediana, método propuesto y del valor de referencia con distintas densidades de ruido para una *imagen entrenamiento* y una *imagen de prueba*

de histograma y detalle, que abarquen mejor todo el rango de grises, con el objetivo de lograr un filtro que no fuera específico para una sola imagen y que fuera capaz de restaurar imágenes de forma más general, independientemente de sus características particulares. Para esta etapa se seleccionaron del corpus dos conjuntos de 4 imágenes. Un conjunto fue nombrado *Imagen conocida* y fue utilizado en la etapa de entrenamiento. El otro conjunto, denominado *Imagen no conocida*, fue utilizado sólo en la etapa de pruebas.

Inicialmente, el conjunto de entrenamiento se contaminó con densidades de ruido en el rango de 1% – 50% y se optimizó un filtro para cada caso. En un paso posterior se tomó una imagen del conjunto de prueba y se contaminó con las mismas densidades de ruido que al conjunto de entrenamiento. Finalmente el conjunto de filtros optimizados para las densidades de ruido nombradas, se aplicó al conjunto de imágenes de entrenamiento y a la imagen seleccionada para las pruebas.

Los resultados se presentan en la Fig. 6. Del conjunto de entrenamiento se observa que para distintos porcentajes de ruido el valor PSNR del filtro optimizado (círculos verdes) es superior al valor de Referencia (asteriscos azules) y a la mediana (asteriscos rojos). A partir del conjunto de prueba también se aprecia que el valor de la PSNR del filtro optimizado supera al valor de Referencia en todo el rango de ruido considerado. De esta manera, los resultados indican que la optimización del filtro con la nueva metodología de entrenamiento produce filtros con una mayor capacidad de generalización, por lo que es factible su uso para la restauración de imágenes con este tipo de ruido.

## 5 Conclusiones y trabajo futuro

En este trabajo se propuso la optimización, mediante PSO, de las funciones de membresía de un sistema borroso para filtrado de ruido sal y pimienta. Se realizó la optimización de los conjuntos de entrada y salida por separado y en conjunto, obteniéndose los mejores resultados cuando sólo se optimizaron los conjuntos de entrada. Para evaluar los resultados del filtro se plantearon dos metodologías

de pruebas. Una entrenando el filtro para una imagen en particular la cual se detectó que hizo muy dependiente al filtro a la imagen para el que fue entrenado. Esto llevó a la conclusión de que el filtro obtenido de esta manera tenía una baja capacidad de generalización. Teniendo en cuenta esto se pensó en una forma de lograr independizar el filtro de la imagen a tratar. Se formaron un conjunto de imágenes para entrenamiento y un conjunto para test. Para cada conjunto, se seleccionaron imágenes con distintas características de histogramas, distribuidas en las intensidades bajas, medias y altas cubriendo todo el rango posible [0,255]. Los resultados muestran que el filtro optimizado con esta configuración reduce sensiblemente el ruido sal y pimienta en una imagen para la cual no fue entrenado. Además, a diferencia del filtro de mediana, el filtro optimizado logra reducir el ruido sin introducir desenfoque en la imagen.

Si bien los resultados obtenidos son prometedores, existen varios aspectos que deberían ser analizados. Por un lado, sería interesante estudiar la detección de los píxeles corruptos con ruido impulsivo siguiendo la dirección del gradiente. Esto podría favorecer la obtención de valores de corrección más acordes con esta variación. Por otro lado, el filtro optimizado presentado en este trabajo no contempla los bordes durante la corrección del ruido. Esta limitación podría abordarse completando con ceros las ventanas para los píxeles de los bordes de la imagen (zero-padding), de esta manera procesarla de forma completa para mejorar los resultados. Otro punto pendiente es la comparación con distintas técnicas del estado del arte, realizando un análisis estadístico adecuado para evaluar las ventajas y desventajas de esta propuesta. Finalmente, sería interesante extender estos resultados a la corrección de ruido en imágenes a color.

## Referencias

1. Cheng, K.Y., Tzou, Y.Y.: Fuzzy optimization techniques applied to the design of a digital PMSM servo drive. *IEEE Transactions on Power Electronics* 19(4), 1085–1099 (2004)
2. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*. Prentice-Hall, 3 edn. (2007)
3. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of PSNR in image/video quality assessment. *IEEE Transactions on Image Processing* 13(4), 800–801 (2008)
4. Jayaraman, S., Esakkirajan, S., Veerakumar, T.: *Digital Image Processing*. Paperback (2009)
5. Kamyra, S., Sachdeva, M.: Fuzzy logic based image de-noising and enhancement for grayscale images. *International Journal of Computer Applications* 74(2), 5–9 (2013)
6. Rahmani, R., Mahmodian, M., Mekhilef, S., Shojaei, A.: Fuzzy logic controller optimized by particle swarm optimization for DC motor speed control. In: 2012 IEEE Student Conference on Research and Development (SCOReD). No. 109–113, IEEE (2012)
7. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*. vol. 2, pp. 1398–1402. IEEE (2004)