

Proyecto integrador DSP

Filtrado de ruido con transformada de wavelet

Gatica, Isaias
 Martin, Santiago
 Saez, Lautaro Andrés
 Vidman, Xavier Harry

I. OBJETIVOS

- Explicar los conceptos teóricos de la transformada wavelet.
- Filtrar ruido de una imagen.
- Comparar los resultados para los diferentes algoritmos de umbrales y familias de wavelets.
- Comparar el filtrado por transformada de wavelet con otros métodos de filtrado de imágenes.

II. INTRODUCCIÓN

En el siguiente informe estudiaremos los conceptos de las funciones wavelets, transformada de wavelet, y cómo utilizarlos para filtrar ruido de señales unidimensionales y bidimensionales. En nuestro caso, nos centramos en imágenes (dos dimensiones). Para el filtrado de ruido se hará un estudio de los umbrales más utilizados, con el fin de poder decidir qué umbral usar para la señal o imagen que se quiera estudiar. Como adelanto, el umbral es el parámetro que decide cómo se filtrarán los coeficientes que caracterizan a la imagen.

III. MARCO TEÓRICO

Una *Wavelet* es una “pequeña onda” que tiene su energía concentrada en un período de tiempo determinado, son de duración definida e irregulares, lo que les permite adaptarse y converger de mejor manera a la señal que se quiere analizar. Las familias de estas wavelet, también conocidas como “wavelets madres”, se definen como:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad a, b \in \mathbb{R} \quad ; a \neq 0. \quad (1)$$

Donde a y b son los parámetros de escala y traslación respectivamente. Se puede ver en la figura 3 como trabajan los parámetros a y b . El parámetro b , asociado a la traslación, hace que la wavelet recorra a la señal $x(t)$ a través del tiempo. Mientras que el parámetro a , permite controlar el ancho de la wavelet.

Por otro lado, la transformada de wavelet queda definida como:

$$C_f(a, b) = \int_{-\infty}^{+\infty} x(t) \overline{\psi_{a,b}(t)} dt \quad (2)$$

Donde $\psi_{a,b}(t)$ es la función que se utiliza para analizar a la señal $x(t)$ que se quiera estudiar. Realizando una analogía con Fourier, el trabajo que realiza $\psi_{a,b}(t)$, es análogo al de las

exponenciales complejas $e^{-j\omega t}$. El resultado de esta transformada es una familia de coeficientes $C_f(a, b)$ que representan de mejor manera la señal teniendo en cuenta la wavelet madre utilizada. Es notorio que el resultado es bidimensional ya que a y b son parámetros independientes. Donde a está relacionado con la frecuencia que posee la señal y b nos brinda información del momento temporal que se analiza. En la figura ?? podemos verlo de mejor manera, donde el gráfico superior muestra cómo cambia la frecuencia de una señal en función de la traslación b , en específico en b_0 . Y el gráfico inferior muestra el resultado de la transformada, evidenciando en el tiempo b_0 el cambio de frecuencia.

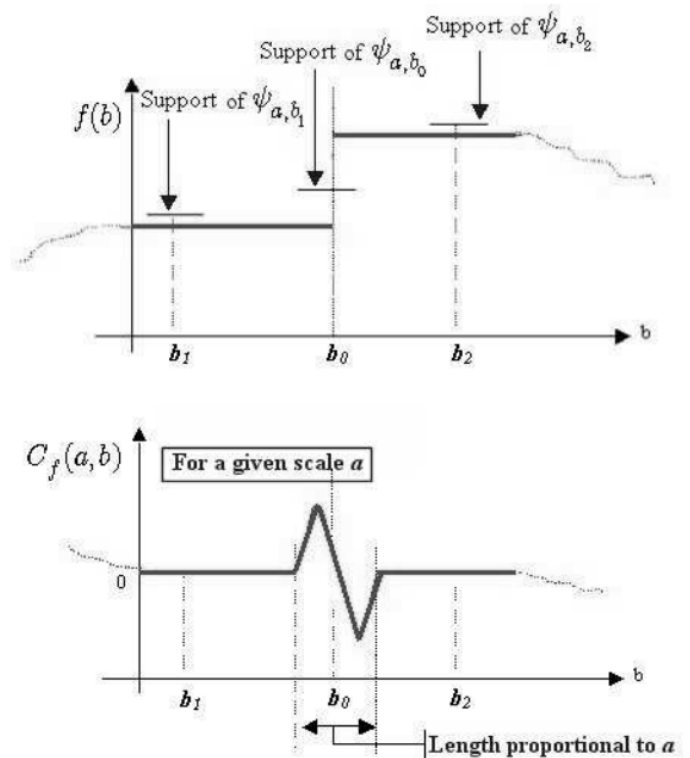


Figura 1: Representación del resultado de transformada wavelet en relación a los coeficientes de traslación y escala

Por lo tanto, el uso de wavelets nos permite realizar un análisis paulatino en tiempo y frecuencia como se mencionó anteriormente. Cabe recalcar que existen diferentes tipos de wavelets madres, en la figura 2 se pueden ver las más

utilizadas:

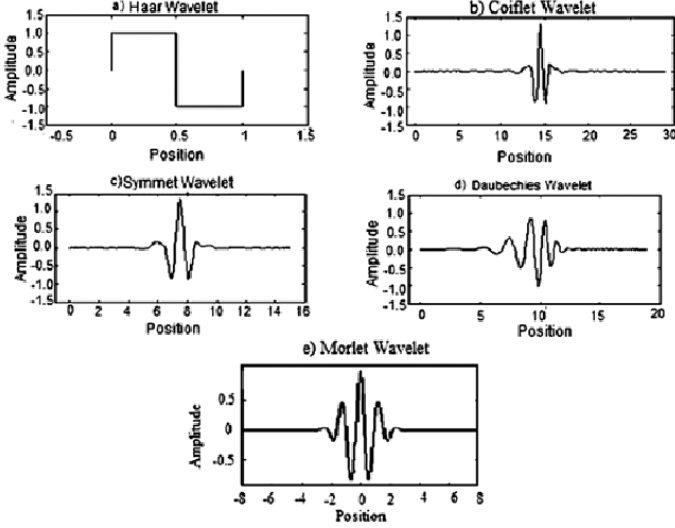


Figura 2: Bases ortonormales de wavelets madres más utilizadas

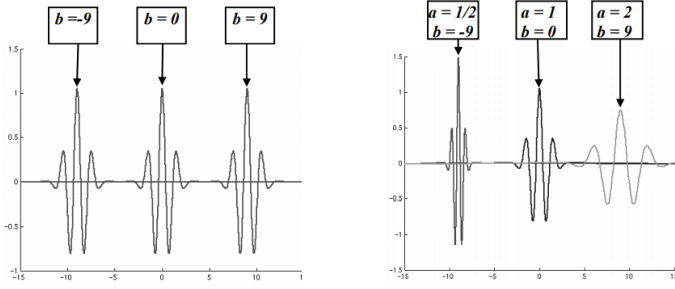


Figura 3: Traslación y escalamiento de una wavelet

En este informe nos limitaremos a estudiar el caso en el que $a = 2$ y $b = 1$, obteniendo la siguiente expresión:

$$\psi_{j,k}(t) = 2^{-j/2} \psi(2^{-j}t - k) \quad (3)$$

Donde las funciones $\{\psi_{j,k}\}_{j,k \in \mathbb{Z}}$ forman una base ortonormal de $L^2(\mathbb{R})$. Con el fin de explicar este requerimiento, es necesario explicar un concepto llamado **análisis multiresolución**.

III-A. Análisis multiresolución

Un análisis multiresolución para $L^2(\mathbb{R})$ consiste en una secuencia de subespacios cerrados de $L^2(\mathbb{R})$, $\{V_j\}_{j \in \mathbb{Z}}$, y una función $\phi \in V_0$ tal que se cumplan las siguientes condiciones:

i. Los espacios V_j están anidados, es decir:

$$\dots \subset V_{-1} \subset V_0 \subset V_1 \dots$$

ii. $\overline{\cup_{j \in \mathbb{Z}} V_j} = L^2(\mathbb{R})$ y $\cap_{j \in \mathbb{Z}} V_j = \{0\}$

iii. Para todo $j \in \mathbb{Z}$, $V_{j-1} = D(V_j)$

iv. $f \in V_0 \rightarrow T_k f \in V_0, \forall k \in \mathbb{Z}$

v. $\{T_k \phi\}_{k \in \mathbb{Z}}$ es una base ortonormal de V_0

Definimos a los espacios V_j . Por otro lado, definimos a W_j como el complemento ortogonal de V_j en V_{j-1} , que permite plantear la siguiente relación:

$$V_{j-1} = V_j \oplus W_j \quad (4)$$

W_j es llamado conjunto de **detalle** debido a que la proyección ortogonal de la señal en este espacio son los detalles, y V_j espacio de **aproximación**, ya que la proyección ortogonal de la señal sobre este espacio son las aproximaciones. Con lo cual, podemos reescribir la ecuación 4 como:

$$A_{j-1}(t) = A_j(t) + D_j(t) \quad (5)$$

Donde $A_j(t)$ se define de la siguiente forma:

$$A_j(t) = \sum_{k \in \mathbb{Z}} \beta_{j,k} \phi_{j,k}(t) \quad (6)$$

Donde:

$$\beta_{j,k} = \langle x(t), \phi_{j,k}(t) \rangle = \int_{\mathbb{R}} x(t) \overline{\phi_{j,k}(t)} dt \quad (7)$$

Estos coeficientes $\beta_{j,k}$ definen los promedios locales de la señal. Por otro lado, $D_j(t)$ se define a través de la siguiente expresión:

$$D_j(t) = \sum_{k \in \mathbb{Z}} \alpha_{j,k} \phi_{j,k}(t) \quad (8)$$

Donde los coeficientes $\alpha_{j,k}$ se relacionan con la diferencia entre promedios locales sucesivos (fluctuaciones de la señal), y quedan definidos de la siguiente forma:

$$\alpha_{j,k} = \langle x(t), \phi_{j,k}(t) \rangle = \int_{\mathbb{R}} x(t) \overline{\phi_{j,k}(t)} dt \quad (9)$$

Para dar un cierre a la idea de multiresolución se profundiza en la ecuación 4 que es el punto más fuerte de esta técnica. Sin más preámbulos, la reescribimos de la siguiente manera:

$$V_J = V_K \oplus W_K \oplus \dots \oplus W_{J+1}, \quad J < K \quad (10)$$

La ecuación 10 expresa que podemos pasar de un nivel de resolución J a cualquier nivel superior sumando los sucesivos detalles de los niveles inferiores. Es decir, la aproximación mejora a medida que el nivel J decrece. También observamos que cualquier nivel de resolución V_j se puede expresar como:

$$V_J = \sum_{j=J+1}^{+\infty} W_j \quad (11)$$

Por lo tanto cualquier nivel es igual a la suma de sus infinitos detalles. Análogamente cualquier función puede ser expresada como una suma de infinitos detalles:

$$L^2(\mathbb{R}) = \oplus_{j=-\infty}^{+\infty} W_j \quad (12)$$

A partir de las ecuaciones 11 y 12 se puede ver que:

$$L^2(\mathbb{R}) = V_J \oplus (\oplus_{j=-\infty}^J W_j) \quad (13)$$

Finalmente, como el espacio V_j se relaciona con el espacio detalle $A_j(t)$ y el $W_j(t)$ con los espacios de detalles $D_j(t)$:

$$x(t) = A_J(t) + \sum_{j=-\infty}^J D_j(t) \quad (14) \quad \text{con}$$

Por lo tanto, podemos definir a cualquier señal $x(t)$ como la suma de una **aproximación**, que vendría ser la parte más “burda” de la señal, con sus infinitos **detalles**, que serían “toques finales”. Esto se observa en la figura 4, donde se utilizó una wavelet symmet 4 y 3 de niveles de descomposición. En rojo la señal original, en azul la aproximación y en verde las tres señales de detalle correspondientes.

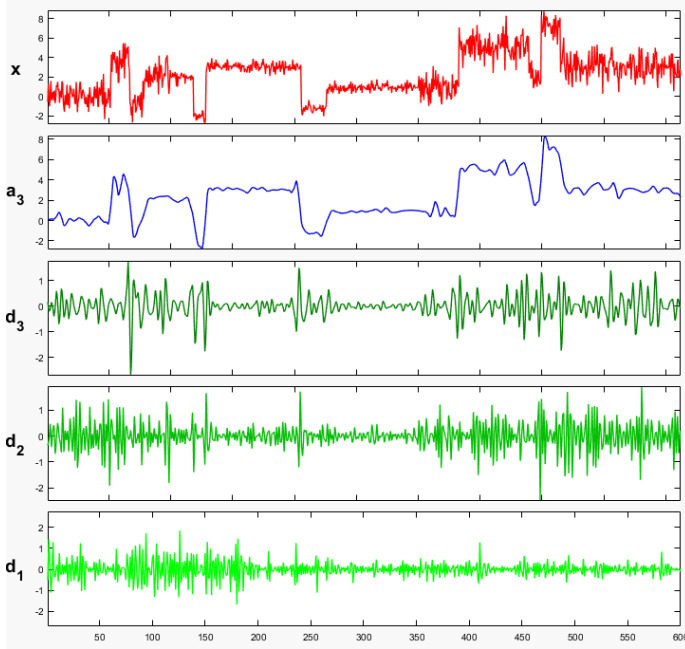


Figura 4: Descomposición de nivel 3 de una señal utilizando una wavelet sym4

A partir de esta idea, vemos que es factible aplicar la transformada de wavelet para eliminar el ruido de una señal. Por lo tanto, si se busca una forma de filtrar los coeficientes a partir de parámetros relacionados con el ruido de la señal, se podrían eliminar las componentes no deseadas. Luego, a la hora de reconstruir la señal, se obtendría una versión sin ruido. Estos parámetros son llamados **umbrales** y se definirán más adelante.

Teniendo en mente la idea del filtrado, y aún con el inconveniente de que el producto interno para calcular los coeficientes (ecuaciones 7 y 9) requiere un alto costo computacional, se plantea una estrategia basada en un algoritmo que relacione las bases ortonormales de wavelets y los bancos de filtros. Dicho algoritmo permite calcular los coeficientes a^j y d^j a partir de los coeficientes de aproximación de una escala anterior de resolución, es decir los a^{j-1} , siendo estos los coeficientes que representan de manera perfecta la señal a analizar. Para llegar a esta relación se parte de la ecuación 5 donde:

$$A_j(t) = \sum_{k \in \mathbb{Z}} a_k^j \phi_{j,k}(t) \quad (15)$$

$$D_j(t) = \sum_{k \in \mathbb{Z}} d_k^j \psi_{j,k}(t) \quad (16)$$

$$a_k^j = \langle A_j(t), \phi_{j,k}(t) \rangle \quad (17)$$

$$d_k^j = \langle A_j(t), \psi_{j,k}(t) \rangle \quad (18)$$

Si deseamos obtener a_k^j y d_k^j a partir de A_{j-1} , realizamos el producto interno A_{j-1} y la función de escala y wavelet

$$\langle A_{j-1}(t), \psi_{j,k}(t) \rangle = \langle A_j(t) + D_j, \psi_{j,k}(t) \rangle \quad (19)$$

$$\langle A_{j-1}(t), \phi_{j,k}(t) \rangle = \langle A_j(t) + D_j, \phi_{j,k}(t) \rangle \quad (20)$$

Operando matemáticamente (Ver anexo) llegamos a que:

$$a_k^j = \sum_{p \in \mathbb{Z}} a_p^{j-1} \langle \psi_{j-1,p}(t), \psi_{j,k}(t) \rangle = \sum_{p \in \mathbb{Z}} a_p^{j-1} \sqrt{2} a_{p-2k} \quad (21)$$

$$d_k^j = \sum_{p \in \mathbb{Z}} a_p^{j-1} \langle \psi_{j-1,p}(t), \phi_{j,k}(t) \rangle = \sum_{p \in \mathbb{Z}} a_p^{j-1} \sqrt{2} b_{p-2k} \quad (22)$$

Donde:

$$\sqrt{2} a_{p-2k} = \langle \phi_{j-1,p}(t), \phi_{j,k}(t) \rangle \quad (23)$$

$$\sqrt{2} b_{p-2k} = \langle \phi_{j-1,p}(t), \psi_{j,k}(t) \rangle \quad (24)$$

Se observa que estos coeficientes son productos internos de funciones ya conocidas, la función wavelet y su función de escala. Estos valores ya fueron calculados y se los asume como conocidos, por lo cual si los consideramos como los coeficientes de filtros podemos definir a $LoD = \{[LoD]_n\}_{n \in \mathbb{Z}}$ y a $HiD = \{[HiD]_n\}_{n \in \mathbb{Z}}$ como el filtro pasabajos y un filtro pasaltos respectivamente.

Por lo cual:

$$[LoD]_n = \sqrt{2} a_{-n} \quad (25)$$

$$[HiD]_n = \sqrt{2} b_{-n} \quad (26)$$

Quedando entonces definidos nuestros coeficientes en un nivel j como:

$$a_k^j = \sum_{p \in \mathbb{Z}} a_p^{j-1} [LoD]_{2k-p} \quad (27)$$

$$d_k^j = \sum_{p \in \mathbb{Z}} a_p^{j-1} [HiD]_{2k-p} \quad (28)$$

Estos coeficientes presentan un aspecto muy similar a la suma de convolución entre el coeficiente del nivel anterior

con el filtro correspondiente. Para que esta sea la convolución se introduce la operación de submuestreo, definida como:

$$y = dec(x) \Leftrightarrow y_n = x_{2n} \forall n \in \mathbb{Z} \quad (29)$$

usando las ecuaciones 27, 28, 29 y la convolución llegamos a:

$$a^j = dec(a^{j-1} * LoD) \quad (30)$$

$$d^j = dec(a^{j-1} * HiD) \quad (31)$$

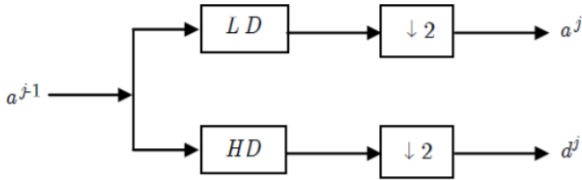


Figura 5: Descomposición con banco de filtros para 1D

Lo que nos permite ver que cada señal se descompone en 2 señales, la de aproximación y detalle de la mitad de la dimension de la señal original.

Es posible demostrar que se puede realizar proceso inverso y reconstruir la señal aproximada, de la misma manera en que se descompuso, la suma de señales convolucionadas con filtros pasabajos y pasaaltos, matemáticamente se puede ver en la siguiente expresión:

$$a_k^{j-1} = \sum_{p \in \mathbb{Z}} a_p^j [LoR]_{k-2p} + \sum_{p \in \mathbb{Z}} d_p^j [HiR]_{k-2p} \quad (32)$$

Donde LoR y HiR son los filtros pasabajos y pasaaltos de reconstrucción.

Como mencionamos anteriormente, las señales $a^j y d^j$ tienen una dimension menor a la original que se desea reconstruir, por lo que es necesaria “rellenarlas” para que tengan la dimension correcta.

Este desarrollo lo podemos extender a dos dimensiones, permitiendo descomponer a las imágenes en cuatro señales diferentes que corresponden a las señales de aproximación y una descomposición de señales de detalles, una para detalles horizontales, verticales y diagonales. Lo mencionado se visualiza en la figura 6.

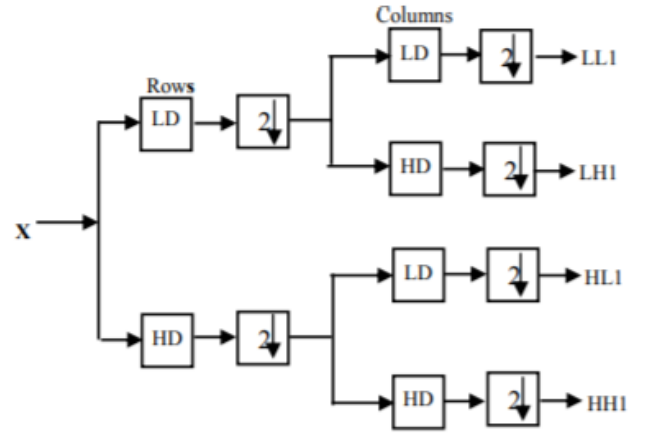


Figura 6: Descomposición con banco de filtros para 2D

Donde las matrices finales representan:

- LL representa la aproximacion.
- LH representa los detalles verticales.
- HL representa los detalles horizontales.
- HH representa los detalles diagonales.

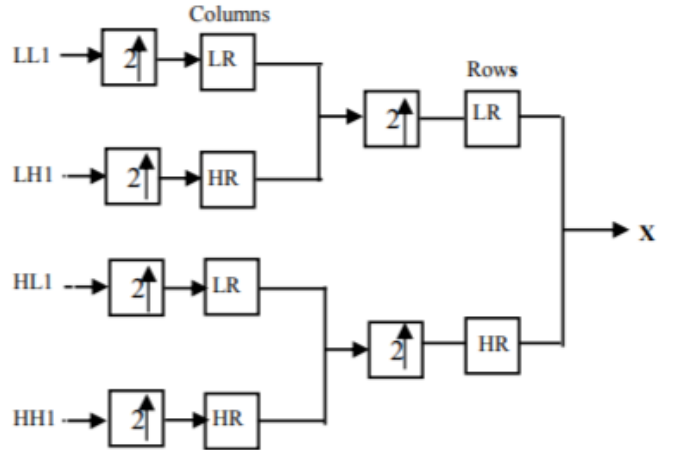


Figura 7: Recomposición con banco de filtros para 2D

III-B. Umbrales

Como dijimos anteriormente, es necesario determinar parámetros para realizar el filtrado de los coeficientes. Estos parámetros se denominan **umbrales**. Con el fin de seguir la nomenclatura utilizada en la bibliografía aplicada, a este método se lo denominará umbralización. La umbralización tiene dos características importantes a definir: el tipo de umbralización y las distintas forma de calcular el valor del umbral. En la figura 8 observamos los tipos de umbralización **hard** y **soft**. El valor del umbral se encuentra notado con τ . El tipo de umbralización **hard** anula todos los coeficientes que estén en el rango $[-\tau, \tau]$ y no modifica el resto. Lo mismo cumple el umbralizado **soft** con la diferencia que los coeficientes fuera de este rango se ven reducidos linealmente. El tipo hard reproduce de mejor forma los picos

y discontinuidades si se piensa una señal en una dimensión, o por ejemplo, los bordes si se piensa en imágenes. Por otro lado, el tipo soft genera resultados mas suaves en la aproximación de la señal, lo que es satisfactorio si no se busca un resultado discontinuo. En las imágenes este método genera un efecto de difuminado justamente para suavizar el resultado final.

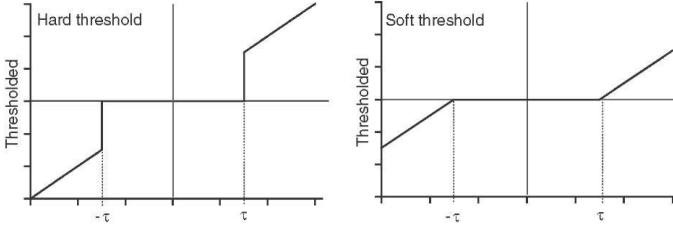


Figura 8: Tipos de umbralización más utilizados

Ahora, para el cálculo del umbral λ existen distintos tipos de algoritmos. En este informe se estudiarán los siguientes:

- VisuShrink.
- LevelShrink.
- BayesShrink.
- NormalShrink.
- AWT(Adaptative Wavelet Treshholding).

III-B1. VisuShrink: este algoritmo, más conocido como umbral universal se calcula con la siguiente ecuación:

$$t_{univ} = \sigma_n \sqrt{2 \ln(n)} \quad (33)$$

Donde σ es la varianza del ruido y n es la dimensión de la señal. Esta ecuación se deriva a partir de que es conocido que el máximo valor de una señal que siga una distribución gaussiana con media cero y varianza σ^2 , ($x \sim N(0, \sigma^2)$) va a ser menor que el valor $\sigma_n \sqrt{2 \log(n)}$ con una probabilidad que tiene a uno cuando n tiene a infinito. Por lo tanto, este umbral igualará a cero los coeficientes que presenten ruido puro con una alta probabilidad. La simpleza del algoritmo lo hace de los mas utilizados, sin embargo al solo tener en cuenta las características del ruido y de la dimensión de la imagen suele suavizar mucho la imagen. Esto se debe a que elimina muchos coeficientes de información útil al intentar eliminar el ruido presente.

Antes de continuar con el siguiente umbral, es necesario mencionar dos aspectos importantes, el primero de ellos es por qué los umbrales dependen de la dimension de la señal. Esto se debe a que mientras mayor sea la dimensión aumenta la redundancia de la señal. Por lo tanto, hay menos información nueva (al aumentar la dimensión) que la información de base. Desde el punto de vista de wavelet, esto no genera que se aumenten los coeficientes importantes. Por consecuente, existe una relación proporcional entre los coeficientes de ruido y la dimensión de la señal, lo que implica la necesidad de agregarlo al cálculo del umbral.

Y como segundo aspecto de importancia para la estimación de

la varianza de ruido de la imágenes se utilizan los coeficientes de mayor frecuencia que son los que presentan mayor energía de ruido. Estos coeficientes quedan representados en las matrices de detalle diagonal, ya que si recordamos la figura 6 podemos ver que la matriz de detalle diagonal se forma luego de filtrar dos veces la señal con un filtro pasaalto.

III-B2. LevelShrink: este umbral depende de la escala de nivel, a diferencia de VisuShrink, que aplica el mismo umbral a cada nivel de descomposición. Su expresión es la siguiente:

$$t_{level} = \sigma_n \sqrt{2 \ln(n)} 2^{-(J-j)/2} \quad (34)$$

Donde J es el número total de niveles de descomposición y j el nivel correspondiente. Si bien esto genera un umbral que se adapta de mejor manera, sigue sin tener en cuenta los detalles de la señal a analizar.

III-B3. BayesShrink: la base de este umbral es minimizar la inferencia bayesiana. Sus ecuaciones son:

$$t_{bayes} = \begin{cases} \sigma_n^2 / \sigma_s & \text{si } \sigma_n^2 < \sigma_y^2 \\ \max(|y_i|) & \text{si } \sigma_n^2 \geq \sigma_y^2 \end{cases} \quad (35)$$

Donde σ_n^2 / σ_s es la varianza del ruido, σ_y^2 es la varianza de la señal original y σ_s es la varianza de la señal pura que se estima con la siguiente ecuación:

$$\sigma_{sk} = \sqrt{\max \left(\left(\frac{1}{\text{size}(H_k)} \sum_{y_i \in H_k} y_i^2 \right) - \sigma_n^2, 0 \right)} \quad (36)$$

Este algoritmo tiene como ventaja respecto a los anteriores que utiliza información de la señal a analizar pero es necesario conocer información de la imagen original (sin ruido).

III-B4. NormalShrink: este algoritmo suple la desventaja de BayesShrink ya que no se necesita información de la señal sin ruido y presenta todas las ventajas incorporadas de utilizar información de la señal a analizar. Se lo define como:

$$t_{normal} = \frac{\beta_k \sigma_n^2}{\sigma_s} \quad (37)$$

Donde:

$$\beta_k = \sqrt{\frac{\ln(\text{size}(H_k))}{J}} \quad (38)$$

Y σ_s es el desvío estándar de la señal de cada nivel.

III-B5. AWT(Adaptative Wavelet Treshholding): por último, presentamos el algoritmo AWT el cual depende de la media aritmética y de la media geométrica de los coeficientes de la señal en cada nivel de análisis. Se lo define como:

$$t_{AWT} = c \sigma_n - |AM - GM| \quad (39)$$

Donde AM es la media aritmética, GM es la media geométrica y $c = 2^{J-j}$.

IV. IMPLEMENTACIÓN

Para llevar a cabo el filtrado de ruido en imágenes es necesario realizar los siguientes pasos:

- 1) Agregar ruido a una imagen seleccionada. En los ejemplos que mostraremos, utilizaremos ruido gaussiano ya que es el más presente en todo tipo de señales.
- 2) Acondicionar la imagen para que sea cuadrada.
- 3) Calcular el nivel óptimo de descomposición según la wavelet elegida, y la dimensión de la imagen. Si bien podemos elegir un nivel arbitrario de descomposición, aumentarlo por encima del nivel óptimo no mejora el resultado. Esto solo produce un filtrado de la información útil de la imagen, suavizando los detalles.
- 4) Iterar entre los siguientes pasos:
 - i. Descomponer la imagen en sus matrices de detalle (horizontal, vertical y diagonal).
 - ii. Calcular el desvío estándar del ruido a partir de la matriz de detalle diagonal para cada nivel.
 - iii. Filtrar los coeficientes de las matrices de detalle obtenidas en el paso i., variando alguno de los siguientes parámetros:
 - Tipo de umbralización : hard o soft.
 - Algoritmos de umbralización.
 - Cantidad de niveles de descomposición.
 - Tipo de wavelet.
- 5) Reconstruir la imagen a partir de los coeficientes de detalle filtrados en los pasos previos y la matriz de aproximación.
- 6) Corroborar funcionamiento con imágenes con ruido no sintético.

V. RESULTADOS

A continuación, presentamos los resultados obtenidos para las siguientes imágenes de prueba:

Luego de pruebas con diferentes wavelet, algoritmos de umbralización y cantidad de niveles, se llegó a que resultado se obtuvo con *agregarlos mejores parametros*, obteniéndose las imágenes filtradas y reconstruidas.

Para corroborar este resultado presentamos las siguientes tablas comparativas de las distintas combinaciones de parámetros descritos en el ítem 4-iii en la sección IV para las imágenes *agregarlabel*

Con el fin de verificar el funcionamiento del filtrado con imágenes reales (es decir que el ruido presente en ellas no se agregó artificialmente), se filtraron las siguientes imágenes. Los resultados obtenidos para los siguientes parámetros *agregarparametros* son :

VI. CONCLUSIONES