

Compresión, basada en Wavelets, de modelos digitales de elevaciones

Rafael Arturo Trujillo Codornú
Mirelis Rasúa López

Resumen

En este artículo se presenta un esquema de compresión progresivo de modelos digitales de elevaciones, en dos etapas, basado en wavelets. Esta propuesta utiliza lo mejor de dos mundos; usa una técnica, de alto rendimiento y baja complejidad, basada en wavelet para la compresión con pérdidas en una primera etapa. En la segunda etapa el modelo reconstruido se resta del modelo digital de elevaciones original para obtener los residuales que son luego cuantizados uniformemente y codificados sin pérdida para garantizar una cota máxima del error en cada punto.

Palabras clave

Modelo digital de elevaciones; wavelet; compresión sin pérdida; compresión con pérdida.

Wavelet-based Compression for Digital Elevation Models

Abstract

This paper presents a wavelet-based two-stage progressive compression scheme to compress digital elevation models. This two-stage approach uses the best of two worlds: it uses a high performance and low complexity wavelet-based coding techniques for lossy compression as first stage; during the second stage the reconstructed model is subtracted from the original digital elevation model in order to obtain residuals which later are uniformly quantized and encode without loss to ensure the maximum height error in each point.

Keywords

Digital elevation model; wavelet; lossless compression; lossy compression.

1. INTRODUCCIÓN

Un modelo digital de elevaciones (MDE) es un conjunto estructurado de datos que define la distribución espacial de la altitud de la superficie del terreno. La unidad básica del MDE es un punto acotado, de un valor de altitud z y los valores de las coordenadas x, y .

Como representación de la superficie terrestre, los MDE constituyen uno de los más importantes tipos de datos geográficos. Las aplicaciones de estos modelos son muy variadas, entre las más importantes podemos mencionar la ortorrectificación de imágenes; cartografía de intervisibilidad; modelos hidrogeológicos o geomorfológicos, trazado de curvas de nivel en mapas topográficos y cálculos volumétricos de movimiento de tierras.

En cartografía, las altitudes suelen representarse mediante curvas de nivel y cotas. En función del tamaño de la zona cubierta, la mayoría de los MDE utilizan, para las pequeñas zonas, una red rectangular regular, mientras que para las zonas grandes utilizan una red pseudorectangular cuyos lados son meridianos y paralelos. Las redes rectangulares de elevación pueden ser procesadas fácilmente para obtener otros derivados, como las curvas de nivel y las pendientes, por ello en muchos países, como norma, se establece este formato para los modelos digitales de elevación. La representación de los MDE en forma de redes implica, sin embargo, el procesamiento de grandes volúmenes de información. Un modelo digital de elevaciones con una resolución de 5 m (como los requeridos para las ortofotos) en un área de 100 km² requeriría de alrededor de 9 GB de espacio.

Aunque los costos de almacenamiento de la información han descendido mucho, el procesamiento de grandes volúmenes de información presenta otros problemas, en particular el incremento del tiempo de procesamiento y el de transmisión. Estos tiempos son cruciales en aplicaciones sobre Internet y en aplicaciones de tiempo real (como las de

carácter militar) por todo lo cual reviste gran importancia el estudio de la compresión de los modelos digitales de elevaciones.

La compresión de datos se define como el proceso de reducir la cantidad de datos necesarios para reproducir eficazmente una información, es decir, la eliminación de datos redundantes. La operación de restaurar los datos originales de los datos comprimidos es llamada reconstrucción. Hay dos tipos de compresión: la compresión sin pérdida y la compresión con pérdida. En el primer caso el interés es reconstruir los datos de manera exacta sin pérdida de información. En el segundo caso se puede aceptar cierto error siempre y cuando la calidad (medida por algún criterio) de la reconstrucción sea aceptable.

Los compresores ordinarios de datos (como los compresores ZIP o RAR) son compresores sin pérdida que generalmente están basados en el algoritmo de Lempel-Ziv-Welch (Ziv & Lempel 1977) o sus modificaciones. Este algoritmo se basa en un análisis inicial del archivo para identificar secuencias repetidas que permitan armar un diccionario de equivalencias, asignando códigos breves a estas cadenas repetidas. Aunque pueden ser usados para reducir el tamaño de los MDE, no ofrecen altas tasas de compresión ya que los modelos digitales de elevación reflejan el cambio continuo de las ondulaciones del terreno y consecuentemente hay pocas secuencias repetidas. Por ello, y teniendo en cuenta el carácter, como regla suave, de los MDE se ha analizado el uso de los wavelets en la compresión.

La aplicación de los wavelets en las Geociencias ha tenido un gran auge en los últimos 15 años, principalmente asociado a los MDE. Marchuck & Simonov (2005) logran detectar cráteres de impacto en la superficie de la tierra mediante un filtrado especial del MDE basado en wavelets. Rucker & Fowler (2006) comprimen eficientemente los datos de la temperatura oceánica de una región arbitraria (no necesariamente rectangular) usando la Transformada Wavelet Discreta (TWD) 3D y

codificadores entrópicos adicionales. Zabala *et al.* (2007) analizan los efectos de la compresión wavelet de imágenes en la fotointerpretación. Varios trabajos se dedican al estudio de las mejores representaciones de los MDE basados en TWD y el posible impacto que puede tener la compresión con pérdida en la usabilidad de los datos y la calidad de derivados tales como las pendientes, las líneas de flujo, y las curvas de nivel. Oimoen (2004), utilizando compresión wavelet en su diseño más rudimentario, concluye que aún con tasas de compresión altas (del orden de 20 a 1) se obtienen MDE de calidad aceptable. McArthur *et al.* (2000) construyen bases de datos estructuradas jerárquicamente con múltiples niveles de detalle a partir del MDE, usando la descomposición wavelet, con el objetivo de detectar, de manera automática, qué zonas tienen una alta variabilidad (y por tanto requieren una triangulación más fina) y cuales tienen poca variabilidad. Basado en la observación de que el relieve conserva las mismas propiedades estadísticas sobre un amplio rango de escalas, Bjørke & Nilsen (2003) realizan una caracterización multiescala de los MDE basado en wavelets, mientras que Grazzini & Chrysoulakis (2005) aplican el análisis de fractales apoyado en el concepto de la auto similaridad estadística.

La tendencia dominante en la compresión de los MDE con wavelets ha sido utilizar, para los MDE, los mismos algoritmos bien establecidos para la compresión de imágenes (Ottonson 2001, Bjørke & Nilsen 2003, Oimoen 2004). En definitiva las imágenes y los MDE comparten muchas similitudes. Ambos son representables como matrices rectangulares que se diferencian solo en que, en un caso, el valor de cada celda especifica el color de un pixel de la imagen y en el otro la elevación de un punto del terreno sobre el nivel del mar. Pero al margen de estas similitudes existe una gran diferencia en cuan admisibles son los cambios en los datos. En el caso de las imágenes podemos aceptar variaciones relativamente grandes en la intensidad y color de los pixeles siempre y cuando el resultado sea visualmente indistinguible del original. Para los MDE se requiere algo más que la

percepción visual. Para los MDE se requiere que los derivados que se obtienen de los mismos no se alteren sustancialmente.

La mayoría de los esquemas de compresión de imágenes utilizan como medida de calidad el error medio de reconstrucción. Este criterio, a nuestro juicio no es adecuado para los MDE, ya que puede enmascarar errores importantes en la elevación de un punto siempre que el promedio de los errores en el modelo completo sea pequeño. Por otra parte una modificación significativa en la elevación de un punto puede alterar, por ejemplo, las líneas de drenaje del terreno. Como es conocido, la retención de la integridad hidrogeológica es una de las demandas más críticas en cualquier algoritmo que afecte los MDE.

Otros enfoques (por ejemplo, Zhen *et al.* 2008) proponen la aplicación de la transformada wavelet entera para lograr una compresión sin pérdidas o casi sin pérdidas de los MDE, sin embargo Yea & Pearlman (2006) han subrayado que, para imágenes, con esta variante se pierde eficiencia en la compresión y el flujo de bits progresivo no garantiza una disminución óptima de la distorsión.

En el presente trabajo se establece un esquema de compresión wavelet para los MDE, de bajo costo computacional, progresivo e incrustado, que a diferencia de los propuestos por Ottonson (2001) y Oimoen (2004) garantiza que el error de reconstrucción del MDE **en cada punto** sea menor que un valor prefijado previamente. Formulando matemáticamente el esquema que se propone garantiza que si M_{ij} es la matriz original del MDE y $\delta > 0$ una tolerancia prefijada, entonces la matriz \hat{M}_{ij} reconstruida luego de la compresión satisface:

$$\|M - \hat{M}\|_{\infty} = \max_{ij} |M_{ij} - \hat{M}_{ij}| < \delta$$

2. MATERIALES Y MÉTODOS

La transformada wavelet discreta (TWD) realiza una descomposición multirresolución que decorrela la señal separando la información en un

conjunto de bandas frecuenciales útiles para la posterior codificación (ver por ejemplo Fowler & Pesquet-Popescu 2007). La aplicación de la TWD descompone una señal unidimensional en una señal de baja frecuencia y otra de alta frecuencia.

Existen múltiples enfoques para la definición de los wavelets. Expondremos el que requiere menos prerequisites y que se basa en el esquema "lifting" (Figura 1). El esquema "lifting" introducido formalmente por Sweldens (1997) es un método bien conocido que consta de los siguientes pasos: Sea \bar{b}_0 un vector con un número par de coordenadas. Entonces;

- a. Se divide \bar{b}_0 en dos vectores; el vector de aproximación \bar{b} formado por los valores de \bar{b}_0 con índice par y el vector de detalles \bar{d} formado por los valores de \bar{b}_0 con índice impar.
- b. El paso de predicción. En este paso se predice el vector de detalles usando el vector de aproximación y un operador lineal de predicción P .

$$d_1[n] = \bar{d}[n] - P(\bar{b})[n]$$

Si la predicción es buena esta diferencia contiene mucha menos información que el conjunto original.

- c. El paso de actualización. En este paso se actualiza el vector de aproximación a partir del vector de detalles y un operador lineal de actualización U .

$$\bar{b}_1[n] = \bar{b}[n] - U(\bar{d}_1)[n]$$

Opcionalmente se pueden concatenar varios pasos de predicción y actualización y realizar una normalización de los arreglos resultantes.

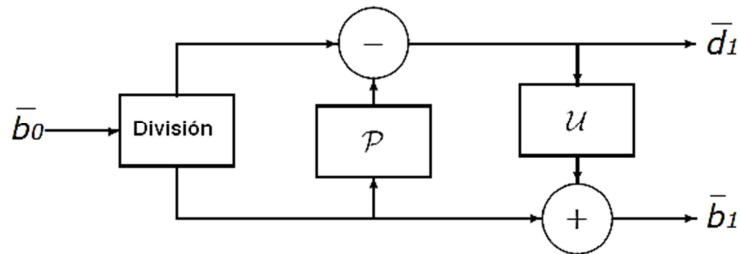


Figura 1. Fases del esquema "Lifting": División, Predicción y Actualización.

Como resultado de estos pasos se obtiene la pareja de arreglos (\bar{b}_1, \bar{d}_1) . La transformación depende de la selección de los operadores P y U . A diferencia de la transformada de Fourier, que utiliza senos y cosenos de manera exclusiva, existe una amplia variedad de bases de wavelet que pueden ser elegidas. Las más usadas en los MDE son las que pertenecen a la familia de wavelets ortogonales de Daubechies (DAUB4, DAUB6, DAUB8) y la familia de wavelets biortogonales Cohen-Daubechies-Feauveau (Cohen *et al.* 1992), en especial las funciones base CDF 5/3 y CDF 9/7 adoptadas por el nuevo estándar de compresión de imágenes JPEG2000. Oimoen (2004), por ejemplo, utiliza la función base DAUB6 para el análisis de los efectos de la compresión wavelet en MDE.

Para ilustrar el esquema "lifting" se muestran las fórmulas correspondientes a la transformada CDF 5/3. Las operaciones de predicción y actualización de esta transformada son:

$$\bar{d}_1[n] = \left(\bar{d}[n] - \frac{\bar{b}[n] + \bar{b}[n+1]}{2} \right) \quad \text{y} \quad \bar{b}_1[n] = \left(\bar{b}[n] + \frac{\bar{d}_1[n-1] + \bar{d}_1[n]}{4} \right)$$

Seguido de la normalización:

$$\bar{d}_1[n] = \frac{1}{\sqrt{2}} \bar{d}_1[n] \quad \text{y} \quad \bar{b}_1[n] = \sqrt{2} \bar{b}_1[n]$$

Como se aprecia, las fórmulas no son aplicables ni al primer elemento del vector \bar{d} ni al último elemento del vector \bar{b} . Para poder efectuarlas se asume una prolongación bilateral de ambos vectores simétrica o periódica. La transformada inversa se obtiene invirtiendo los pasos

anteriores. Para el ejemplo mostrado, la inversión se obtendría a través de las fórmulas:

$$\bar{d}_1[n] = \sqrt{2} \bar{d}_1[n] \quad \text{y} \quad \bar{b}_1[n] = \frac{1}{\sqrt{2}} \bar{b}_1[n]$$

$$\bar{b}[n] = \left(\bar{b}_1[n] - \frac{\bar{d}_1[n-1] + \bar{d}_1[n]}{4} \right) \quad \text{y} \quad \bar{d}[n] = \left(\bar{d}_1[n] + \frac{\bar{b}[n] + \bar{b}[n+1]}{2} \right)$$

El procedimiento de descomposición puede ser repetido de manera recursiva a la banda de más baja frecuencia (\bar{b}_1) para obtener múltiples niveles de descomposición de cada vez menor resolución.

2.1 Transformada Wavelet 2D

La transformada multidimensional se construye típicamente aplicando la transformada wavelet 1D independientemente a lo largo de cada dimensión del conjunto de datos, produciendo un número de sub-bandas.

En el caso bidimensional esta descomposición diádica de la TWD se muestra en la Figura 2. La matriz de datos original B_0 (o imagen) es descompuesta en 4 sub-bandas B_1 , H_1 , V_1 y D_1 , resultantes de aplicar la transformada Wavelet a cada fila y luego a cada columna. El conjunto B_1 contiene una imagen suavizada de la matriz mientras que H_1 , V_1 y D_1 son los detalles (reflejan cambios abruptos del valor de un coeficiente de la matriz a otro adyacente).

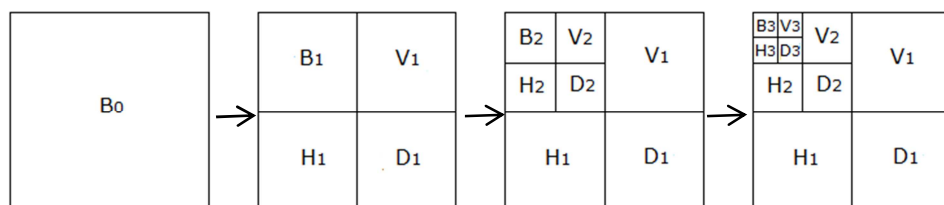


Figura 2. El arreglo original B_0 y el primer nivel, segundo nivel y tercer nivel de la transformación diádica wavelet.

Cada sub-banda es de un cuarto del tamaño de la matriz original. Luego la sub-banda de aproximación B_1 puede ser transformada para obtener B_2 y detalles adicionales H_2 , V_2 y D_2 donde los subíndices

reflejan el nivel de la transformación. A su vez es posible transformar B_2 y así sucesivamente.

De por sí, la TWD es completamente reversible, pero la oportunidad de una compresión importante aparece debido a que muchos valores de la matriz transformada son pequeños en amplitud y consecuentemente aportan poca información. Si la matriz original de datos representa una función "suave" (o sea, que varía poco de un punto a los adyacentes), la energía de la matriz transformada se concentrará en la banda B_n y los coeficientes en las bandas de detalles serán pequeños. Si inicialmente, en un MDE típico, hay una distribución relativamente uniforme de las elevaciones en un rango dado, luego de la transformada wavelet 2D la mayoría de los coeficientes tienen magnitudes cercanas a cero (ver Figuras 3 y 4).

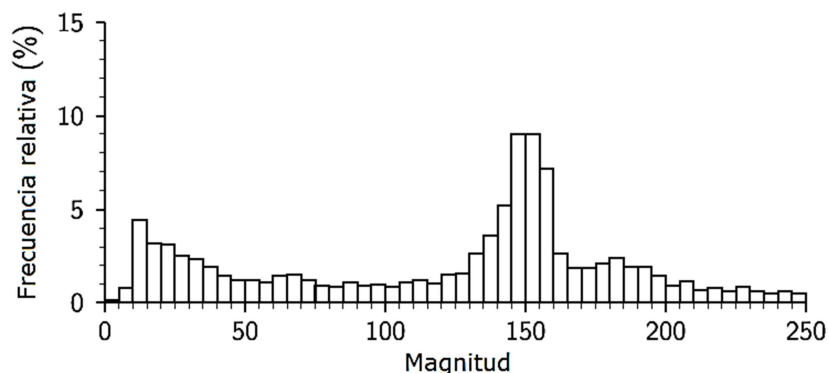


Figura 3. Distribución estadística de las magnitudes de una imagen patrón (tomado de Pearlman & Said 2008).

Muchos de estos coeficientes podrán codificarse con muy pocos bits o simplemente ignorarse. Los ceros que aparecerán en las bandas de detalles pueden comprimirse con compresores de rachas (ver, por ejemplo, Costa & Malvar 2003). Por ello se obtienen tasas de compresión muy altas conservando, al mismo tiempo, buena calidad en la reconstrucción del arreglo original.

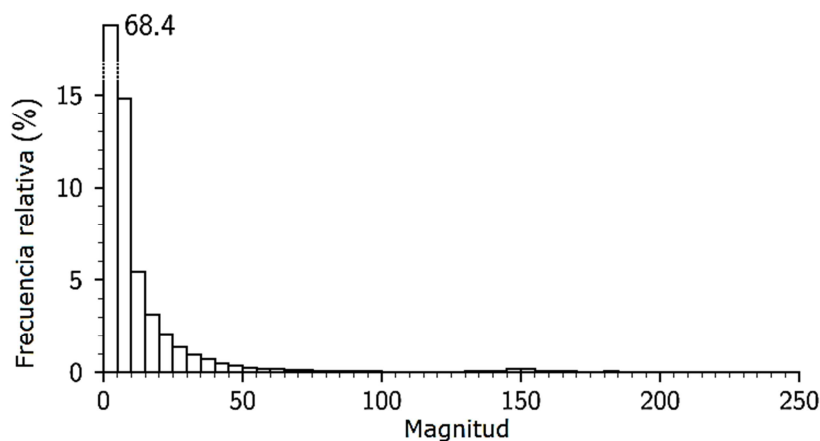


Figura 4. Distribución estadística de las magnitudes de los coeficientes de la TWD de la imagen patrón de la Figura 3.

2.2 Compresión de datos mediante wavelets

La compresión de datos con wavelets involucra varios pasos. En primer lugar debe aplicarse la transformada wavelet a los datos originales, con tantos niveles de descomposición como se determine (generalmente 5 o 6 niveles son suficientes). Luego se aplica un umbral al conjunto de coeficientes wavelets obtenido. El umbral indica el valor absoluto mínimo que deben tener los coeficientes para que no sean considerados insignificantes y sean despreciados haciéndolos igual a cero. Este paso obviamente es irreversible y es opcional. El tercer paso consiste en cuantizar los coeficientes umbralizados. La cuantización permite transformar la matriz de números reales resultante del paso anterior a una matriz de números enteros que sirve de insumo al próximo paso. En el caso de la cuantización lineal uniforme, dado $\Delta > 0$ cada coeficiente C_{ij} se cuantiza de acuerdo con:

$$Q_{ij} = \frac{C_{ij}}{2\Delta} + \frac{1}{2} \text{sign}(C_{ij})$$

donde $\lfloor x \rfloor$ representa la parte entera de x . El valor de Δ es una medida del error máximo que se introduce en la cuantización. En este caso el valor de C_{ij} se puede restaurar a partir del coeficiente cuantizado a través de:

$$C_{ij} = 2\Delta Q_{ij}$$

Obviamente la cuantización produce un efecto de truncamiento irreversible en cada coeficiente pero los errores que se introducen son, como regla, menores que los que pueden producirse en la umbralización.

Por último, la matriz de enteros obtenida tras la cuantización es comprimida y convertida en un flujo de bits utilizando algún codificador (códec) que explote la correlación existente. Si el flujo de bits que produce el códec satisface la propiedad de que la recepción puede ser detenida en cualquier punto y a partir del flujo recibido se pueden descomprimir y reconstruir, con cierta precisión, los datos se dice que el flujo es *incrustado*. Los flujos incrustados permiten la escalabilidad. El flujo comprimido es *progresivo* si la representación progresivamente mejora la calidad (reduce la distorsión) de la reconstrucción a medida que se reciben más datos. Esto se logra si la información se transmite en orden de importancia.

La restauración de los datos originales se obtiene realizando las transformaciones inversas en cada uno de los pasos mencionados anteriormente a excepción de la umbralización que es totalmente irreversible.

Dentro de los esquemas de codificación de los coeficientes cuantizados uno de los más usados es la codificación por planos de bits.

2.3 Codificación por planos de bits

En la codificación por planos de bits se almacenan los coeficientes wavelet cuantizados de manera que los bits más significativos se transmitan primero. Existen algoritmos sofisticados para realizar esta tarea en imágenes, entre ellos se pueden mencionar SPIHT, SPECK y EBCOT, sin embargo el algoritmo que se propone es sencillo y alcanza tasas de compresión cercanas a los mencionados (ver, por ejemplo, Dunn 2006). Dado un $n > 0$ se puede definir la función $S_n(i, j)$ que toma el valor de 1 si en la representación binaria de $|Q_{ij}|$ el n -ésimo bit es igual a 1 y cero en caso contrario. Se asume que el bit 0 es el menos

significativo del número. Al conjunto $P_n = \{S_n(i, j) : \forall i, j\}$ se le denomina plano de bit. Si se almacenan los planos de bits en orden decreciente de n se garantiza obviamente la trasmisión progresiva. El algoritmo para la codificación plano a plano es el siguiente:

1. *Inicialización*: Codificar y llevar al flujo de salida el valor $n = \log_2(\max_{i,j} |Q_{ij}|)$. Crear la lista LCI (lista de coeficientes insignificantes) que contiene inicialmente todos los coeficientes wavelets cuantizados Q_{ij} . Crear la lista vacía LCS (lista de coeficientes significativos). El orden interno de los coeficientes es tal que, dentro de cada banda, se ordenan por filas, las bandas, a su vez, se ordenan de la siguiente manera $B_n, V_n, H_n, D_n, V_{n-1}, H_{n-1}, D_{n-1}, \dots, V_1, H_1, D_1$ (ver Figura 5).
2. *Fase de Ordenamiento*: Para cada coeficiente Q_{ij} que pertenezca a la lista LCI codificar y llevar al flujo de salida $S_n(i, j)$. Si $S_n(i, j) = 1$ entonces codificar y dar salida a su signo, sacar el coeficiente de la lista LCI y pasarlo a la lista LCS ya que pasó a ser significativo para ese plano de bit.
3. *Fase de Refinación*: Para cada coeficiente de la lista LCS que fue incluido en el plano anterior (no en el plano actual) codificar $S_n(i, j)$.
4. *Actualización del Plano de bits*: Se decrementa n y si $n \geq 0$ se pasa al punto 2.

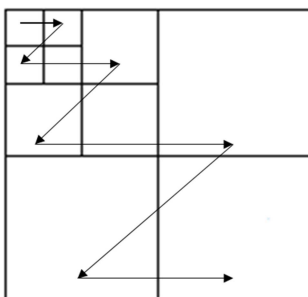


Figura 5. Orden en "zigzag" en que se organizan las bandas de los coeficientes wavelet cuantizados.

En la figura 6 se ilustra como los coeficientes pasan a ser significativos y los que están en la fase de refinación (pertenecen a la lista LCS).

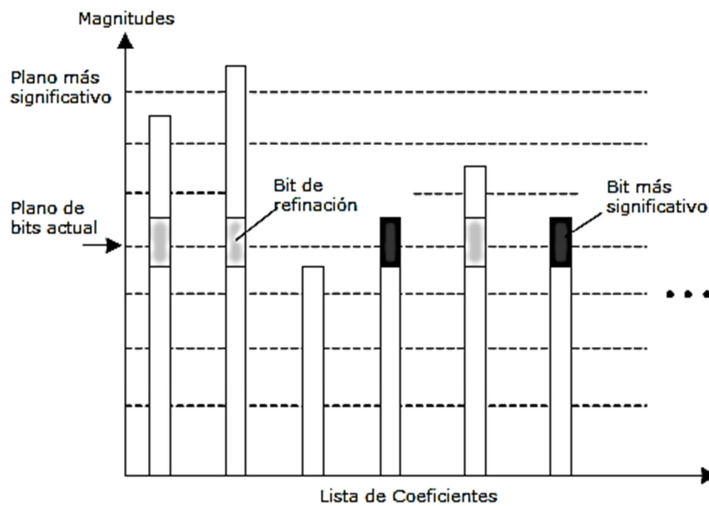


Figura 6. Codificación plano a plano de los coeficientes wavelets cuantizados. Los coeficientes que contienen rectángulos grises pertenecen a la lista LCS y sólo se refinan. Los que contienen rectángulos negros se han detectado como significativos en esa iteración y serán pasados de la lista LCI a la lista LCS. Los demás se mantienen en la lista LCI. (Tomado de Dunn 2006)

Debe subrayarse que los bits que se llevan al flujo de salida en la fase de ordenamiento son, al menos para los planos más significativos, en su gran mayoría ceros. Esto se desprende de la distribución probabilística mostrada en la Figura 4. Consecuentemente estos bits pueden ser efectivamente comprimidos codificando las rachas de ceros resultantes. Por otra parte, en la fase de refinación, la probabilidad de que se tenga a la salida un 1 o un 0 es aproximadamente igual, por lo que intentar comprimir esta secuencia no ofrece muchos resultados.

2.4 Códigos de Golomb-Rice

Un codificador de rachas simple y muy efectivo es el basado en los códigos de Golomb con parámetro g . Este código es un código de longitud variable que a cada racha de longitud r le hace corresponder una secuencia de bits formada por 2 componentes:

- Un prefijo unario de tantos bits como $\lfloor r/g \rfloor$
- Un sufijo binario donde se codifica $r \bmod g$

Un caso particularmente simple de los códigos de Golomb son los llamados códigos de Golomb-Rice que se obtienen cuando $g = 2^k$ para algún entero k (ver Tabla 1). En este caso el sufijo del código siempre es la representación con k bits del número $r \bmod 2^k$

Tabla 1 . Código de Golomb-Rice (GR) con parámetro k . La longitud del prefijo es $p = \lfloor r/2^k \rfloor$. El sufijo z es la representación con k bits del resto de la división de r por 2^k

Valor de entrada	Flujo de bits de salida
r	$\underbrace{111 \dots 11}_p 0 \underbrace{b_{k-1} b_{k-2} \dots b_0}_k$ <p>Prefijo de p bits Sufijo z de k bits</p>

La eficiencia de la codificación de las rachas con los códigos de Golomb-Rice depende del parámetro k y de la distribución de probabilidades de la longitud de las rachas. Como esta distribución puede variar de banda a banda se han propuesto códigos de Golomb-Rice adaptativos (Malvar 1999) que van modificando el parámetro k en la medida que se van codificando las rachas.

En general, las ventajas, del uso de los códigos de Golomb-Rice adaptativos para la codificación en la fase de ordenamiento incluyen la simplicidad, la eficiencia computacional de los códigos y la eficacia con que se adaptan a los cambios en la distribución de probabilidades de las rachas.

Es bien conocido que las magnitudes $|Q_{ij}|$ tienen cierta correlación espacial. Por ejemplo la probabilidad de que un coeficiente sea significativo en un plano de bits aumenta si algunos de los coeficientes vecinos en la matriz son ya significativos. Estas consideraciones indican que en vez de manejar una sola lista de coeficientes insignificantes podrían utilizarse dos o más listas que creen sub-secuencias en el flujo de bits con una distribución más uniforme. Por ejemplo podría definirse una lista de coeficientes insignificantes con vecinos significativos y una lista de coeficientes insignificantes con vecinos insignificantes. Las alternativas

posibles en este sentido y su impacto en la compresión de imágenes pueden verse en Ordentlich *et al.* (1998)

3. RESULTADOS Y DISCUSION

3.1 Descripción del esquema propuesto

El presente trabajo propone un esquema en dos etapas. Esta variante sigue los pasos propuestos por Marpe *et al.* (2000) y Yea & Pearlman (2008). En una primera etapa se realiza una compresión wavelet con pérdidas del MDE utilizando un esquema de codificación plano a plano que garantiza un flujo de bits incrustado y progresivo. En la segunda etapa se codifican los residuos resultantes de la diferencia entre el MDE original y el MDE reconstruido a partir del flujo de bits generado por la primera etapa.

Se ha observado (ver Yea & Pearlman 2008) que si la primera etapa del algoritmo realiza una compresión muy precisa (dejándole poco trabajo a la segunda etapa), o una compresión muy burda (delegando la mayor parte del trabajo a la segunda etapa) no se obtienen tasas óptimas de compresión. Esto sugiere que existe un punto óptimo de balance entre la compresión que debe realizar cada una de las capas. Este balance óptimo fue encontrado, para imágenes, por Yea & Pearlman (2008) y es el que se propone para la compresión de los MDE. Para ello debe determinarse la entropía de los residuales en el dominio de las transformadas Wavelets $H_1(\hat{R})$ y la entropía de los residuales en el dominio original $H_1(R)$.

En efecto, sean M_{ij} los coeficientes de la matriz del MDE y Q_{ij} los coeficientes cuantizados después de varios niveles de la transformada wavelet. Sea \tilde{Q} la imagen de la matriz Q obtenida a partir de los primeros N bits del flujo progresivo obtenido por la codificación plano a plano expuesta en 2.3. Las diferencias $Q_{ij} - \tilde{Q}_{ij}$ son los residuales en el dominio de los wavelets. Si se trata ahora de restaurar M a partir de \tilde{Q} se obtiene una imagen \tilde{M} de M . Las diferencias $M_{ij} - \tilde{M}_{ij}$ cuantizadas son los residuales en el dominio original. A medida que el número de bits N utilizado para \tilde{Q} se incrementa, la entropía de primer orden de los

residuales en el dominio de los wavelets $H_1(\hat{R})$ va aumentando hasta que iguala a la entropía de primer orden de los residuales en el dominio original $H_1(R)$. Llegado a ese punto los residuales en el dominio wavelet pasan a ser más aleatorios que los del dominio original y debe abandonarse la compresión wavelets de los residuales pasando a comprimirlos en el dominio original. Es importante destacar que Yea & Pearlman (2008) establecen un algoritmo computacionalmente simple que permite ir controlando ambas entropías en la medida que se codifican los coeficientes.

3.2 Algoritmo propuesto

Dados la matriz M del modelo digital del terreno y un error máximo δ :

1. Realizar la transformada wavelet con 5 niveles de descomposición utilizando el filtro CDF 9/7.
2. Cuantizar los coeficientes de la matriz obtenida con paso de cuantización δ .
3. Codificar plano a plano los coeficientes cuantizados de acuerdo al algoritmo mostrado en el punto 2.3. Cada vez que se encuentre un nuevo coeficiente significativo o se refine un coeficiente ya significativo, actualizar, de acuerdo al esquema de Yea & Pearlman (2008), las entropías de los residuales en el dominio de los wavelets y en el dominio original. Si se alcanza el valor crítico detener la codificación plano a plano y pasar al punto 4. Comprimir las rachas de ceros en la fase de ordenamiento de las bandas de detalles con códigos de Golomb adaptativos. Los bits de ordenamiento de la banda de aproximación B_n no se comprimen con códigos de Golomb ya que es en esta banda donde se concentran los coeficientes de mayor valor absoluto de la matriz.
4. Decodificar los coeficientes cuantizados con el flujo de bit parcial para obtener la matriz \tilde{Q} .
5. Aplicar las transformaciones inversas para obtener la matriz \tilde{M} que aproxima el MDE original.

6. Si los residuales $M_{ij} - \tilde{M}_{ij}$ son, todos, en valor absoluto, menores que δ finalizar, en caso contrario pasar al punto 7.
7. Cuantizar los residuales $M_{ij} - \tilde{M}_{ij}$ con paso de cuantización δ y codificarlos plano a plano sin códigos de Golomb.

3.3 Evaluación del algoritmo

Para la evaluación del algoritmo se utilizó un modelo digital de elevaciones de la zona de Moa, consistente en una matriz cuadrada de 128 x 128. Se comprimió el MDE utilizando;

1. Transformada Wavelet DAUB6, con umbralización para retener un porcentaje de los coeficientes y codificación de los ceros resultantes con un compresor de rachas (algoritmo A). Este algoritmo es analizado por Oimoen (2004).
2. Cuantificación del MDE con paso δ , utilización de la transformada wavelet entera 2/6 y compresión con el codec de JPEG2000 (Algoritmo B). Este algoritmo es el utilizado por Zheng *et al.* (2008).
3. Algoritmo de dos capas propuesto en el presente trabajo (Algoritmo C).

Los criterios de evaluación usados se muestran en la Tabla 2.

Tabla 2 . Criterios de evaluación de los esquemas de compresión. L es el tamaño (en bits) del MDE original M y L_C el tamaño (en bits) del MDE comprimido. \hat{M} es la matriz reconstruida y \bar{M} es el valor medio de las elevaciones.

Tasa de Compresión	Error cuadrático medio	Error máximo
$CR = \frac{L}{L_C}$	$MSR = \frac{1}{(128)^2} \sum (M_{ij} - \hat{M}_{ij})^2$	$\ M - \hat{M}\ _{\infty} = \max_{ij} M_{ij} - \hat{M}_{ij} $

Los resultados obtenidos se muestran en la Tabla 3. Puede apreciarse que para altas tasas de compresión, el algoritmo que se propone genera menor distorsión, lo que se explica, por la eficiencia del filtro CDF 9/7. Para tasas de compresión bajas y consecuentemente bajas distorsiones se comporta mejor el algoritmo B aunque la diferencia con el que se propone es pequeña. El algoritmo A queda relegado en todos los análisis

fundamentalmente por su carácter no progresivo. Los algoritmos B y C garantizan que, si se utilizan todos los bits del flujo comprimido, el error máximo en cada punto sea menor que un valor dado.

Tabla 3. Resultados de la comparación entre los tres esquemas. Se tomó una tolerancia mínima de $\delta = 0,5$. Se truncó el flujo incrustado en los algoritmos B y C para obtener diferentes tasas de compresión. En negritas los mejores valores. La primera fila corresponde a la compresión utilizando todos los bits del flujo.

CR	Esquema	MSR	$\ M - \hat{M}\ _{\infty}$
5,11	A	0,65	0,51
7,53	B	0,42	0,49
7,27	C	0,42	0,49
11	A	0,81	2,48
	B	0,67	1,94
	C	0,68	1,96
15	A	3,14	5,52
	B	1,99	3,97
	C	1,97	3,98
20	A	9,02	10,60
	B	6,34	7,96
	C	5,60	7,94
30	A	12,12	16,78
	B	9,71	11,89
	C	8,65	11,87

4. CONCLUSIONES

El algoritmo presentado en este trabajo para la compresión de Modelos Digitales de Elevaciones ofrece buenas prestaciones en todas las tasas de compresión, lo que lo hace factible, tanto para aplicaciones de alta exigencia, como para aplicaciones de visualización que no requieren una gran precisión en la reconstrucción. Es un algoritmo computacionalmente simple que ofrece un código progresivo e incrustado. Puede ser mejorado (a costa de una complejidad computacional extra) si se utilizan subsecuencias en la fase de codificación plano a plano que exploten la correlación espacial que

existe entre los coeficientes y si se añade una codificación aritmética al flujo de bits de salida.

5. REFERENCIAS

- BJØRKE, J.T.; NILSEN, N.** 2003: Wavelets applied to simplification of digital terrain models. *International Journal of Geographical Information Science*, 17(7): 601– 621.
- COHEN, A.; DAUBECHIES I., FEAUVEAU J.C.** 1992: Biorthogonal Bases of Compactly supported wavelets. *Comm Pure and Applied Math*, 45: 485-560.
- COSTA, M.H; MALVAR H.S.** 2003: Efficient Run-Length Encoding of Binary Sources with Unknown Statistics. *Microsoft Research Technical Report*, MSR-TR-2003-95.
- DUNN, C.;** 2006: Scalable Bitplane Runlength Coding. *120th Convention of Audio Engineering Society*, Paris, France, May 20-23, 2006, Convention Paper 6749.
- FOWLER, J; PESQUET-POPESCU, B.** 2007: An Overview on Wavelets in Source Coding, Communications, and Networks. *EURASIP Journal on Image and Video Processing*, Article ID 60539: 1-27
- GRAZZINI, J.; CHRYSOULAKIS, N.** 2005: Extraction of Surface Properties from a High Accuracy DEM Using Multiscale Remote Sensing Techniques. *Proc. of the 19th Conf. Informatics for Environmental Protection*: 352-356.
- MALVAR, H.S.;** 1999: Fast progressive wavelet coding. *Proc. Data Compression Conf.*, Snowbird, Utah, Mar. 1999: 336–343.
- MARCHUK, AN.G.; SIMONOV, K.V.** 2005: Detecting possible impact craters on the Earth's surface using the DEM data processing. *Bull. Nov. Comp. Center, Math. Model in Geoph.*, 10: 59-66.
- MARPE, D.; BLATTERMANN, G.; RICKE J.; MAAB P.** 2000: A two-layered wavelet-based algorithm for efficient lossless and lossy image compression. *IEEE Trans. Circuits Syst. Video Technol.*, 10: 1094-1102.
- MCARTHUR, D.E; FUENTES, R.W.; DEVARAJAN ,V.** 2000: Generation of Hierarchical Multiresolution Terrain Databases Using Wavelet. *Photogrammetric Engineering & Remote Sensing*, 66(3): 287-295.

- OIMOEN, M.J.** 2004: The effects of wavelet compression on digital elevation models (DEMs) *Geoscience and Remote Sensing Symposium, IGARSS '04 Proceedings. IEEE International* , ISBN: 0-7803-8742-2.
- ORDENTLICH, E.; WEINBERGER, M; SEROUSSI, G.** 1998: A Low-Complexity Modeling Approach for Embedded Coding of Wavelet Coefficients. *Proc. Data Compression Conf.*, Snowbird, Utah, Mar. 1998: 408–417.
- OTTOSON, P.** 2001: Compressing Digital Elevation Models with Wavelet Decomposition. *ScanGIS*, 15-31.
- PEARLMAN, W.A.; SAID A.** 2008: Set Partition Coding: Part I of Set Partition Coding and Image Wavelet Coding Systems. *Foundations and Trends in Signal Processing*. 2(2): 95–180.
- RUCKER, J.; FOWLER, J.** 2006: Shape-Adaptive Embedded Coding of Ocean-Temperature Imagery. *Proc. of the 40th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA*: 1887-1891
- SWELDENS, W.** 1997: The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2) 511-546.
- YEA, S.; PEARLMAN, W.A.** 2006: A Wavelet-Based Two-Stage Near-Lossless Coder. *IEEE Transactions on Image Processing*, 15(11): 3488-3500
- ZABALA, A.; MODUGNO S., PONS X., BURRIEL J.A., PINO J.** 2007: Efecto de la Compresión Wavelet de Imágenes en su Fotointerpretación. *7th Geomatic week, Barcelona*, 20-23 February.
- ZHENG, J.J.; FANG, J.Y.; HAN, C.D.** 2008: The Selection of Reversible Integer-to-Integer Wavelet Transforms for DEM Multi-Scale Representation and Progressive Compression. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. XXXVII. Part B4. Beijing
- ZIV, JACOB; LEMPEL, JACOB.** 1977: A Universal Algorithm for Sequential Data Compression. *IEEE Transactions on Information Theory* 23 (3): 337–343.

Rafael Arturo Trujillo Codornú

Doctor en Ciencias Matemáticas. Profesor Titular.
Departamento de Matemáticas e Informática.
Instituto Superior Minero Metalúrgico.

rtrujillo@ismm.edu.cu

Mirelis Rasúa López

Licenciada en Matemáticas. Profesora Asistente.
Departamento de Matemáticas e Informática.
Instituto Superior Minero Metalúrgico.

mrasua@ismm.edu.cu