

METODO RAPIDO DE COMPRESION FRACTAL MULTI-RESOLUCION.

**A. Martínez Ramírez, *A. Díaz Sánchez, *M. Linares Aranda, **J. Vega Pineda*

*INAOE Instituto Nacional de Astrofísica Óptica y Electrónica
Luis Enrique Erro No.1 Tonantzintla, Puebla, C.P. 72840, México
Tel., Fax +52 (222) 2 47 05 17

**Instituto Tecnológico de Chihuahua
Ave. Tecnológico 2909, Chihuahua, Chih., C.P. 31310, México
aramirez@susu.inaoep.mx, adiazsan@inaoep.mx, mlinares@inaoep.mx, jvega@itchihuahua.edu.mx

ABSTRACT

Nowadays fractal image coding is performed by several methods focused to reduce the number of comparisons. Some methods use block classification to reduce the number of comparisons. In this work, a novel method of fractal image compression is described. The method uses a quadtree scheme on multiple levels of image resolution and the fractal coding is applied taking advantage of the multiresolution decomposition. The simplicity and regularity of the method make it suitable to be implemented on programmable logic devices, such as FPGAs, or in custom VLSI integrated circuits.

RESUMEN

Actualmente la compresión fractal de imágenes es realizada por varios métodos enfocados a reducir el número de comparaciones. Algunos de los métodos usan clasificación de bloques para reducir el número de comparaciones. En este trabajo se describe un método nuevo para hacer la compresión fractal de imágenes. El método usa un esquema de particionamiento en árbol cuádruple sobre múltiples niveles de resolución de la imagen y la codificación fractal es aplicada aprovechando las ventajas de la descomposición multiresolución. La simplicidad y regularidad del método lo hace adecuado para ser implementado en dispositivos lógicos programables, tales como FPGAs, o en circuitos integrados VLSI.

METODO RAPIDO DE COMPRESION FRACTAL MULTI-RESOLUCION.

*A. Martínez Ramírez, *A. Díaz Sánchez, *M. Linares Aranda, **J. Vega Pineda

**INAOE Instituto Nacional de Astrofísica Optica y Electrónica
Luis Enrique Erro No.1 Tonantzintla, Puebla, C.P. 72840, México
Tel., Fax +52 (222) 2 47 05 17*

***Instituto Tecnológico de Chihuahua
Ave. Tecnológico 2909, Chihuahua, Chih., C.P. 31310, México
aramirez@susu.inaoep.mx, adiazsan@inaoep.mx, mlinares@inaoep.mx.*

RESUMEN

Actualmente la compresión fractal de imágenes es realizada por varios métodos enfocados a reducir el número de comparaciones. Algunos de los métodos usan clasificación de bloques para reducir el número de comparaciones. En este trabajo se describe un método nuevo para hacer la compresión fractal de imágenes. El método usa un esquema de particionamiento en árbol cuádruple sobre múltiples niveles de resolución de la imagen y la codificación fractal es aplicada aprovechando las ventajas de la descomposición multiresolución. La simplicidad y regularidad del método lo hace adecuado para ser implementado en dispositivos lógicos programables, tales como FPGAs, o en circuitos integrados VLSI.

1. INTRODUCCION

La codificación de una imagen I en escala de grises mediante el método de codificación fractal propuesto por Jacquin [1] separa a I en $N_R \times N_R$ bloques rangos $r_{i,j}$ de $n \times n$ píxeles, como se indica en la ecuación (1).

$$I = \bigcup_{i,j=1}^{N_R} r_{i,j} \quad r_{i,j} \cap r_{\hat{i},\hat{j}} = \emptyset \quad \text{usando } (i,j) \neq (\hat{i},\hat{j}) \quad (1)$$

Cada uno de los bloques rangos es aproximado por un bloque dominio $d_{k,l}$ el cual es obtenido de la misma imagen I por medio de una transformación contractiva w_u . Cada bloque dominio tiene $2n \times 2n$ píxeles y se traslapan por la mitad. La transformación w_u es indicada en (2):

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = w_u \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & s_u \end{bmatrix} \begin{bmatrix} a_u & b_u & 0 \\ c_u & d_u & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_u \\ f_u \\ 0_u \end{bmatrix} \quad (2)$$

donde (x,y) son puntos pertenecientes a un bloque dominio y (x',y') son puntos pertenecientes a un bloque dominio transformado. El valor de z es la luminosidad del píxel en el punto (x,y) en el bloque dominio y z' es la luminosidad del píxel en el punto (x',y') en el bloque dominio transformado. Para cada uno de los bloques rangos se selecciona el bloque dominio transformado que mejor se le asemeje. El parámetro s_u sirve para transformar el contraste y el parámetro o_u es usado para ajustar el brillo. El índice h se usa para escoger el conjunto de parámetros a_u, b_u, c_u y d_u , los cuales seleccionan una de 8 transformaciones isométricas [2].

2. INDEPENDENCIA DE LA RESOLUCION

Una vez que la imagen I ha sido codificada, obteniendo un conjunto de parámetros para cada una de las transformaciones contractivas w_u , para $u=1,\dots,R$, donde R es el número de rangos en los cuales se ha dividido la imagen, esta puede ser recobrada mediante la ejecución iterada del conjunto de transformaciones w_u , las cuales forman el código fractal de la imagen. El código fractal no hace ninguna referencia a la resolución de la imagen original I, así que puede ser iterado sobre una resolución diferente. Se obtiene un buen resultado cuando la resolución de la imagen decodificada es menor o igual que la resolución de la imagen original. Si el código fractal se ejecuta sobre una resolución mayor que la de la imagen original, se obtiene un resultado aceptable cuando este nivel de resolución duplica tanto la resolución horizontal como la vertical de la imagen original [3,4]. La ventaja de codificar a la imagen I en un nivel de resolución menor es que si la imagen I_s , de menor resolución, es particionada en un número igual de bloques rangos, y si se usa en ella una distribución de bloques similar a la usada en la imagen I, entonces I_s puede

codificarse usando bloques rangos que contienen un número menor de píxeles que aquellos bloques con los que se codifica la imagen I. Esto reduce el número de comparaciones entre píxeles cuando se busca la similitud de un bloque rango con un bloque dominio transformado.

3. BUSQUEDA LOCAL DE SIMILITUD

Para reducir el número de comparaciones necesarias para codificar cada bloque rango de $p \times p$ píxeles, la búsqueda de similitud será limitada a aquellos bloques dominios que están dentro de una ventana de $(2L+1) \times (2L+1)$ bloques dominio, con $0 \leq L < (N_R/2)$, estando centrada en la misma posición que el bloque rango que está siendo codificado. La reducción de los dominios disponibles permite que el número de comparaciones de bloque por cada bloque rango sea ahora de $8(2L+1)^2$. El número de comparaciones entre píxeles, necesarias para codificar la imagen I, es $8(2L+1)^2 p^2$, y el número de comparaciones entre píxeles necesarias para codificar I_s es $8(2L+1)^2 (p/2)^2$. La calidad de la imagen se reduce, pero puede ser aceptable si se utiliza un valor de L suficientemente grande, por ejemplo, $L \geq 3$.

4. USANDO LA PARTICION EN ARBOL CUADRUPLE SOBRE MULTIPLES NIVELES DE RESOLUCION

4.1. Descripción del método propuesto

La partición en árbol cuádruple es usada para incrementar la razón de compresión, manteniendo una calidad de imagen aceptable en la imagen decodificada. Generalmente se usan bloques rangos grandes para aquellas zonas de la imagen que presentan poca variación y son de baja frecuencia; se usan bloques rangos medianos para aquellas zonas que presentan variación media y bloques rangos pequeños para aquellas zonas de la imagen que presentan mucha variación y alta frecuencia. Nuestra propuesta consiste en la aplicación de la partición en árbol cuádruple a través de los diferentes niveles de resolución de una pirámide de imágenes obtenidas a partir de la reducción de la resolución de la imagen original I, y aplicando un nivel de partición sobre un nivel de resolución de la pirámide. Este procedimiento se ilustra en la figura 1 usando tres niveles de resolución de imagen. Empezando con una imagen I cuya resolución es de $P \times P$ píxeles, considérense las imágenes I_{s_1} y I_{s_2} con resoluciones de $(P/2) \times (P/2)$ y $(P/4) \times (P/4)$ píxeles, respectivamente. La imagen I_{s_1} es obtenida submuestreando en 2 los renglones y las columnas de la imagen I. La imagen I_{s_2} se obtiene submuestreando en 2 los renglones y las columnas de I_{s_1} . Sobre la imagen I se

aplica una partición en $N_R \times N_R$ bloques rangos. Sobre la imagen I_{s_1} se aplica una partición en $(N_R/2) \times (N_R/2)$ bloques rangos, y la imagen I_{s_2} es partida en $(N_R/4) \times (N_R/4)$ bloques rangos. La partición en $(N_R/4) \times (N_R/4)$ bloques rangos representa el primer nivel en la partición en árbol cuádruple, y con los bloques de esta partición se cubrirán aquellas zonas de la imagen que presentan poca variación. En el método que proponemos este nivel de particionamiento es asignado a la imagen I_{s_2} , la cual tiene la menor resolución. La partición en $(N_R/2) \times (N_R/2)$ bloques rangos es el segundo nivel en la partición en árbol cuádruple, y con los bloques de esta partición se cubrirán aquellas zonas que presentan un nivel medio de variación. En nuestro método este nivel de partición es asignado a la imagen I_{s_1} . La partición en $N_R \times N_R$ bloques rangos es el tercer nivel de particionamiento en árbol cuádruple, y es usado para cubrir aquellas zonas en la imagen que presentan variaciones grandes. En nuestro método este nivel de partición se lleva a cabo sobre la imagen I, la cual presenta los detalles de mayor variación. Se obtiene un código fractal para cada nivel de resolución y se seleccionan aquellos códigos que aproximan bien, dentro de un margen de error establecido, a algunas zonas en esa imagen. Los códigos seleccionados en ambas sub-imágenes I_{s_1} e I_{s_2} , junto con los códigos seleccionados en la imagen I, serán usados para formar el código fractal para la imagen completa I.

La aproximación fractal de la imagen en cada nivel de resolución sirve como un separador de bloques de acuerdo al contenido de variación. El código fractal que se obtiene sobre la imagen I_{s_2} nos permite aproximar bloques de imagen que presentan pocas variaciones cuando ellos son desplegados en la resolución de la imagen original. El código fractal que es obtenido sobre la imagen I_{s_1} nos permiten aproximar bloques que presentan variaciones medias cuando son desplegados en la misma resolución que la imagen original I, y los códigos fractales obtenidos sobre la imagen original aproximan los bloques que tienen alta variación. Para hacer más eficiente en rapidez a este método se propone que el proceso de particionamiento y clasificación de bloques en niveles de variación sea previo al proceso de codificación fractal de los bloques en cada nivel de resolución de la imagen.

La ventaja de este método comparado con el de particionamiento en árbol cuádruple sobre un solo nivel de resolución es que en este método los bloques rangos son más pequeños desde el primero hasta el penúltimo nivel de la partición. Se logra reducir el número de operaciones de comparación entre píxeles que se lleva a cabo para codificar cada bloque rango cuando se busca similitud con los bloques dominios. Además, en el método que proponemos, todos los bloques rangos en los diferentes niveles de resolución son de igual tamaño, y

esto es particularmente importante cuando el método será implementado ya sea usando dispositivos lógicos programables o en circuito integrado VLSI, debido a que se obtiene regularidad de los bloques de procesamiento.

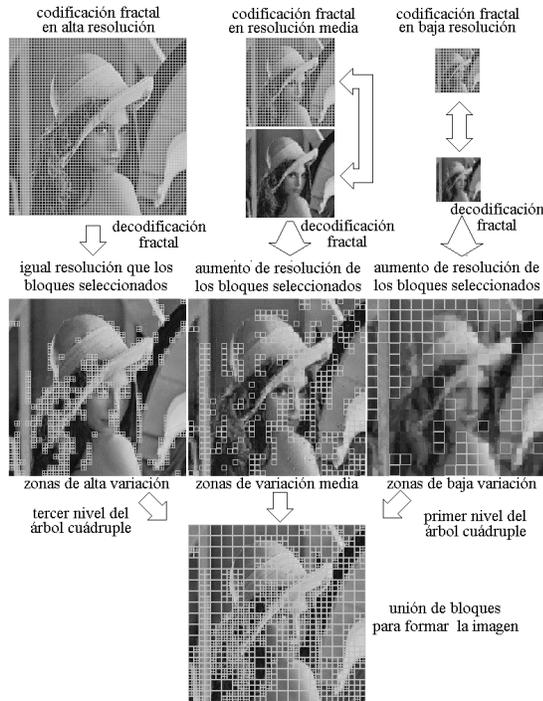


Figura 1. Procedimiento de codificación fractal usando partición en árbol cuádruple sobre múltiples niveles de resolución de imagen.

4.2. Ejemplo de codificación aplicando el método propuesto

Supóngase que la imagen I a codificar tiene una resolución de 256×256 píxeles, y que se aplica en ella un particionamiento en árbol cuádruple con tres niveles de particionamiento. La imagen I es codificada usando una partición en 4096 bloques rangos de 4×4 píxeles. Este particionamiento es el tercer nivel del árbol cuádruple. Puede usarse búsqueda de similitud de bloques rangos y dominios en una forma extensa o puede usarse una búsqueda local en ventanas de $(2L+1) \times (2L+1)$ bloques dominios. En este nivel de resolución se puede usar $L = 7$. El siguiente paso es formar la imagen Is_1 , con resolución de 128×128 píxeles. Sobre Is_1 se usa una partición con 1024 bloques rangos de 4×4 píxeles. Esta partición es el segundo nivel del árbol cuádruple. Sobre Is_1 se aplica una codificación fractal. Si se utiliza búsqueda de similitud en ventanas, en este nivel se recomienda usar $L = 7$. A continuación se forma la

imagen Is_2 , la cual tiene una resolución de 64×64 píxeles, y se hace sobre ella una partición en 256 bloques rangos de 4×4 píxeles. Este es el primer nivel del árbol cuádruple. Se hace la codificación fractal sobre Is_2 . Si se usa búsqueda local de similitud en ventanas en este nivel de partición, entonces se recomienda usar $L = 7$. La forma en la que se hace la partición es similar a la del método down-up, en la cual la codificación empieza con la partición que proporciona los bloques rangos más pequeños, aquellos requeridos para codificar las zonas de mayor variación, y la codificación termina en el nivel de partición que tiene los bloques más grandes, que son en los que se codifican las zonas de la imagen que presentan poca variación y de los cuales se obtienen mayor razón de compresión. La figura 2 se muestra la imagen de Lena en resolución de 256×256 píxeles, la cual fue codificada usando particionamiento en árbol cuádruple con tres niveles y una pirámide de tres niveles de resolución de imagen. Se usó en ella búsqueda local de similitud con $L=7$ y $s_u=0.75$ para todas las transformaciones w_u .

La tabla 1 muestra el número de bloques rangos usados sobre cada nivel de partición en la codificación de Lena por medio del árbol cuádruple sobre múltiples niveles de resolución, así como el número de operaciones de comparación entre píxeles requeridos para codificar el total de bloques rangos, en cada uno de los niveles y en la pirámide de imágenes completa. La tabla 2 muestra el número de comparaciones entre píxeles que se requiere para codificar la misma imagen de Lena usando particionamiento en árbol cuádruple sobre un solo nivel de resolución.

En la figura 4 se muestra un ejemplo del particionamiento en árbol cuádruple de tres niveles de la imagen de Lena. Las zonas marcadas en negro corresponden al primer nivel de partición y representan cuadros de 32×32 píxeles que tienen una variación en el nivel de luminosidad de 40 (la variación máxima de luminosidad en la escala de grises es de 255), las zonas marcadas en gris oscuro corresponden al segundo nivel de particionamiento y representan cuadros de 16×16 píxeles que tienen una variación en el nivel de luminosidad de 80,

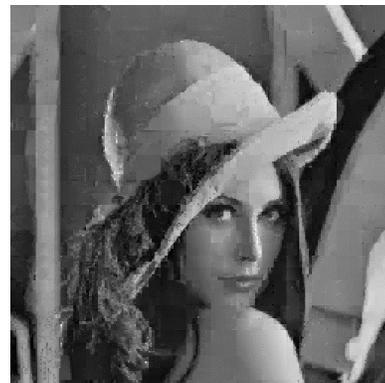


Figure 2. Lena en resolución de 256x256 píxeles. Codificada usando particionamiento en árbol cuádruple a través de tres niveles diferentes de resolución. Con respecto a Lena en la misma resolución, esta imagen tiene PSNR=28 dB, RMSE=13.4 y RC = 14.6:1.

Nivel del árbol	Rangos codificados	No. de píxeles/rango	Comp. de píxeles/rango	Total de comp. entre píxeles
1	256	256	28800	7372800
2	616	64	28800	17740800
3	1092	16	28800	31449600
Total	1964	-	-	56563200

Tabla 1. Operaciones empleadas en la codificación de Lena mediante la aplicación del particionamiento en árbol cuádruple aplicado sobre múltiples niveles de resolución. Se utilizó búsqueda local con L=7. Número de píxeles/rango=16. Comparaciones de bloques/rango = 1800.

Nivel del árbol	Rangos codificados	No. de píxeles/rango	Comp. de píxeles/rango	Total de comp. entre píxeles
1	256	256	460800	117964800
2	616	64	1843200	70963200
3	1092	16	28800	31449600
Total	1964	-	-	220377600

Tabla 2. Operaciones empleadas en la codificación de Lena mediante la aplicación del particionamiento en árbol cuádruple aplicado sobre un nivel de resolución. Se utilizó búsqueda local con L = 7. Comparaciones de bloques/rango = 1800.

y las zonas marcadas en gris claro corresponden al tercer nivel de particionamiento y representan cuadros de 8x8 píxeles que tienen una variación en el nivel de luminosidad de 255. Los bloques del primer nivel de particionamiento se codifican en una resolución de imagen de 64x64 píxeles. Los bloques del segundo nivel de particionamiento se codifican en una resolución de imagen de 128x128 píxeles, y los bloques del tercer nivel de particionamiento se codifican en una resolución de imagen de 256x256 píxeles. Esta asignación de 255 a la variación del nivel de luminosidad en los bloques del tercer nivel de partición hace que en ese nivel se codifiquen todos los bloques que no se tomaron en cuenta en los niveles anteriores del particionamiento y a la vez, no deja bloques para otros niveles mas finos de particionamiento. Por lo que el árbol cuádruple solamente tendrá tres niveles.

En la figura 5 se muestra la aproximación fractal de Lena usando el particionamiento en árbol cuádruple mencionado aplicado sobre tres niveles de resolución. Para esta aproximación fractal a la imagen de Lena se obtuvo una PSNR = 26.89 dB y RMSE = 11.53. La razón de compresión es RC = 33.5:1. La imagen de Lena se

dividió en 1978 bloques y el código fractal cuenta con igual número de transformaciones w_u . Todos estos bloques fueron procesados como bloques de 4x4 píxeles al hacer la codificación multiresolución descrita en este método.

Se obtienen mejores resultados de calidad visual y mayor PSNR si se utiliza una descomposición wavelet multi-resolución para formar las sub-imágenes I_{s1} e I_{s2} [5,6], pero es necesario llevar a cabo pre-procesamiento



Figure 4. Mapa de particionamiento en árbol cuádruple de tres niveles de la imagen de Lena. El particionamiento se efectúa de acuerdo a la variación del nivel de luminosidad.

de la imagen y aún más, las sub-bandas a , h , v y d del procesamiento wavelet deben ser procesadas sobre cada nivel de resolución, y no se desea almacenar ni procesar los coeficientes de las bandas de alta frecuencia de la descomposición wavelet. Se ha querido mantener a este método libre de preprocesamiento, para obtener un método rápido que realice la codificación fractal de imágenes logrando alta razón de compresión.

5. CONCLUSIONES

La codificación fractal usando particionamiento en árbol cuádruple a través de múltiples niveles de resolución reduce el número de operaciones de comparación entre píxeles requerido cuando se codifica cada bloque rango. Esto se logra porque los bloques rango que cubren las áreas de la imagen con poca variación son codificados en imágenes de menor resolución, y en ellas tienen menor tamaño del que tendrían si se codificaran en la imagen original. Este método de codificación fractal conserva las

ventajas del método de partición en árbol cuádruple sobre un solo nivel de resolución, como la obtención de buena calidad de imagen y grandes razones de compresión, pero reduciendo el número de operaciones de comparación, y por lo tanto, reduciendo el tiempo de codificación con respecto a ese método. Todos los bloques rangos en cada nivel de resolución de la imagen tienen igual tamaño, así que los bloques de procesamiento son del mismo tipo en cada nivel de resolución de la imagen. Esta característica hace a este



Figure 5. Lena en resolución de 512x512 píxeles. Codificada usando particionamiento en árbol cuádruple a través de tres niveles diferentes de resolución. Con respecto a Lena en la misma resolución, esta imagen tiene PSNR=26.9 dB, RMSE=11.5 y RC = 33.6:1.

método apropiado para implementarse sobre dispositivos lógicos programables, como FPGA's, y adecuado para realizar arquitecturas trasladables al diseño VLSI, ya que pueden usarse bloques de procesamiento similares y estructuras regulares para procesar la imagen, sin importar el nivel de particionamiento que se está considerando ni el nivel de resolución de la imagen correspondiente en ese nivel de particionamiento.

6. REFERENCIAS

- [1] A.E. Jacquin, "Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations", *IEEE Trans. on Image Proc.*, Vol. 1, No. 1, pp 18–30, Jan 1992.
- [2] John Kominek, "Advances in Fractal Compression for Multimedia Applications". *Department of Computer Science, University of Waterloo, Ontario, Canada*, Technical report, pp. 9,1995.
<ftp://links.uwaterloo.ca/pub/Fractals/Papers/External/>
- [3] M. Gharavi - R. De Nardo - Y. Tenda - T. Huang. "Resolution Enhancement of images using fractal coding" in *Visual Communications and Image Processing '97*, J. Biemond and E. J. Delp, Eds., San Jose, CA, USA, Feb 1997, vol. 3024 of *SPIE Proceedings*, pp. 1089 – 1100.
- [4] Harvey A. Cohen, "Fractal Image Compression Based thumbnails". *Proc. ICIP-96 IEEE International Conference on Image Processing*, Lausanne, Sept. 1996.
- [5] R. Rinaldo and G. Calvagno, "Image coding by block prediction of multiresolution subimages", *IEEE Transactions on Image Processing*, vol. 4, no. 7, pp. 909 – 920, July 1995.
- [6] G. M. Davis, "A wavelet-based analysis of fractal image compression", *IEEE Transactions on Image Processing*, vol. 7, no. 2, pp. 141 – 154, Feb. 1998.