

## ESTUDIO DE COMPRESION FRACTAL PARA IMÁGENES CON TEXTO

Jesús Yalja Montiel Pérez, Saúl de la O torres, Jesús Alfredo Martínez Nuño

Escuela Superior de Cómputo, Instituto Politécnico Nacional

Av. Juan de Dios Bátiz s/n esquina Miguel Othón de Mendizabal. Unidad Profesional Adolfo López Mateos.

Col. Lindavista C.P. 07738, México, D. F. email: yalja@ipn.mx

### RESUMEN

Se presenta un método mixto mediante lógica difusa y un algoritmo PI saturado para controlar la mezcla de aire – combustible (AFR) en un motor de combustión interna de ignición por chispa. La técnica propuesta tiene la particularidad de ser poco dependiente de los parámetros exactos del modelo matemático. Aunque por otro lado se requiere de cierto grado de sintonización por parte del usuario. La utilización de un sensor de oxígeno UEGO permite situar el punto de ajuste en un amplio rango de valores diferentes al de la combustión estequiométrica (14.66). El algoritmo fue probado satisfactoriamente en pruebas de simulación.

### 1. INTRODUCCIÓN

La imagen como elemento fundamental de información ha sido el centro de estudio en varias disciplinas. Con respecto a la computación se han desarrollado múltiples algoritmos de tratamiento y análisis de imágenes, entre estos lo que es la compresión. La representación de una imagen digital puede ser sin compresión o con compresión de datos y con sus diversas variantes de algoritmos. Los formatos más utilizados son TIFF, GIF, PNG JPEG, BMP, PBM etc. Cada uno con sus características propias. Actualmente el JPEG por sus características y razones de compresión es de los más utilizados en diversas aplicaciones y áreas de trabajo que requieran del uso de las imágenes [1]. Los algoritmos de compresión fractal de imágenes es un tema de investigación que se encuentra en proceso de desarrollo y en general es un método que hace uso de las similitudes de los elementos que componen a una imagen [2, 3]. Entre las variantes de compresión fractal de imágenes se tienen los algoritmos basados en segmentación de bloques y regiones, expansiones de auto-similitudes, etc. [2, 3, 4].

### 2. DESARROLLO

En el presente trabajo se utilizó un algoritmo de similitudes de regiones de una imagen, con una definición de las segmentaciones de regiones con

base a un algoritmo de quadtree [5]. Cada segmento se utilizó una comparación con tres transformaciones: rotaciones de  $90^\circ$  y  $180^\circ$  y escalamiento al doble del tamaño original. Cada segmento origen se guarda y se compara con los demás de la imagen, si en alguna de las transformaciones resulta similar, evaluado con un error  $\varepsilon$ . Se registra la posición del segmento y la transformación realizada.

La compresión fractal basada en Quadtree es un método para cambiar el rango de un bloque  $R_i$  al dominio de un bloque  $D_i$ , basado en la relación del error permisible  $\varepsilon$ , para elegir coeficiente de transformación razonable y reducir la búsqueda en tiempo de información; siguiendo los pasos:

- 1) Segmentar la imagen original  $I$  dentro del rango del bloque de tamaño  $2r_{max} \times 2r_{max}$ , no hay superposición como en el rango original del bloque.
- 2) Se toma el bloque de dominio  $D_i$  con 2 veces el tamaño del rango del bloque  $R_i$  de la imagen original  $I$ , haciendo un promedio de 8z transformaciones afines  $D_i^t$ , para el bloque de dominio  $D_i$ .
- 3) Calcular la raíz cuadrática media del bloque del rango  $R_i$  y cada bloque de dominio correspondiente, así como el error de emparejamiento o coincidencia. El error de emparejamiento debe satisfacer  $d < \varepsilon$ ,  $\varepsilon$  es un error preestablecido, en caso contrario pasar al paso 5.
- 4) Se la búsqueda completa a terminado, y no encuentra la condición  $d < \varepsilon$ , segmentar en bloque original  $R_i$  en cuatro bloques iguales, repetir las operaciones 2 y 4.
- 5) Registrarla información actual de la codificación fractal, continua hasta terminar la codificación.

El diagrama de estados del caso de uso compresión consta de cuatro estados, Figura 1, los cuales son:

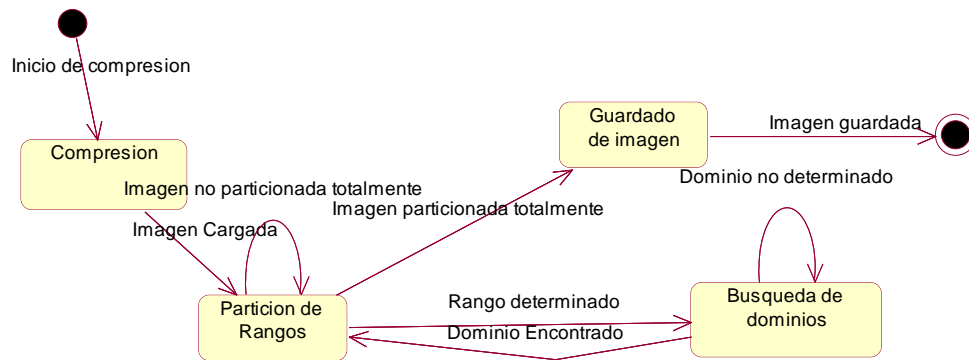


Figura 1. Diagrama de estados del caso de usos de la compresión de una imagen

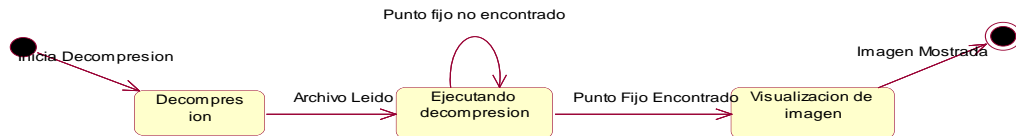


Figura 2. Diagrama de estados del caso de usos de la descompresión de una imagen

Estado Compresión. Este estado es el que inicia el proceso de compresión de imágenes, su único objetivo es el cargar la imagen y el inicio de la compresión en si.

Estado Partición de rangos. Este estado tiene como función el particionamiento de la imagen y el llamado de la función de búsqueda de dominios, cuando haya terminado de particionar la imagen procederá al estado de guardado de la imagen.

Estado búsqueda de dominios: Este estado es el que se encargará de la búsqueda de dominio y de la obtención de la información del IFS, regresa al estado de particionamiento de rangos cuando encuentra un dominio viable.

Estado de Guardado de imagen: Este estado es el que se encarga del guardado de la información al archivo

Estado de descompresión, Figura 2, se encarga de la lectura del archivo y del guardado de los coeficientes de las ecuaciones, después de haber guardado todas las ecuaciones a memoria pasara al estado de Realizando Descompresión.

Estado Realizando Descompresión. Este estado efectuará las iteraciones de las funciones, reconstruyendo de esta forma la imagen.

El algoritmo Quadtree al aplicarse con las condiciones y características descritas arriba se puede observar graficamente para la imagen de la Figura 3 a), como lo mostrado en la Figura 3 b). En las figuras simples se observa una gran cantidad de similitudes por lo que la razón de compresión se incrementa. Y bajo esta hipótesis se aplica el algoritmo para imágenes que contienen texto y graficos.

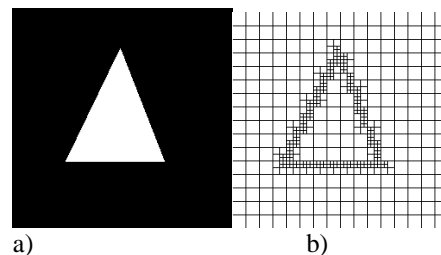


Figura 3. a) Imagen original con una forma simple; b) Diagrama de estados del caso de usos de la compresión de una imagen

Para medir la calidad de la compresión de una imagen existen diversas técnicas objetivas tales como la Relación Señal a Ruido Máxima (PSNR) y el Índice de Similitud Estructural (SSIM).

El PSNR es una métrica que mide el nivel de ruido que obtiene una señal al compararse con otra, la unidad de esta métrica es el decibel (dB).

La PSNR se define como:

$$PSNR = 10 \cdot \log_{10} \left( \frac{MAX^2}{MSE} \right) (1)$$

Donde:

MAX es el valor máximo posible de píxeles de la imagen. Cuando los píxeles se representan utilizando 8 bits por muestra, esto es 255.

MSE es el error medio cuadrático que se calcula:

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |I(i,j) - K(i,j)|^2 (2)$$

Donde: M es el número de píxeles Ancho de la Imagen y N es el número de píxeles Alto de la Imagen.

Los valores típicos de la PSNR en pérdida de imagen y compresión son entre 30 y 50, donde el obtener mayor valor, significa que se obtuvo una pérdida menor.

### 3. PRUEBAS Y RESULTADOS

Los resultados obtenidos al aplicar el algoritmo de compresión de similitudes en conjunto con el algoritmo quadtree se reportan a continuación. De manera general se tiene una imagen BMP en escala de grises con dimensiones de 1024X1024 y un tamaño de 1MB, Figura 4. En la Figura 5 se muestra esta imagen comprimida en una razón de 1:11.

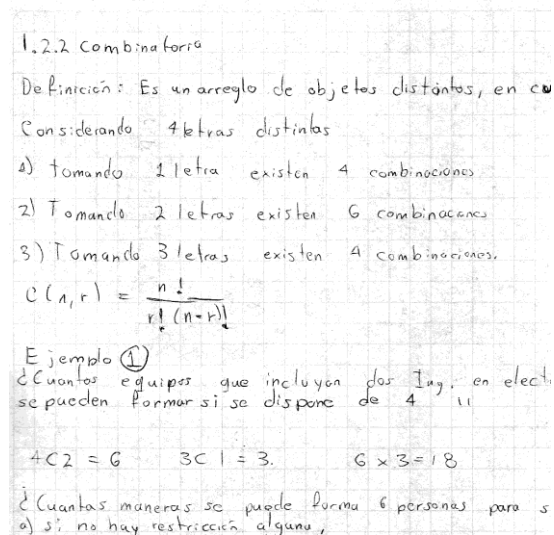


Figura 4. Imagen original

Las pruebas que se realizaron fue con un banco de imágenes de apuntes a mano con diferentes características. El proceso fue que a la imagen original se comprimió con diferentes razones de compresión, teniéndose como limitante el tamaño de la imagen y su calidad medida por el PSNR.

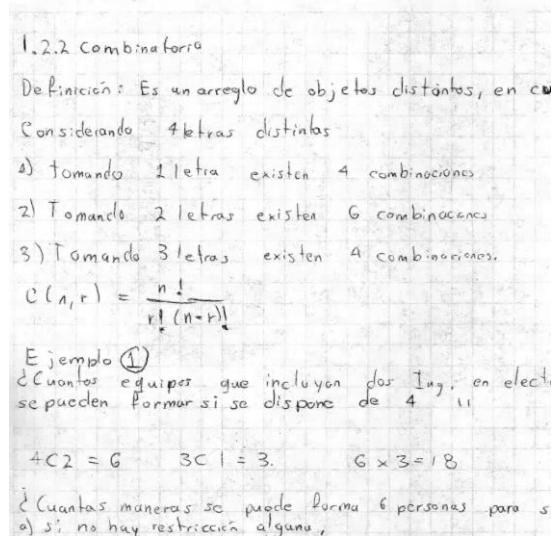


Figura 5. Imagen comprimida con algoritmo fractal y razón de compresión 1:11

La más alta razón de compresión para la imagen de la Figura 4 es de 1:50 donde se pierden detalles de la imagen y esta se muestra en la Figura 6.

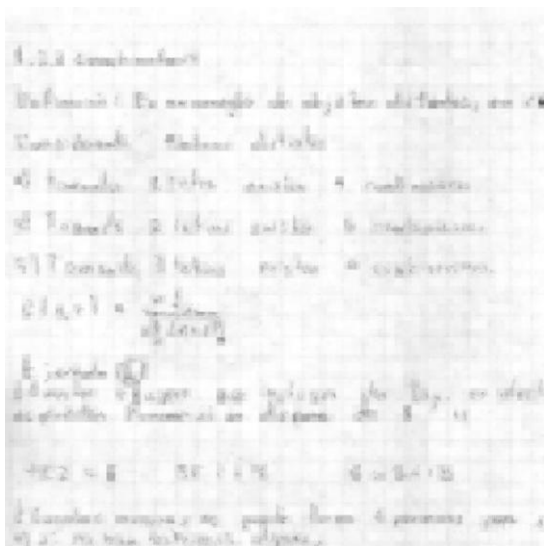


Figura 6. Imagen compresada con razón de 1:50

Se utilizaron bancos de imagen con características similares en cuestión de contenidos. Por ejemplo, apuntes con solo texto, apuntes con texto y graficos, texto y formulas matemáticas, etc., obteniéndose resultados semejantes.

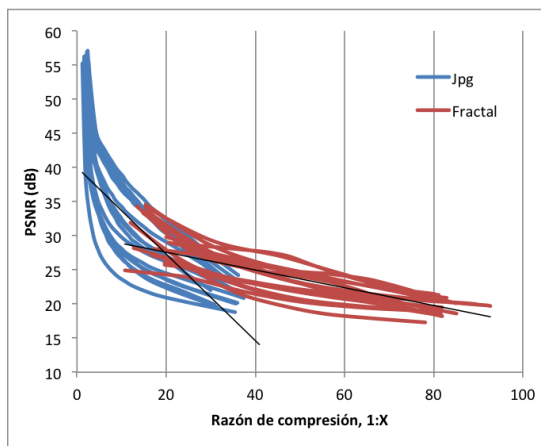


Figura 7. Gráfica de razón de compresión y calidad PSNR con imágenes con sólo texto

En la Figura 7 se muestran los resultados de aplicar el algoritmo de compresión fractal a imágenes con sólo texto. Donde en general las razones de compresión van de 1:20 a 1:80. Para realizar una compración de referencia, las mismas imágenes originales se comprimen con un formato jpg. Teniendo que con este formato de compresión

se tiene razones desde 1:3 hasta 1:38.

#### 4. CONCLUSIONES

Se señalan las siguientes ventajas al aplicar el método en imágenes donde solo se incluía texto y formulas matemáticas:

Razón de compresión mayor a otros métodos. La compresión fractal, comparada contra otros métodos de compresión tiende a tener una razón de compresión mayor. En el presente artículo se comparo principalmente con jpeg.

Descompresión rápida. Esta ventaja lo hace ideal para aplicaciones donde sólo se tenga que guardar una vez, pero se tenga que leer muchas veces.

Detalle a todo nivel. La imagen no presenta problemas de píxelado.

Tiempo de codificación. El tiempo de codificación en la compresión fractal tiende a ser muy grande, sin embargo es factible la optimización, donde de manera clara al aplicarlo únicamente a documentos con texto y formulas se mejora.

Con respecto a la razón de compresión y calidad de las imágenes la grafica de la Figura 7, muestra que para las imágenes donde se utiliza el algoritmo de jpg, la pérdida de información que repercute en la calidad de la imagen se pierde rapidamente con respecto al aumento de la razón de compresión. Cuestion que no sucede con los algoritmos de compresión fractal, donde la calidad se pierde lentamente con respecto a como aumenta la razón de compresión y por las mismas características del algoritmo, es posible continuar aplicando la compresión, aunque para un momento dado ya no es adecuada la compresión porque se pierde la información del texto.

La información de una imagen de texto es diferente a la imagen con un paisaje por ejemplo. El texto se debe de poder leer siempre. El algoritmo comprime mucho mas que el estándar jpg y puede tener multiples aplicaciones para un futuro próximo.

#### 5. BIBLIOGRAFÍA

- [1] Wilhelm Burger, Mark J. Burge; "Digital imageprocessing"; Springer; 2010.
- [2] ErjunZhao, Dan Liu; "Fractal ImageCompressionMethods: a review"; Proceedings of theThird International ConferenceonInformationTechnology and Applications; 2005.

- [3] VijayshriChaurasia, AjaySomkuwar;  
“Review of a novel technique: fractal  
imagecompression”;International  
JournalonEmerging Technologies Vol.1; No.1;  
pp. 53-56; 2010.
- [4] SachinDhawan; “A Review of  
ImageCompression and Comparison of  
itsAlgorithms”; International Journal of  
Electronics&CommunicationTechnology; Vol. 2;  
No. 1; 2011.
- [5] Hui Yu, Li Li, Dan Liu, HongyuZhai,  
Xiaoming Dong;  
“BasedonQuadtreeFratalImageCompressionImpro  
vedAlgorithmforResearch”; International  
Conferenceon E-Product E- Servic and E-  
Enterteiment (ICEEE); pp. 1-3; 2010. }