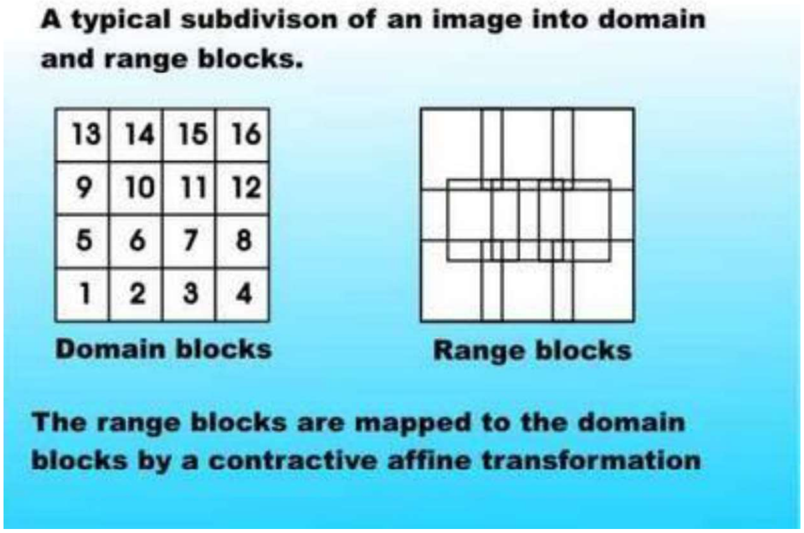
ABSTRACT

Fractal compression is a lossy compression method for digital images, based on fractals. The method is best suited for textures and natural images, relying on the fact that parts of an image often resemble other parts of the same image[3]. A fractal is a natural phenomenon or a mathematical set that exhibits a repeating pattern that displays at every scale. In this project, we are going to implement a compression scheme in which an image is first compressed with fractal compression technique using Quadtree decomposition and then hauffman compression technique is used to further compress the image without huge loss in image quality.

Fractal Compression

Given an original image, say B (here we assume B is nonempty, otherwise there is nothing to be compressed), with a resolution of m by n pixels, the image file consists of a header followed by m n cells of intensity data, one for each pixel. Given the resolution, the spatial coordinates of each pixel are implied (e.g. the first n cells represent the top most row of pixels, starting from left, etc.). The fractal image compression first partitions the original image into non-overlapping domain regions (they can be any size or shape) in our case its square[1]. Then a collection of possible range regions is defined. The range regions can overlap and need not cover the entire image, but must be larger than the domain regions, For each domain region the algorithm then searches for a suitable range region that, when applied with an appropriate affine transformation[2], very closely resembles the domain region.



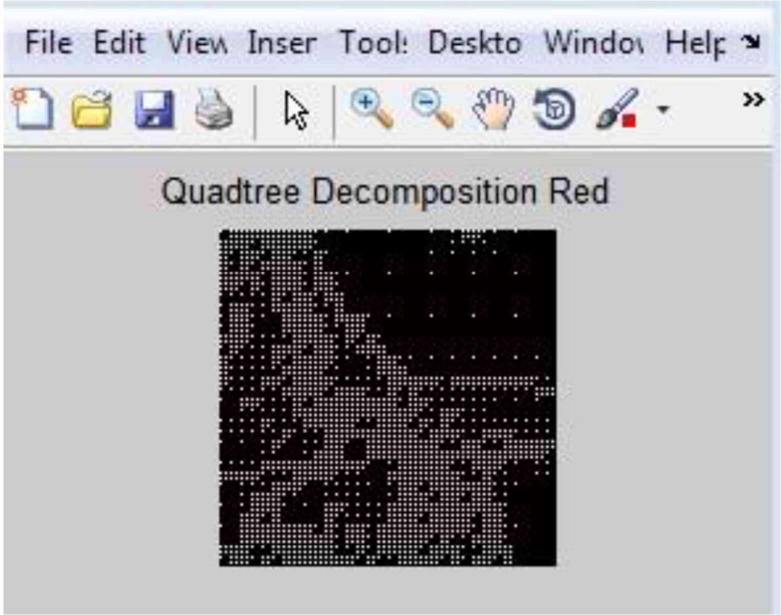
A typical subdivison of an image into domain and range blocks.

The range blocks are mapped to the domain blocks by a contractive affine transformation

For every range block the number of the appropriate domain and relevant information needed to retrieve that range are stored. Hence the compression is achieved in place of storing a range block only the parameters are stored. The decoder performs a number of iterative operations in order to reconstruct the original image.

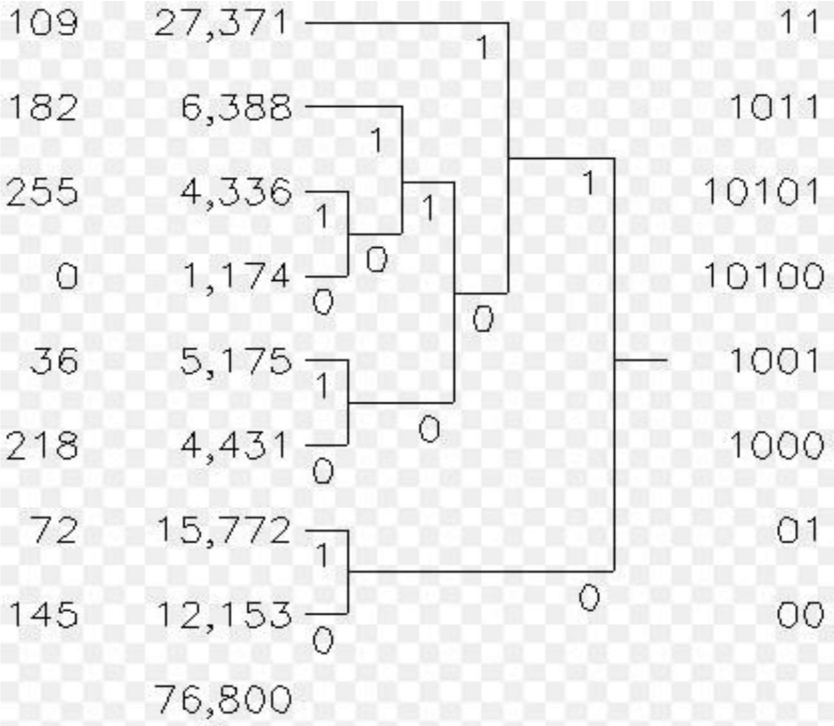
Quadtree Decomposition

The Quad tree approach[4] divides a square image into four equal sized square blocks, and then tests each block to see if meets some criterion of homogeneity. If a block meets the criterion it is not divided any further, and the test criterion is applied to those blocks. This process is repeated iteratively until each block meets the criterion of homogeneity.


File Edit View Inser Tool: Deskto Windo Help
Quadtree Decomposition Red

Hauffman Compression

Hauffman Compression requires a set of symbols that needed to be encoded, than the prior probabilities of the symbols are calculated based on their frequencies. Generation of partial trees starts with the combination of symbols having least probabilities. Partial tress as generated to show the encoding scheme.



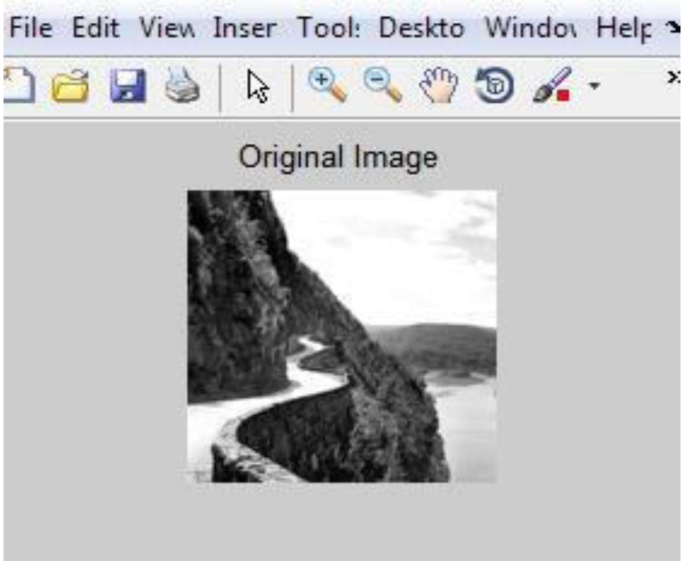
Hauffman decoding starts with the binary bits that were obtained as a result of encoding procedure. From the initial set of symbols, we can identify the encoded message.

The initial set of symbols is expected to be present in both the sides, sender and receiver.

Results

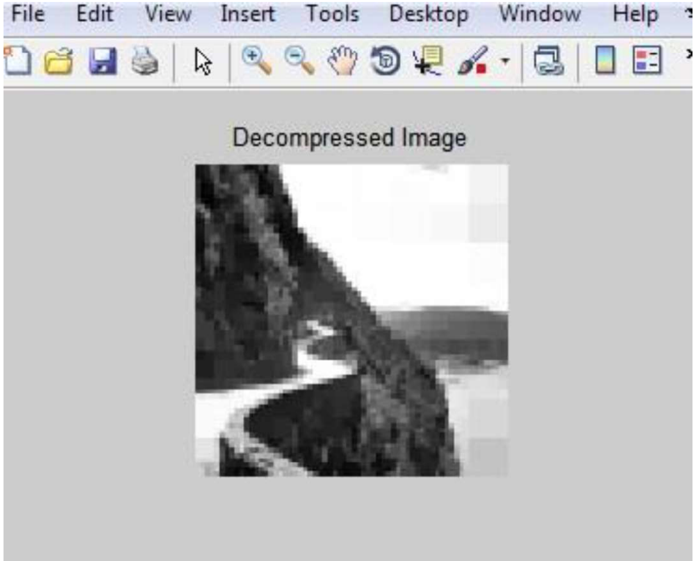
The above compression scheme is implemented in MATLAB and various metrics such as compression-ratio and SNR are calculated in order to justify the objective of the scheme.

Input:



Size: 354KB

Output:



Size: 150.17KB

Observation

From the above results we observe that using Quadtree decomposition in Fractal Compression followed by Hauffman Compression we obtain very efficient output. The size of image was reduced to almost half, although the quality of image was also reduced but it was enough to interpret the original image.

Limitations

- Slow in performance, best recommended for those applications which requires reduced size of image. It is the area of improvement we find in our scheme.

References

[1] http://www.math.psu.edu/tseng/class/Fractals.html

[2] http://www.i-programmer.info/babbages-bag/482-fractal-image-compression.html?start=3

[3] http://en.wikipedia.org/wiki/Fractal_compression

[4] http://stackoverflow.com/questions/23355462/quadtree-decomposition

[5] Khalid Sayood, Data Compression and Encryption

[6] Fisher Y, editor (1995) "Fractal image compression: theory and application", New York, SpringerVerlag.