Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 11, Number 4 (2015), pp. 2589-2599

http://www.ripublication.com

Identification of Favorable Scanning Paths for Optimal Image Compression

V.Vaithiyanathan[1], B.Karthikeyan[1,*], B.Venkatraman[2]

[1] *School of Computing, SASTRA University, Thanjavur-613401, India*
[2] *RSEG, Indira Gandhi Centre for Atomic Research, Kalpakkam-603102, India*
[*] *e-mail: karthikeyan@it.sastra.edu*

Abstract

Image compression serves the intention of dipping redundancy or trivial information contained in the image in order to make it suitable for efficient transmission. The image information can be figured out by traversing across various paths or pixel positions. Analysis of these traversal or scanning paths directs the user to the best path which helps to achieve finest compression. For this purpose, calculation of three parameters, which give an idea of how far the image is compressed, is crucial. Experimentation on several images of different sizes and textures unfolds the best traversal path. Once the best scanning traversal path is identified, it is compressed using Huffman coding technique. Generally, this method can be applied for compression of radiographic images, X-ray imaging of various human body parts and in medical imaging. Compression techniques can also be used in many areas including technical drawing and transfer of images through the internet.

Keywords Radiographic Image, Image compression, Scanning path, Correlation, Huffman coding

1. Introduction:

Images are used extensively because they can put across more information than a piece of text. The urge to store more information in images gave rise to compression techniques. These techniques aim at eliminating redundant and irrelevant data. Data redundancy can occur in multiple ways as in coding and inter pixel redundancies and compromises the quality of the resulting image. When an image is well compressed, the size of the file is reduced without affecting its quality. For instance, compression is useful in high speed imaging applications where the system bandwidth imposes restriction on the speed (Fu et al 2011). Compressed images are in great demand in different fields of application, some of which include, medical imaging, remote sensing, video networking, security industry.

Compression algorithms exist in various categories. There are lossless compression algorithms in which data is not lost and lossy compression algorithms in which some data may possibly be lost during compression. Data loss is perceived as distortion and is measured by the signal to noise (SNR) ratio. This ratio is unique to each image and hence might not be analogous form one image to another (Jackson et al 1997). Plenty of lossless compression techniques have been designed, some of which include run length encoding (Chi et al 2009) and LZW encoding (Jackson et al 1997). Likewise lossy compression algorithms include Transformation Coding like H.264 (Wong et al 2012), DCT, DCW and Fractal Coding (Jackson et al 1997)(Chi et al 2009). A few lossy compression techniques follow dimensionality reduction because high dimensions may sometimes restrict efficient processing(Amador 2007). For images of rigid motion, Park et al proposed an adaptive lossless compression using integer wavelet transform (Park et al 2004) and Li and Sayood proposed the same using predictive coding (Li &Sayood 2007). For images of non-rigid motion like aurora images, Jiaji Wu ant et al proposed an algorithm using weighted motion compensation and context based model (Wu et al 2013).

Compression of an image is feasible because every pixel in the image is highly correlated to its neighbor. All the compression algorithms deal with increase of correlation between image pixels. The LZW encoding technique adopts a sequential traversal approach and is extensively used to replace strings of characters with single codes. It also has a remarkable effect on random data and high compression competence on successive bytes of data. However, this process could result in excessive time consumption (Zhang et al 2011). Among the lossy techniques, the DCT the compression is put into action through a quantization process and variable length coding of the image blocks. The lossy DWT scheme focuses on spatial and frequency resolutions of the compressed images. It provides high spatial resolution at high frequencies and better frequency resolution at low frequencies. It also separates images into several scales and utilizes correlation among pixels at each scale (Pan et al 2010).

Lossless compression of medical images allows a file size up to a factor of 2 or 3 resulting in a compression rate of 2:1 or 3:1(Zhang et al 2011). Generally the lossless compressions techniques have limitations over the compression ratios, unlike the lossy compression algorithms, which can potentially achieve high compression ratios. Compression ratio is calculated by the equation,

$$CR = OS/CS$$

Where, CR is the compression ratio, OS is the original size of the file and CS is size after compression (Hu & Chang 2000).

Inevitably the nature of application of lossless and lossy compression techniques exhibit huge variation. While lossless techniques are preferred for text files and data records, lossy techniques favor multimedia applications. In case of still images, the performance is tested through quality metrics. These metrics are applied on various compression schemes and are determined mathematically by the peak signal to noise ratio (PSNR) or mean squared error (MSE) (Eckert & Bradley 1998). For most applications analyzing the shape of the image makes operations easier. Shape is a very prominent feature as it contains meaningful information about the contents of the image. Representation of edges in the form of contours and arcs helps to retrieve large statistical data but as the number of edges increase, large contours cannot be efficiently described. Hence image compression serves as the primary solution. Most of the redundant data lie in the high curvature points of the contours. To perform efficient compression in contours, lossless techniques paired with quantization methods result in good compressed images (Xiao et al 2001). Quantization can be done in several ways one of which includes compression blocks of the image. Images are considered as blocks of pixel and compression of identical blocks leads to quantization (Dudek et al 2007).

2. Experimental Procedure:

Prior to compression, the image is scanned to identify an increase in the correlation between its pixels. Better compression of the image can be achieved in case of a high correlation. There are various ways of traversing an image and each of these ways offer different amounts of compression. So once the best one among all the traversal techniques is identified, lossless compression can be achieved through Huffman compression. To identify the best traversal, certain parameters can be used to indicate the amount of feasible compression. Three such parameters decide the approximation of the amount of compression resulting from the various traversal paths. This paper throws light on the various ways of traversing the pixels of images and the parameters for indicating the amount of compression. Also, it reveals the effect of each kind of traversal techniques in the resulting compression.

The traversal techniques and the parameters weighing the amount of compression are briefly discussed below with appropriate instances for better understanding.

2.1 Scanning Traversals

Scanning traversals refer to the various ways of scanning the pixels of a given image. Each of the traversal techniques indicates a different correlation between the neighboring pixels. Various traversals are considered for identifying the one that might provide better correlation between pixels. Some of the traversals are Raster Horizontal, Raster Vertical (Asada et al 1997), Snake Horizontal, Snake Vertical, Z Horizontal, Z Vertical, Diagonal, Zigzag and Spiral. These methods are illustrated with a sample 4x4 pixel position matrix as in Fig.1.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

\Fig 1: Sample Pixel Position Matrix

Traversal/ Scanning Method	Path
Raster Horizontal	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Raster Vertical	1 5 9 13 2 6 10 14 3 7 11 15 4 8 12 16
Snake Horizontal	1 2 3 4 8 7 6 5 9 10 11 12 16 15 14 13
Snake Vertical	1 5 9 13 14 10 6 2 3 7 11 15 4 8 12 16
Z Horizontal	1 2 5 6 3 4 7 8 9 10 13 14 11 12 15 16
Z Vertical	1 5 2 6 9 13 10 14 3 7 4 8 11 15 12 16
Diagonal	1 5 2 9 6 3 13 10 7 4 14 11 8 15 12 16
Zigzag	1 2 5 9 6 3 4 7 10 13 14 11 8 12 15 16
Spiral	1 5 9 13 14 15 16 12 8 4 3 2 6 10 11 7

Considering the image to be compressed, its pixels are traversed with various paths. Now, the three crucial parameters are calculated for each of these traversal paths.

2.2 Parameters

The three parameters are Activity Measure (AM), Average Run Length (ARL) and Sum of Differences (SoD), assist in identifying the amount of compression achieved.

2.2.1 Activity Measure

The number of transitions in the pixel values, called activity measure is calculated for each of the traversals. The count of activity measure is given by the equation,

$$Activity = \sum_{i=1}^{n-1} \partial(S_i - S_{i-1} Activity = \sum_{i=1}^{n-1} \partial(S_i - S_{i-1})$$

Where

$$\partial(x,y) = \begin{cases} 1; x \neq y \\ 0; x = y \end{cases} \partial(x,y) = \begin{cases} 1; x \neq y \\ 0; x = y \end{cases}$$

For example, consider pixel values as 200 200 201 202 202 202 202 204. Here number of transitions=3

Which implies, Activity Measure=3

When the number of transitions is low, it indicates that it can be compressed to a greater extent.

2.2.2 Average Run Length

Run length is calculated by finding the number of occurrences and length of the sequence. Let c be the run length, n be the number of occurrences, and L be the total length of the sequence. Then average run length can be calculated as per the formula, ∑ c* (c*n/L)

For example, consider pixel values as 200 201 201 201 200 200 201 202 202 202

Run Length	Count	c* (c*n/L)
1	2	1*(1*2/10)=0.2
2	1	2*(2*1/10)=0.4
3	2	3*(3*2/10)=1.8

Here average run length=2. 4

When the average run length is high for a traversal it indicates that the image can be compressed to a greater extent.

2.2.3 Sum of Difference

The difference between i^{th} pixel value and $(i+1)^{th}$ pixel value is calculated and summation of all the differences for each of the traversals is considered.

$$sum\ of\ difference = \sum_{i=0}^{n-1} |a_i - a_{i-1}|$$

For example, consider pixel values 200 200 201 202 202 202 202 204.

Sum of difference=0+1+1+0+0+0+2=4

When the sum of differences is low for a traversal, the traversal can be compressed to a greater extent. When multiple images are taken into account, the displacements between subsequent images can sometimes be less than one pixel and hence, they may not fit into the pixel grid. In such cases, interpolation methods can be adopted to minimize erratic circumstances (De Strycker et al 2011).

2.3 Algorithm

2.3.1 General Algorithm for Image Compression:

i. Identify the amount of correlation between the pixels.
ii. Quantization
iii. Entropy coding

2.3.2 Proposed Algorithm for Image Compression:

i. Traverse the pixels of the image to be compressed in nine ways mentioned above

ii. Calculate the three parameters for each of the traversal methods
iii. Compare the values obtained using Volume Method and Rank method
iv. Identify the best traversal path among the various traversals
v. Carry out Huffman coding

This process is applicable for encoding. The reverse process supports decoding.

Huffman coding takes into consideration the statistical frequencies of the image information and assigns appropriate number of bits for each pixel in the image according to its frequency. It is simple and effective and needs low computational cost, thus making it a popularly used encoding scheme (Hu & Chang 2000).

2.4 Comparing Traversals

Now that the three parameters are calculated for each of the traversals, they are compared to find the best traversal among them. Two methods Volume method and Rank method are adopted for comparison. To obtain optimal compression, the space and time complexities of the scanning paths are given primary importance. Reduction in search space and faster processing of paths yields good results and helps to retain image quality (Duh et al 2005).

2.4.1 Volume Method

A cuboid is constructed with three parameters as the length, breadth and height of the cuboid. One cuboid is constructed for each of the traversal, thus, generating nine cuboids on the whole. The volume of each of the cuboid is calculated. To The value obtained for the traversals are compared. The lowest value is identified and the traversal which corresponds to this value gives the optimized way of traversing the image. Low values indicate more correlation between the pixels, and hence better compression can be achieved.

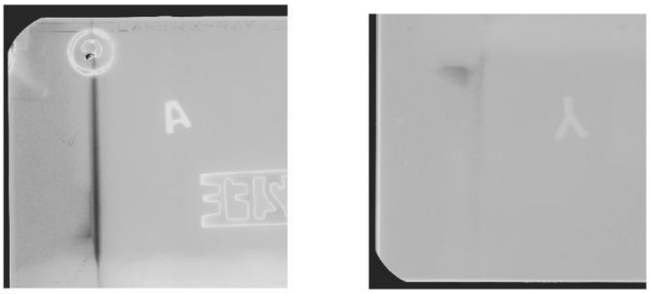
2.4.2 Rank Method

In this method individual ranks are assigned for the three parameters (AM, ARL and SoD) which is used for identifying better scan path for image compression. For the parameters AM and SoD, the scan paths are ranked such a way that the highest rank is given for the lowest value. For the parameter ARL, the highest rank is given to the scan path whose value is high. Sum of the ranks for each scan path is calculated. The method with the lowest sum of rank is the optimal method for traversing the image.

3. Results

The image to be compressed is traversed in all the nine ways and stored in files (Fig.2). Each of these files is compressed with Huffman coding. The file which is compressed to a greater extent is identified. This file uses the traversal path corresponding to the outcome of evaluation methods like Volume method and Rank method. This implies that by scanning the traversal in a particular method one can achieve better compression.Many samples of images of various sizes and texture were taken for experimentation.

(a) (b)



(c) (d)

Fig 2: Uncompressed Images: (a), (b), (c) and (d) represent images img11, img33, img44, and img88 respectively.

Table 1: Performance comparison of expected and actual compression values obtained for various scanning paths

File	Predicted Scanning Path		Optimal Scanning Path After Compression	
	Volume	**Rank**	**Lowest (Bytes)**	**Second Lowest (Bytes)**
Img11	Snake Vertical	Snake Vertical	Raster Vertical (7312085)	Snake Vertical (7315509)
Img33	Snake Horizontal	Snake Vertical	Raster Vertical (7428638)	Snake Vertical (7434195)
Img44	Snake Horizontal	Snake Vertical	Raster Vertical (7384961)	Snake Vertical (7391333)
Img55	Snake Vertical	Snake Vertical	Raster Vertical (7336857)	Snake Vertical (7339670)
Img88	Snake Horizontal	Snake Vertical	Raster Vertical (7148633)	Snake Vertical (7150799)

Img1_2	Zigzag	Snake Vertical	Raster Vertical (130570)	Diagonal (130727)
Img2_1	Snake Vertical	Snake Vertical	Raster Vertical (133032)	Snake Vertical (133291)

Table 2: Comparison of scanning paths for img11

Traversal	Activity	Average Run Length	Sum of Difference	Optimization –Volume	Optimization-Rank
Raster Horizontal	909572	72676	2.51E+07	7	22
Raster Vertical	909331	73451.875	2.37E+07	3	10
Snake Horizontal	909099	72830.625	2.36E+07	2	12
Snake Vertical	**909047**	**73512.5**	**2.31E+07**	**0**	**4**
Z Horizontal	909854	72719	2.47E+07	6	12
Z Vertical	909629	73103.25	2.40E+07	4	12
Diagonal	909982	72958.125	2.55E+07	8	12
Zigzag	909739	72818.875	2.45E+07	5	12

The statistical survey of the scanning methods highlights the most optimal one and the best method can therefore be adopted for compression. A low activity count, high average rum length and a low sum of difference indicates high quality compression.

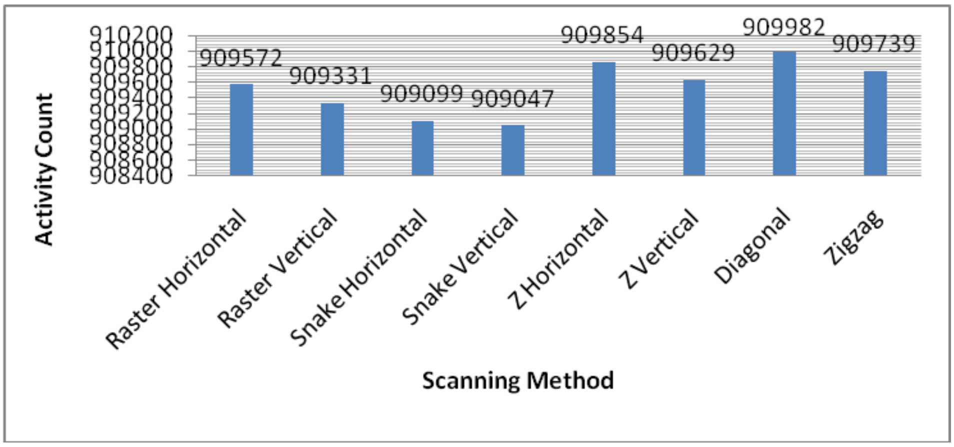


Fig 3: Graph indicating activity counts for various scanning methods.

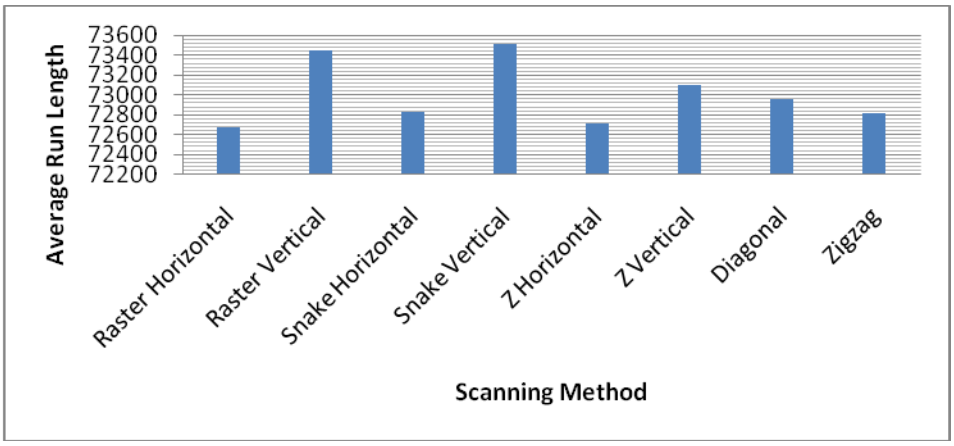


Fig 4: Graph indicating average run length for various scanning methods.

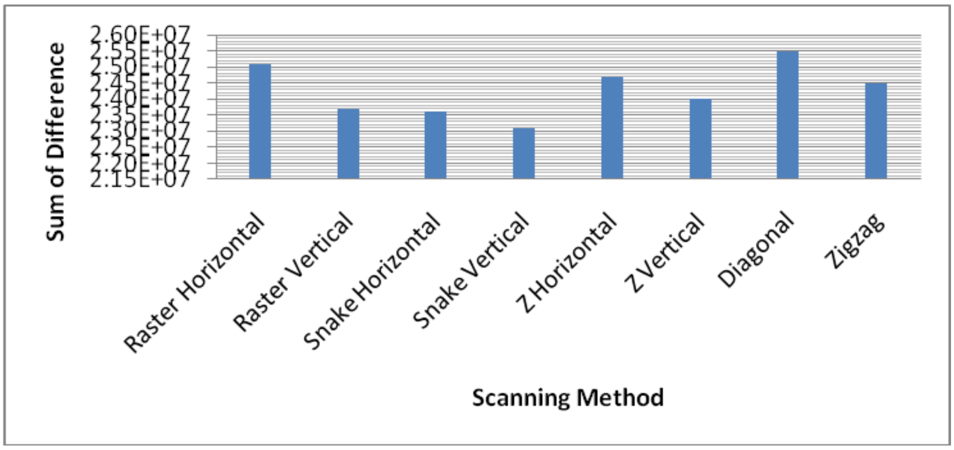


Fig 5: Graph indicating sum of difference for various scanning methods.

From all the different samples experimented, the optimization parameters indicate that snake vertical is the best traversal. This proves that when an image is scanned in the snake vertical method, there is a higher correlation among the pixels of the image and thus better compression can be achieved (Fig.3-Fig.5). As raster vertical and snake vertical methods have very low variation in the number of pixels, they produce best results when compressed using Huffman coding (Table 1-Table 2). According to the optimization rank method, snake vertical scanning produces best result, thus proving its efficiency as the best of all scanning paths.

4. Conclusion

Analyzing various scanning techniques provides a head start for the task of incorporating better compression in images. The focus here lies on the path taken to scan images which is vital for the quality of compression. Each of the scanning techniques taken up during the experimentation phase produces different degrees of compression. Correlation between the pixels is the key feature to measure the degree of compression. The experimental results prove that high correlation among the neighboring pixels paves way for better compression. It is also evidently seen that an image can be compressed to a greater degree when snake vertical method is adopted for scanning.

References

[1] Amador J J 2007 Random projection and orthonormality for lossy image compression, *Image Vision Comput*. 25(5): 754–766

[2] Asada K, Akita J, Watabe R 1997 A Tree Structure of Automata for Selective Image Scanning and Its Implementation. *Computers and Electrical Engineering*. 23(6): 453-461

[3] Chi Y M, Etienne-Cummings R, Cauwenberghs G 2009 Focal-Plane Change Triggered Video Compression for Low-Power Vision Sensor Systems. *PLoS One* 4(7) art.no e6384

[4] De Strycker M, Lava P, Van Paepegem W, Schueremans L, Debruyne D 2011 Measuring Welding Deformations With The Digital Image Correlation Technique. *Weld J*. 90(6): 107-112

[5] Dudek G, Borys P, Grzywna Z J 2007 Lossy dictionary-based image compression method, *Image Vision Comput*. 25(6): 883–889

[6] Duh D J, Jeng J H, Chen S Y 2005 DCT based simple classification scheme for fractal image compression. *Image VisionComput*. 23(13): 1115–1121

[7] Eckert M P, Bradley A P 1998 Perceptual quality metrics applied to still image compression. *Signal Process*. 70(3): 177-200

[8] Fu B, Pitter M C, Russell N A 2011 A Reconfigurable Real-Time Compressive-Sampling Camera for Biological Applications. *PLoS One* 6(10) art.no e26306

[9] Hu Y C, Chang C C 2000A new lossless compression scheme based on Huffman coding scheme for image compression. *Signal Process-Image*. 16(4): 367-372

[10] Jackson D J, Mahmoud W, Stapleton W A, Gaughan P T 1997 Faster fractal image compression using quadtreerecomposition. *Image Vision Comput* 15(10): 759–767

[11] Li Y, Sayood K 2007 Lossless video compression using adaptive prediction. *IEEE T Image Process*. 16(4): 997-1007

[12] Pan H, Jin L Z, Yuan X H, Xia S Y, Xia L Z 2010 Context-based embedded image compression using binary wavelet transform. *Image VisionComput*. 28(6): 991–1002

[13] Park S G, Delp E J, Yu H 2004 Adaptive lossless video compression using an integer wavelet transform. *Proc. IEEE*: 2251–2254

[14] Wong A, Shafiee M J, Azimifar Z 2012 Statistical Conditional Sampling for Variable-Resolution Video Compression. *PLos One* 7(10) art.no e45002

[15] Wu J, Liang C, Han J, Hu Z, Huang D, Hu H, Fang Y, Jiao L 2013 A two-stage lossless compression algorithm for aurora image using weighted motion compensation and context-based model. *Opt Commun*. 290: 19-27

[16] Xiao Y, JiaZou J, Yan H 2001 An adaptive split-and-merge method for binary image contour data compression. *Pattern RecognLett*. 22: 299-307

[17] Zhang F, Li Z, Wen M C L, Jia X, Chen C 2011 Implementation and Optimization of LZW Compression Algorithm Based on Bridge Vibration Data. *Procedia Engineering* 15: 1570–1574