Volume 6, Issue 6, June 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Performance Analysis of Image Compression using BTC with Region Based Segmentation Algorithm

Nitin Prasad
M.Tech. Scholor, EC Dept., NIIST, Bhopal, India

Brahmi Shrman
Assistant Professor, EC Dept., NIIST, Bhopal, India

Lalit Jain
Assistant Professor, EC Dept., NIIST, Bhopal, India

Abstract— In the present era of communication system, the requirement of image storage and transmission for image processing are increasing exponentially. This is why; the need for better compression technology is in extremely demands. In this paper, a gray image compression method using region based segmentation is proposed. This method having the advantages of BTC and quantization both. The BTC algorithm with quantization has some controlling parameters through which we can control the quality and compression of the image. The performance of the proposed method has been evaluated in terms of PSNR, MSE & SSIM. The result of the proposed work is evaluated by comparing the performance with that of the existing methods.

Keywords— BTC, Compression, quantization, algorithm, image retrieval, multimedia.

I. INTRODUCTION

20th century is information driven society where information's are generated, manipulated and transmitted over the distances by means of some media. The amount of this digital information is huge and a need of large amount of storage space is required. The transmission of such information through limited bandwidth channel is very time consuming. With the use of internet the data transmitted are huge in amount which consumes more bandwidth. Now a day's these data are increasing rapidly and most of the data are image. The usefulness of digital images in communicating information is well appreciated. But the cost of storage and transmission of data is very expensive. Hence the need of image compression technique is much more in demand.

An image compression method represents an image in a more compact way. The compression techniques decrease the effective volume required to store an image or data hence allowing more data to be stored in the available storage device. It also increases the bandwidth of the transmission medium and higher volume of data can be transmitted over the same transmission channel. Thus, image compression refers to a process in which the amount of data used to represent an image is reduced. The quality of the reconstructed image must satisfy the requirement of certain application, and the complexity of computation should also be affordable for the application [1,2].

The required quality of the reconstructed image is application dependent. For example, in medical science and some other scientific measurement, the reconstructed image has to be an exact replica of the original image. Only lossless compression methods are used in these application fields. However, in applications like motion pictures, video phony etc. some amount of loss of information is acceptable. It is termed as lossy compression.

II. COMPRESSION TECHNIQUE

Different types of redundancies in image information are used to help to compress them. There are two categories in which compression technique exists: one is transform domain methods and second is spatial domain methods. Compression techniques may also be classified as lossy/lossless.

II.I Transform Domain Methods

The performance of transform domain methods is superior to that of spatial domain methods in terms of compression ratio and quality of the restored image. Recent used compression standards like, JPEG (DCT based) [3, 5] and JPEG2000 (wavelet based) [4,6] are transform based methods. In transform domain methods images are first transformed from spatial domain to frequency domain then quantization coefficients are carried out by the entropy coding. In traditional transform coding, discrete cosine transform or Fourier transform are used. The limitation of these transformations is the block size which needs to handle both smooth as well as active area. The active area is localized in spatial domain whereas the smooth area information in frequency domain. So, it is very difficult to obtain a compromise between these using the traditional transformation.

II.II Spatial Domain Method

The decoding part of Spatial domain compression methods are computationally less exhaustive compared to the transform domain methods. Block truncation coding (BTC) [8] and 1-D/2-D run length coding [9] are most popular spatial domain techniques. Other methods include block matching coding [10], quad tree coding [11], context-based coding [12], predictive coding (e.g., Differential pulse code modulation (DPCM)), bit plane coding and fractal based coding [22]. In bit plane method an image is considered as series of binary images and each binary image is compressed with binary compression method. In block match coding method there are two buffers, the search buffer and the look-ahead buffer. Each block is first matched with all the blocks in the search buffer and then the best matched block is selected for the conclusion.

II.III Lossless/ Lossy Compression

Image compression is decisive steps in multimedia applications, and their visual quality has to be maintained up to a certain satisfactory level. But in some applications like, clinical analysis, scientific research & measurement and legal issues the decompressed image has to be an exact replica of the original image. Hence the selection of the method depends on the requirement; a compression method may be lossy or lossless. All the compression methods are either lossy or lossless based on scheme related to coding. After an extent, there is a limit in lossless compression beyond which the compression may not be achieved [14,15].

In lossy image compression, a direct relation between the amount of compression and the distortion occurred exists. Most often a lossy image compression provides better compression than lossless compression. For example Lempel-Ziv-Welch coding, Contour coding, Huffman coding, Run length coding, and arithmetic coding. In Lossy coding methods predictive coding, Quantization, transform based coding (JPEG and JPEG2000), block truncation coding, and fractal based coding occurs.

II.IV Objective quality Measurement

The goal of objective quality assessment research is to suggest quality metrics that can quantify the quality of image automatically. Objective quality assessment techniques in literature are mainly error-based methods. It uses pixel based difference metrics for example MSE, RMSE, mean absolute error (MAE), signal to noise ratio, and peak signal to noise ratio (PSNR). Let $f(x,y)_{M\times N}$ and $g(x,y)_{M\times N}$ be the original and reconstructed images, respectively. The MSE between f and g is defined as:

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\{f(x,y) - g(x,y)\}^2 \quad (1)$$

For 8-bit gray scale image PSNR is defined as

$$PSNR = 10log_{10}\frac{255^2}{MSE}dB \qquad (2)$$

SSIM: The Structural Similarity (SSIM) index is used to measure the similarity between two images. It can be treat as a quality measure of a images with respect to the compared image. It is a full reference metric; it means, the measurement or probability of image quality is based on an original or error free image as reference [16,17].

It is designed to improve on traditional methods PSNR and MSE, which is often seen as inconsistent with human visual perception. It is evaluated on various windows of an image. The measure between two windows x and y of common size N×N is:

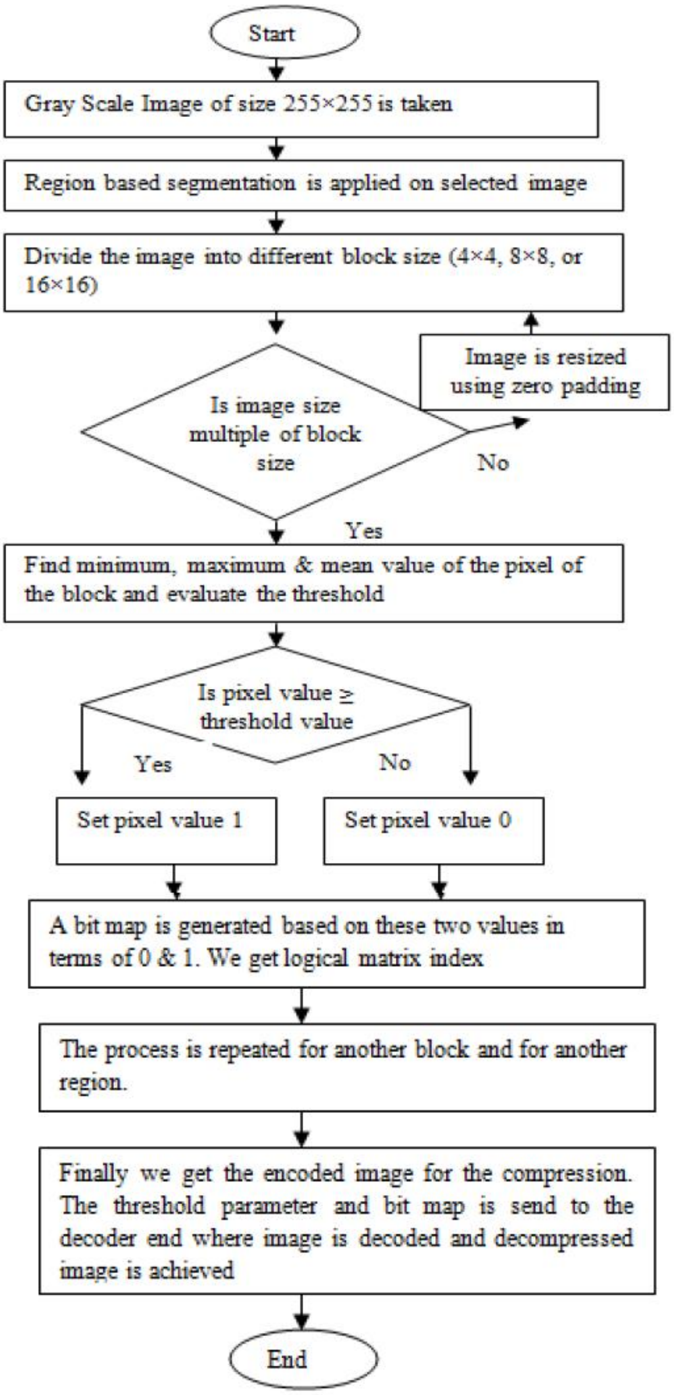
$$\text{SSIM (x,y)} = \frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x{}^2+\mu_y{}^2+c_1)(\sigma_x{}^2+\sigma_y{}^2+c_2)} \quad (3)$$

Where,
μ_x is the average of x; μ_y is the average of y; ${\sigma_x}^2$ is the variance of x; ${\sigma_y}^2$ is the variance of y; σ_{xy} is the covariance ; $c1 = (k_1L)^2$, $c2 = (k_2L)^2$ L the dynamic range of the pixel-values (typically this is $2^{\#\, bits\, per\, pixel} - 1$); $k_1 = 0.01$ and $k_2 = 0.03$ by default.

III. PROPOSED WORK

In this proposed work lossy image compression has been implemented. For the implementation of Lossy image compression Block Truncation coding with region based segmentation has been applied. In the first stage image of size 256×256 has been segmented on the basis of region. Then a block of size n×n (where n= 4, 8 or 16) has been chosen. We obtain the minimum, maximum and mean value of pixel of that block. On the basis of threshold based on above parameter we evaluate a bit map for the particular block. The process is applied on every block of the image. Thus obtained bit pattern or logic matrix with parameters is send to the decoder end. On the decoder end the image is decompressed on the basis of transmitted bit map information.

$$Threshold = \frac{max.\ pixel\ value + min.\ pixel\ value + mean}{3}. \quad (4)$$



IV. SIMULATION RESULT

This paper has been implemented on the basis of above discussed algorithm using MATLAB simulator. First of all the work has been simulated on 4×4 block size then 8×8 and 16×16 block size for different input and reference images. Region based segmentation provides better quality in terms of MSE, PSNR and SSIM, which has been evaluated on the simulator. Finally the result has been compared with different existing method in the literature.

Simulation of Block truncation coding with vector Quantization for 4×4 block size

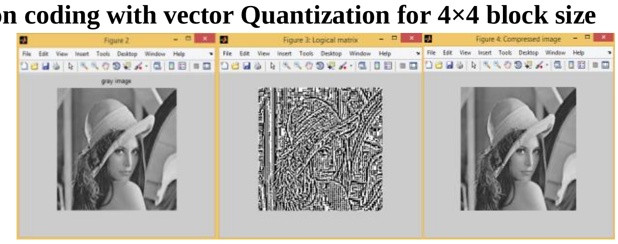


Figure 1: Lena Input image (a) Gray scale image (b) logical matrix image (c) compressed image

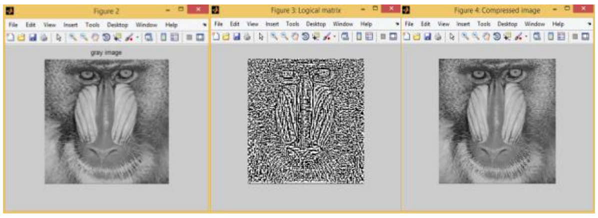


Figure 2: Baboon Image (a) Gray scale image (b) logical matrix image (c) compressed image

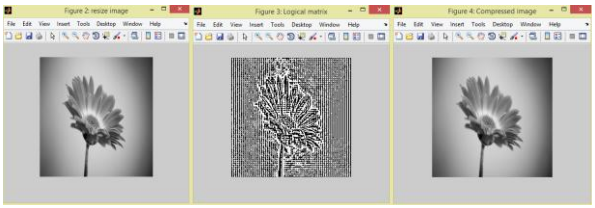


Figure 3: Flower Image (a) Gray scale image (b) logical matrix image (c) compressed image

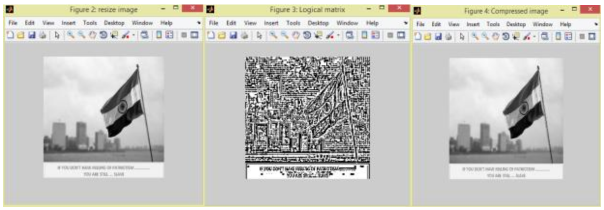


Figure 4: Flower Input image (a) Gray scale image (b) logical matrix image (c) compressed image

Figure 5: Books Input image (a) Gray scale image (b) logical matrix image (c) compressed image

RESULT ANALYSIS FOR 4*4 Block Size

Table 1: Performance in terms of MSE, PSNR & SSIM for different image

Image	MSE	PSNR	SSIM
Lena Image	16.0643	81.8468	0.9965
Baboon Image	32.2887	72.9427	0.9952
Flower Image	6.5197	90.4435	0.9989
Flag Image	8.8026	89.0748	0.9966
Books Image	13.7426	79.4591	0.9956

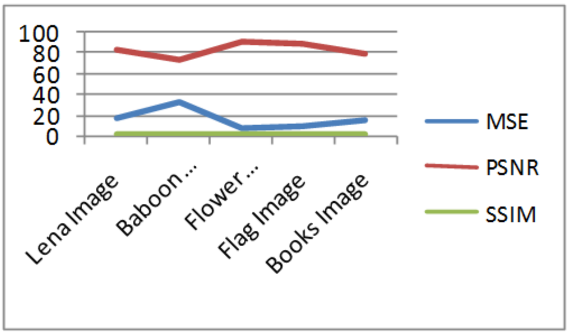


Figure 6: Graph for MSE, PSNR & SSIM for different image of 4 ×4 block size

Simulation of Block truncation coding with vector Quantization for 8×8 block size

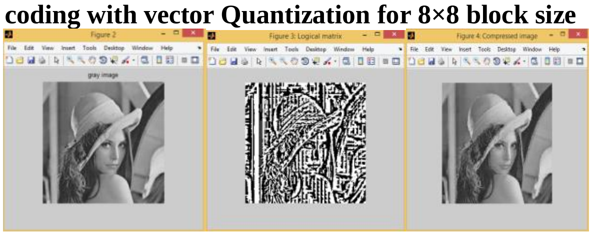


Figure 7: Lena Input image (a) Gray scale image (b) logical matrix image (c) compressed image

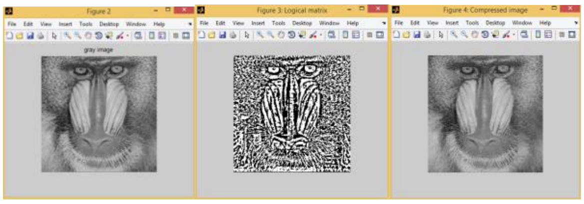


Figure 8: Baboon Input image (a) Gray scale image (b) logical matrix image (c) compressed image

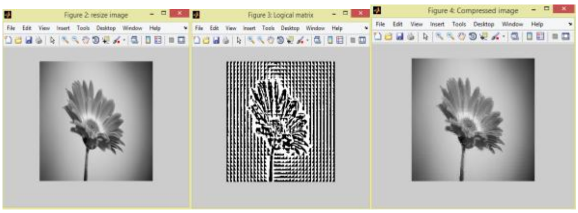


Figure 9: Flower Input image (a) Gray scale image (b) logical matrix image (c) compressed image

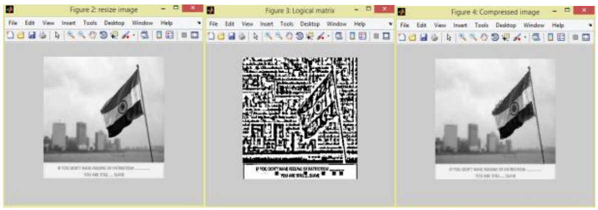


Figure 10: Flag Input image (a) Gray scale image (b) logical matrix image (c) compressed image

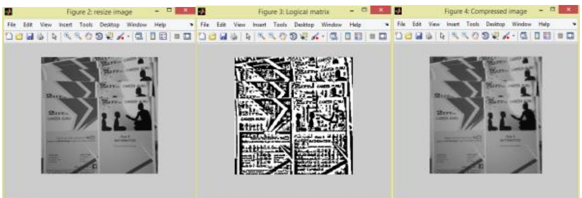


Figure 11: Books Input image (a) Gray scale image (b) logical matrix image (c) compressed image

RESULT ANALYSIS FOR 8×8 Block Size

Table 2: Performance in terms of MSE, PSNR & SSIM for different image

Image	MSE	PSNR	SSIM
Lena Image	26.5617	76.8181	0.9933
Baboon Image	44.0471	69.8373	0.9925
Flower Image	11.8553	84.4639	0.9974
Flag Image	14.4715	84.1035	0.9938
Books Image	20.0289	75.6923	0.9923

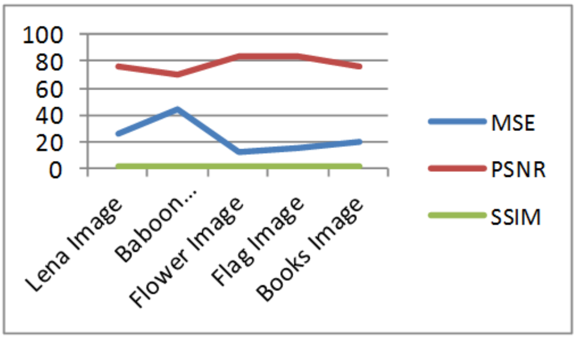


Figure 12: Graph for MSE, PSNR & SSIM for different image of 8 ×8 block size

Simulation of Block truncation coding with vector Quantization for 16×16 block size

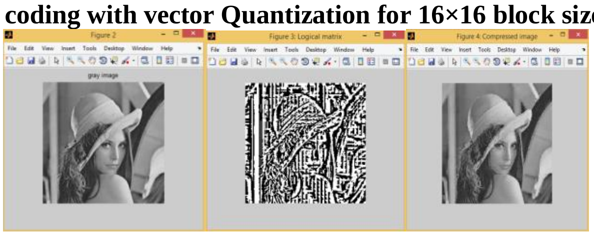


Figure 13: Lena Input image (a) Gray scale image (b) logical matrix image (c) compressed image

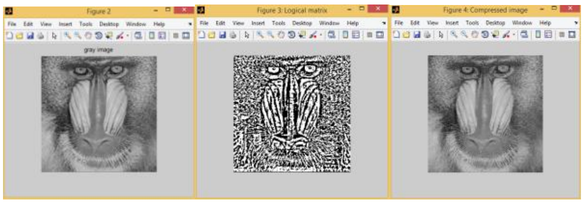


Figure 14: Baboon Input image (a) Gray scale image (b) logical matrix image (c) compressed image

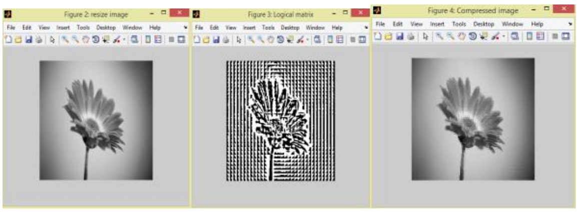


Figure 15: Flower Input image (a) Gray scale image (b) logical matrix image (c) compressed image

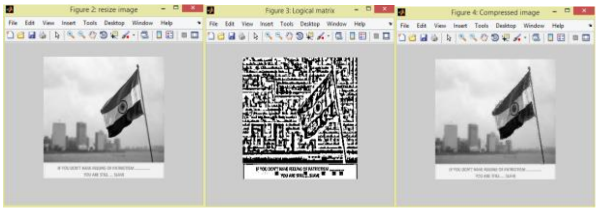


Figure 16: Flag Input image (a) Gray scale image (b) logical matrix image (c) compressed image

Figure 17: Books Input image (a) Gray scale image (b) logical matrix image (c) compressed image

RESULT ANALYSIS FOR 16×16 Block Size

Table 2: Performance in terms of MSE, PSNR & SSIM for different image

Image	MSE	PSNR	SSIM
Lena Image	38.4451	73.1205	0.9897
Baboon Image	53.3275	67.9254	0.9901
Flower Image	19.4488	79.5139	0.9901
Flag Image	20.4149	80.6626	0.9901
Books Image	27.7290	72.4393	0.9885

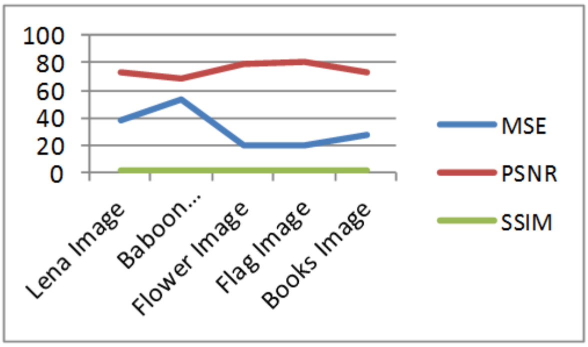


Figure 18: Graph for MSE, PSNR & SSIM for different image of 8 ×8 block size

V. CONCLUSION

In this paper lossy image compression using Block Truncation Coding with region based segmentation has been performed using MATLAB simulator. Different block size taken for the implementation like 4×4, 8×8 & 16×16 and the image size is 256 ×256. The performance analysis has been carried out using MSE, PSNR & SSIM for different real and reference images. Result shows better improvement over the existing methodology.

REFERENCES

[1] E. J. Delp and O. R. Mitchell, “Image compression using block truncation coding,” IEEE Trans. Commun., vol. 27, no. 9, pp. 1335–1342, Sep. 1979.

[2] D. Salomon, Data Compression: The Complete Reference. Springer-Verlag, New York, 2000.

[3] Y. Q. Shi and H. Sun, Image and video compression for multimedia engineering: fundamentals, algorithms, and standards. CRC Press, USA, 1999.

[4] D. Huffman, A method for the construction of minimum redundancy codes. Proceeding of the IRE, 40:1098 – 1101, 1952.

[5] I. H. Witten, M. N. Radford, and J. G. Cleary, Arithmetic coding for data compression. Communications of ACM, 60:520 – 540, 1987.

[6] W. Pratt, Digital Image Processing, John Wiley and Sons, New York, 1978.

[7] K. R. Rao and P. C. Yip Eds, the Transform and Data Compression Handbook. CRC Press, Boca Raton, FL, 2001.

[8] S. Dunn. Digital color image processing using MATLAB. http://davis.wpi.edu/~matt/courses/color.

[9] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Prentice Hall , 2002.

[10] V. Ostromoukhov, “A simple and efficient error-diffusion algorithm,” in Proc. 28th Annu. Conf. Comput. Graph., 2001, pp. 567–572.

[11] P. Li and J. P. Allebach, “Tone-dependent error diffusion,” IEEE Trans. Image Process., vol. 13, no. 2, pp. 201–215, Feb. 2004.

[12] Jing-Ming Guo and Yun-Fu Liu, “Improved Block Truncation Coding Using Optimized Dot Diffusion”, IEEE Transactions on image processing, vol. 23, no. 3, March 2014

[13] Jayamol Mathews, Madhu S. Nair, Liza Jo , “Modified BTC Algorithm for Gray Scale Images using max-min Quantizer”, IEEE, 2013

[14] Seddeq E. Ghrare , Ahmed R. Khobaiz, “Digital Image Compression using Block Truncation Coding and Walsh Hadamard Transform Hybrid Technique”, IEEE, 2014.

[15] T. Ebrahimi, C. A. Christopoulos, and D. T. Lee, Special issue introduction, jpeg2000. Signal Processing: Image Communication, 17:1–144, 2002.

[16] J. M. Shapiro, Embedded image coding using zero trees of wavelet coefficients. IEEE Transactions on Signal Processing, 41:3445 – 3462, 1993.

[17] A. Said and W. A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. IEEE Transactions on Circuits and System for Video Techniques, 6:243– 250, 1996.

[18] D. Marpe and H. L. Cycon, Very low bit rate video coding using wavelet-based techniques, IEEE Transactions on Circuits and Systems for Video Technology, 9:85 – 94, 1999.

[19] D. Gleich, P. Planinsic, and Z. Cucej, Low bitrate video coding using wavelet transform, In Proc. of the EURASIP conference on Video/Image Processing and Multimedia Communications, pages 369 – 374, 2003.

[20] R. Leonardi and J. R. Ohm, Wavelet video coding - an overview (iso/iec jtc1/sc29/wg11 w7824). 2006.

[21] R. B. Arps, Binary image compression. In Image Transmission Techniques, pages 219 – 276. Academic Press, W. K. Pratt (Ed.), New York, 1979.

[22] H. Samet, Application of spatial data structures: Computer Graphics, Image Processing and GIS. Addison Wesley, Reading , MA, 1990.

