Volume 1, No. 2, July-August 2010
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
CASE STUDY AND REPORT
Available Online at www.ijarcs.info

Effect of Data Filling Techniques for Compound Document Images

D. Maheswari *
PhD Research Scholar
Department of Computer Science
Avinashilingam Deemed University for Women
Coimbatore-43
mahileni@gmail.com

Dr. V. Radha
Associate Professor
Department of Computer Science
Avinashilingam Deemed University for Women
Coimbatore-43

Abstract: Information explosion along with the tremendous growth in software and hardware technology are moving industries, educational institutions, organizations towards 'paperless' environment. This change has a direct impact on the amount of information digitized, thus increasing the need for efficient storage mechanisms. Digitized information mostly consists of compound images, which are a combination of different data types like text, graphics, line art, photographs, etc. Layer based compression is one of the frequently used solution to compress these kind of images. This paper discuss about the effect of data filling techniques such as Prefilling with Fast inpainting, Successive projections, Non-Linear, Prefilling with Dc-Value for compound document images. Experimental results were conducted to analyze the performance of the four types of data filling techniques.

***Keywords*:** Compound Image Compression, Data Filling, JPEG Artifacts, Layer Based Compression, Segmentation, Token Compression.

I. INTRODUCTION

Advances in digital image technology coupled with breakthroughs in the price and performance of digital hardware and software have resulted in an ever increasing demand for efficient storage and transmission practices. This demand is continuing to increase as more and more companies go 'digital' and have started to move towards 'paperless' environment. Digital image compression is the method of image data rate reduction to save storage space and reduce transmission rate requirements.

Digital image compression is considered to be a mature branch in image processing area, where several researches have already focused on developing algorithms that produce good compression rates with adequate image quality. These techniques focus on the characteristics of image data to produce good compression results.

Efficient text compressors like Run Length Encoding, arithmetic coders, natural image compressors like JPEG, TIF, and graphic image compressors like GIF have been used frequently in the past. However, the fast development of the Internet and widespread rich media applications produce not only text, natural and graphical images but also images that combine these three together. These images are termed as 'Compound images' and examples include computer generated images, text images, scanned images, document images, etc. (Figure 1).

Compressing compound images with a single algorithm that simultaneously meets the requirements for text, image and graphics has been elusive and thus requires new algorithms that can competently reduce the file size without degrading the quality. According to [1], each of the image data type exhibit different characteristics and best compression performance of such images can be achieved when different compression techniques are used for each of these image data types.

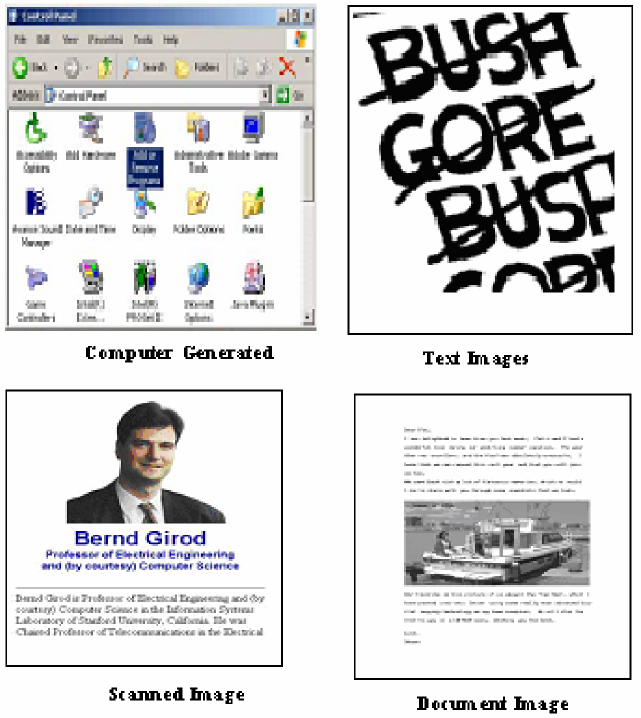

Figure1 Examples for compound images

To perform such a task, a process called segmentation is used where regions of similar data types are grouped together. After successful segmentation existing techniques that best suits each data type can be used to achieve best compression results. The performance of a compound image compressor heavily depends on the segmentation result. Careful usage of segmentation techniques not only lowers the bit rates but also lowers the level of distortion. Most of the segmentation algorithms proposed in the literature belong to three categories, namely object-based, block-based and layer-based segmentation [2].

Object based segmentation deals with segmenting the image into several objects based on Regions of Interest

(ROI) and each object is compressed using different techniques. Block based segmentation is similar to object based segmentation and classifies images into blocks of several types, which can be coded with different encoding schemes. In layer based segmentation, the original image is divided into several rectangular layers with different objects and mask planes. A mask plane has pixel information, which should be included in the final composite page. Each of these layers can then be compressed using different algorithms.

Researches in all these areas are abundant ([3], [4], [5]) and some commercial compressors like DjVu [6], DigiPaper [7], JPEG 2000 [8], also exists. The main objective is to analyze the various datafilling techniques on layer based compound image compression. For this purpose, the MRC (Mixed Raster Model) as proposed initially by [9] and [10] is used. This model uses JPEG encoder for compression and suffers from two disadvantages.

The first is the halo effects, which are caused by the layer based segmentation and the second is the artifacts produced by the JPEG compressor. This study solves these problems by modifying the segmentation process, applying four datafilling techniques to solve the halo effect and introduce a post processor to take care of the artifacts.

The paper is organized as follows: Section 1 have a brief introduction to the topic under discussion. Section 2 explains the segmentation process, preprocessing techniques, postprocessing and compression techniques. Section 3 discuss about the experimental results. Section 4 concludes the paper with future research direction.

II. MRC SEGMENTATION

The segmentation process used in the proposed system is based upon the method of [11]. This method is enhanced in the following manner. The segmentation process of [11] segments the layers into foreground, background and a mask layer. In the enhanced scheme, the segmented layers consist of one color foreground layer, one color background layer, two color layers and a residual layer (have more than 2 colours). The process is pictorially represented in Figure 2 and is explained below:

A. Segmentation

The segmentation process first divides the image into 8 x 8 blocks with non-overlapping pixels categorized into four classes namely, one color foreground class, one color background class, two color classes and residual class. Each class can then be coded using coder that best suits them. During compression, the class labels are compressed along with the layer data. In an 8 x 8 block, if all pixels belong to the foreground, then that block is categorized as foreground block. Similarly, if all pixels belong to the background, then it is categorized as background block. As these two blocks have uniform color, they can be compressed using arithmetic coder. If the pixels have a combination of foreground and background, then they are termed as two color class. The two colored blocks consists of text and graphic regions and need to be coded using spatial resolution. All the pixels, which are unclassified into any of these three classes, are termed as residual classes. The residual classes normally will have overlapping regions, which is a combination of text overlapped with image. The MRC model is applied to the two-color classes, which divides the class into text layer, graphics layer and a mask layer. The foreground is composed of the multilevel text and the background is composed of graphics/pictures.

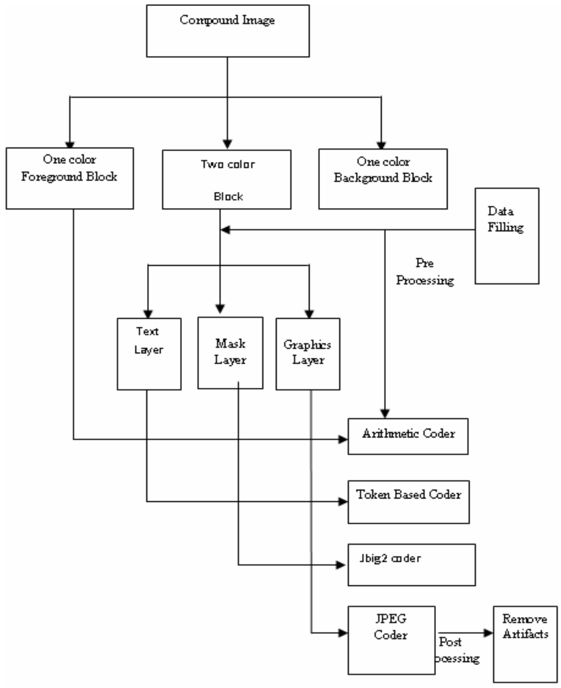

Figure 2. Proposed Compressor Model

B. Encoder

As mentioned earlier the foreground and background blocks contain single color and can be efficiently encoded using arithmetic encoder. The mask layer is encoded using a JBIG2 compressor and JPEG compression is applied to the graphics layer. A token based compressor is used for the text layer and is explained below. The steps involved in the proposed method are discussed here. The first step performs a 'Connect pixels' process where pixels connecting, that is touching, each other are grouped together. For this process a region growth method is used. The second step scans the connected components for similarity matching. In this step, when two tokens are given as input, it initially checks its height and width. If both are same, then the colour range of each pixel is examined. If the difference is smaller than a given threshold (10% used in experiments) then the tokens are said to match. As and when a new unmatching token is encountered it is stored in the token dictionary. The compression stage just uses the short code assigned to each token in the dictionary along with the position and color information. During compression, for each token constructed, a similarity search is performed in the dictionary and if found, the triple code is send to the compressor. Else the token is considered as new and is stored in a dictionary as a token along with a short code before the triple code is generated and transferred to the coder.

C. Pre –Datafilling

The main problem of layer based technique is the halo effect. Datafilling methods can be used to solve this problem. Various types of datafilling methods are

1.No prefilling
2.Prefilling with DC value
3.Suboptimal Non-linear projections

4.Prefilling with fast inpainting
5.Prefilling using Standard successive Projections etc.

This paper experiments four datafilling techniques and compared with jpeg, DjVu. The first method taken for testing is the Prefilling using fast inpainting method as proposed by Oliveiraet et al [12]. The second datafilling technique is the prefilling using standard successive projections as proposed by Chen et al [13]. The third method experimented in this paper is the suboptimal non-linear projections scheme as proposed by George Pavlidis et al [14].The fourth method discussed here is prefilling with DC value as proposed by .S. Shirani [15]. The datafilling algorithm is repeated for each block and in the algorithm, all the used pixels are termed as "U" and all unused pixels are termed as "X". Initially, the block of pixels is traversed and is filled with either U or X. If the block has all Us, the block is left as it is. If the block has all Xs, then the pixels are replaced by the average of previous blocks. It is has mixed Us and Xs, then the average of neighbours of the previous U is calculated and is used to replace the X value. This process is continued till there is no X in the block.

D. Post processing

Another problem faced with JPEG compression is the artifacts produced. Artifacts removal is solved by using a post processing method as proposed by Oztan *et* al. [16].

E. Compression

The proposed MRC model is then wrapped into a hybrid file format, which supports markup language similar to XML format to embed the information into a single file structure.

III. EXPERIMENTAL RESULTS

The performance of the system was analyzed with several executions with the test images. The main objective of the tests was to evaluate the performance of the datafilling techniques on compound image compression for gray scale and color images. The system was compared with compression ratio, time and PSNR values. Moreover the visual quality of the output images are also compared and presented. All the experiments were conducted using a Pentium IV dual processor with 512 MB RAM.

A. Test Images

In order to verify the performance of the datafilling techniques experimental evaluation is performed using a series of compound document images. The color images used during quality assessment experiment are shown in Figure 3.

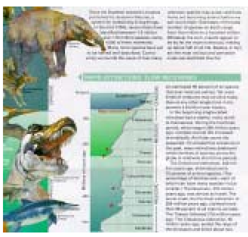

image1

image2

image3

image4

image5

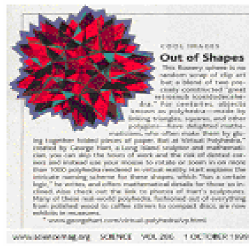

image6

Figure 3. Test Images

For experimentation 6 images were selected. All the images were of the size 256x 256 pixels.

B. Performance Metrics

The system was evaluated using various aspects like Compression Ratio, Compression and Decompression Time and Peak Signal to Noise Ratio (PSNR). The compression ratio can be measured as the ratio of the number of bits required to represent the image before compression to the number of bits required to represent the same image after compression and is given by the formula,

CR = size of input image / size of output image (1)

From the above equation, it is obvious that as the compression ratio increases the compression technique employed is more effective. Compression and decompression time are the basic measurements used to evaluate an image compression system. Compression and decompression time denotes the time taken for the algorithm to perform the encoding and decoding algorithm respectively.

PSNR is often used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed or reconstructed image. To compute the PSNR, the block first calculates the mean-squared error using the following equation:

$$\text{MSE} = \frac{\sum_{M,N}[I_1(m,n) - I_2(m,n)]^2}{M * N} \quad (2)$$

In the previous equation, *M* and *N* are the number of rows and columns in the input images, respectively. Then the block computes the PSNR for gray scale images using the following equation:

$$\text{PSNR} = 10 \log_{10}\left[\frac{R^2}{\text{MSE}}\right] \quad (3)$$

For color images with three RGB values per pixel, the definition of PSNR is the same except the MSE, which will be the sum over all squared value differences divided by image size and by three. Typical values for the PSNR in lossy image and video compression are between 30 and 50 dB, where higher is better. The PSNR for color images with color components, R, G and B is given as below:

$$\mathrm{PSNR} = 10\log_{10}\left[\frac{255^2}{\frac{\mathrm{MSE(R)}+\mathrm{MSE(G)}+\mathrm{MSE(B)}}{3}}\right]$$

In the previous equation, *R* (=255) is the maximum fluctuation in the input image data type.

C. Results

The results for the selected parameters are presented in this section. The system was compared with two standard compound image compressors, namely, JPEG 2000 and DjVu. The numerical results obtained are presented and discussed in this section.

a) Compression ratio: The consequence of compression on the six test images are presented in Table 1.

Table I: Compression ratio

Image	*JPEG*	*DjVu*	*Inpaint*	*Non-Linear*	*Succive Projec-tions*	*DC Value*
Image1	39.38	40.25	31.52	44.22	36.23	44.89
Image2	38.77	42.13	32.12	42.91	37.89	38.33
Image3	38.56	45.55	39.26	43.90	40.28	46.25
Image4	39.46	43.91	38.77	44.21	41.98	40.02
Image5	42.12	38.65	35.93	41.66	46.23	42.94
Image6	44.26	39.21	34.26	42.32	44.55	45.25

From the results portrayed, it can be seen that the Pre-filling with DC value produces good compression ratio for most of the test images. This can be graphically represented in figure 4.

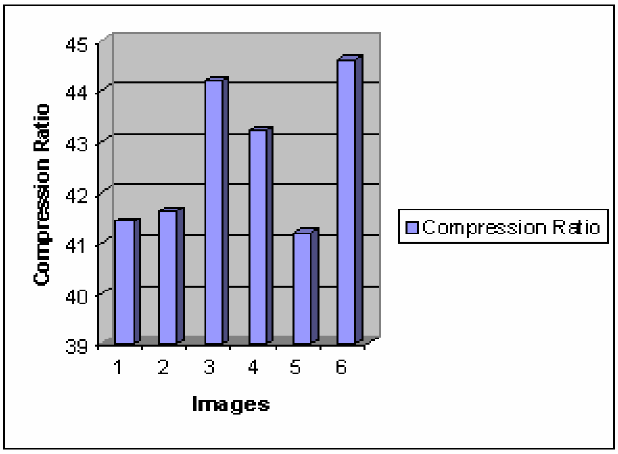

Figure 4: Average Compression Radio

a)Compression and Decompression Time: Table 2 shows the time taken by compression and decompression algo-rithms to encode and decode the images. The total compres-sion time is calculated as the sum of compression time and decompression time. Figure 5 shows the average of the total compression time for each method.

Table II: Compression and decompression time

Method	*Time*	*Image1*	*Image2*	*Image3*	*Image4*	*Image5*	*Image6*
JPEG	CT	0.87	0.86	0.54	0.78	1.23	1.32
	DT	1.04	1.08	0.11	0.18	0.44	0.51
DjVu	CT	0.77	0.78	0.89	0.97	1.10	1.12
	DT	0.91	0.83	0.46	0.42	0.25	0.32
Inpaint	CT	0.96	0.79	0.68	0.74	1.21	1.23
	DT	1.32	0.85	0.45	0.43	0.57	0.44
Non Linear Projec-tions	CT	0.85	0.89	0.82	0.89	1.17	1.10
	DT	0.71	1.01	0.43	0.62	0.88	0.23
Suc-cessive Pro-ject-ions	CT	0.70	0.82	0.65	1.14	1.10	1.27
	DT	0.71	0.89	0.44	0.46	0.97	0.49
DC value	CT	0.69	0.72	0.74	0.77	0.99	1.02
	DT	0.91	0.70	0.43	0.51	0.23	0.32

b)CT – Compression Time; DT – Decompression Time: From the data in Table 2, it is evident that the compression time taken by the Prefilling with DC value is lesser than the other three methods.

From Figure 5, the average total time, (the average of compression time + decompression time), revealed that us-age of the prefilling with DC value is best suited for com-pound document images. The results are diagrammatically portrayed in Figure 5.

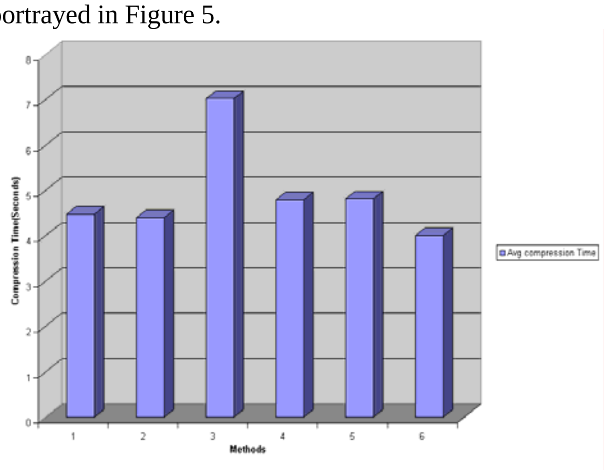

Figure 5: Average Total Compression Time

c) Peak Signal to Noise Ratio: As mentioned previously PSNR is an error metrics that is used to judge the quality of the decompressed image with that of the original image. Table 3 compares the PSNR values obtained by the four compression algorithms.

Table III: Peak signal to noise ratio

Method	*Image1*	*Image2*	*Image3*	*Image4*	*Image5*	*Image6*
JPEG	41.87	42.02	39.02	37.67	47.25	43.21
DjVu	42.36	41.83	40.02	38.12	47.68	43.86
Inpaint	41.32	41.18	39.23	38.09	37.89	38.33
Successive projections	43.26	44.98	44.23	45.87	46.77	45.92
Non linear	45.02	46.15	45.26	46.01	46.77	46.99
Dc Value	42.36	41.83	41.15	42.80	47.68	43.89

The high PSNR obtained during experimentation clearly indicates that the prefilling with DC value reduces the lousiness introduced by compression. According to the report of [17], a PSNR value in the range 30-40 indicates that the resultant image is a very good match to the original image.

VI. CONCLUSION

This paper depicts a layer based compound image compression technique. The image was initially divided into four layers, namely, one color foreground, one color background, text and picture layers. A mask layer was used to combine the MRC segmented text and picture layers. In this four datafilling algorithms are compared with the compression ratio and total compression time. The algorithms are evaluated using PSNR value. According to the experimental results the performance of prefilling with Dc value is better than all other methods.

V. REFERENCES

[1] Said, A. and Drukarev, A. (1999) Simplified segmentation for compound image compression, Proceeding of ICIP, Pp. 229-233.

[2] Ding, W., Lu, Y. and Wu, F. (2007) Enable efficient compound image compression in H.264/AVC intra coding, IEEE International Conference on Image Processing, Vol. 2, Pp. II - 337 - II – 340.

[3] Dr.Radha, V., Aparna R., and Maheswari, D. (2010) Performance evaluation of H.264/AVC intra compound image compression system, International Journal of Computer Applications, Foundation of Computer Science, Vol. 1, No.10, Pp. 48–54.

[4] Xi, Q., Wu, X. and Zhang, S. (2009) Compound Image Compression with Multi-step Text Extraction Method, Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Pp.1270-1273.

[5] Zaghetto, A. and de Queiroz, R.L. (2008) Iterative pre- and post-processing for MRC layers of scanned documents, 15th IEEE International Conference on Image processing, ICIP 2008, Pp. 1009-1012.

[6] Bottou, L., Haffner, P., Howard, P., Simard, P., Bengio, Y. and LeCun, Y. (1998) High quality document image compression using DjVu, Journal of Electronic Imaging, Vol. 7, No.3, Pp. 410-425.

[7] Huttenlocher, D., Felzenszwalb, P. and Rucklidge, W. (1999) DigiPaper: A versatile color document image representation, Proc. ICIP, Vol. I, Pp. 219-223.

[8] Sharpe, L.H. and Buckley, R. (2000) JPEG 2000 .jpm file format: a layered imaging architecture for document imaging and basic animation on the web, Proceedings of SPIE, Vol. 4115, Pp. 464-475.

[9] de Queiroz, R.L., Buckley, R. and Xu, M. (1999) Mixed raster content (MRC) model for compound image compression, Proceedings of Visual Communications and Image Processing '99, Kiyoharu Aizawa; Robert L. Stevenson; Ya-Qin Zhang, Editors,Vol. 3653, Pp. 1106-1117.

[10] Mukherjee, D., Memon, N. and Said, A. (2001) JPEG-matched MRC compression of compound documents, Proc. IEEE ICIP 2001, Thessalonica, Greece, Pp. 434-437.

[11] Cheng, H. and Bouman, C.A. (2001) Document Compression Using Rate-Distortion Optimized Segmentation, Journal of Electronic Imaging, Vol. 10, No. 2, Pp. 460-474.

[12] M. Oliveira, B. Bowen, R. McKenna, Y. Chang, Fast digital image inpainting, Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001), pp. 261–266, Marbella, Spain, September 3–5, 2001.

[13] H. Chen, M. Civanlar, B. Haskell, A block transform coder for arbitraryshaped image segments, in proceedings on the IEEE ICIP 1994, vol. 1, 1994, pp. 85–89.

[14] Pavlidisa, G. and Chamzasb, C. (2005) Compressing the background layer in compound images, using JPEG and data filling, Signal Processing: Image Communication, Elsevier, Vol. 20, Pp. 487–502.

[15] Shirani, F. Kossentini, R. Ward, Reconstruction of baseline JPEG coded images in error prone environments,IEEE Trans. Image Process. 9 (July2000) 1292–1299.

[16] Oztan, B., Malik, A., Fan, Z. and Eschbach, R. (2007) Removal of artifacts from JPEG compressed document images, Color Imaging XII: Processing, Hardcopy, and Applications, edited by Reiner Eschbach, Gabriel G. Marcu, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6493, Pp. 649306-1 – 649306-9.

[17] Schneier, M. and Abdel-Mottaleb, M. (1996) Exploiting the JPEG compression scheme for image retrieval. IEEE Trans. Pattern Anal. Mach. Intell., Vol.18, No. 8, Pp. 849–853.