VOL. 10, NO. 5, MARCH 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

ANALYSIS OF THE EFFECTIVENESS IN IMAGE COMPRESSION FOR CLOUD STORAGE FOR VARIOUS IMAGE FORMATS

Dasaradha Ramaiah K.[1] and T. Venugopal[2]
[1]IT Department, BVRIT, Hyderabad, India
[2]CSE Department, JNTUH, Hyderabad, India

ABSTRACT
Digital image compression technology is of special interest for the fast transmission and real-time processing of digital image. However image compression is a trend of research grown for a long time and there are multiple sophisticated approaches which significantly improve the compression rate and down grade computation time, a basic comparison with the aspect of storage on cloud environment is required. This work analyzes about the background of image compression, including when image compression is needed, categories of techniques and their properties. However compression uses many algorithms that store an exact representation or an approximation of the original image in a smaller number of bytes that can be expanded back to its uncompressed form with a corresponding decompression algorithm. This work also analyzes the performance of multiple image formats for multiple compression algorithms over multiple cloud storage [15].

Keywords: JPEG, DCT based embedded image coding, wavelet transform based SPIHT coder, block full JPEG 2000, cloud storage comparator, Microsoft azure, Google apps engine, Amazon Web service, cloud storage cost comparison.

1. INTRODUCTION

The digital image is most popular way of representing the information over internet because of its effectiveness of presenting information and the continuous efforts to improve the compression algorithms [1] for low cost storage over cloud infrastructure. The requirement for high resolution information for large amount of data storage cannot be ignored. The digital image contains significant amount of duplicate, redundant and complex information in high density, hence the compression of the image data cannot be neglected [2]. A set of great work has been conducted in the area of image compression; however a comparative study needs to be conducted to evaluate the performance of most popular image compression algorithms over different cloud storage platforms. The different two categories for image compression is majorly divided based on the information can be recovered with or without loose. Each category is consisting of multiple algorithms, in which most popular algorithms will be compared and tested in this work. The rest of the work is organized such as we list the major popular algorithms in both categories in Section II, in section III, we introduce multiple cloud storage environments, we introduce the framework for comparison in section IV and we conclude the work with future scope described in Section V.

2. LOSSY AND LOSSLESS COMPRESSION TECHNIQUES

The popular compression techniques based on the information loose encountered during the decompression process are categorized in two different categories as Lossy and Lossless compression techniques [1, 2].

2.1 Lossless compression techniques

We consider few most popular lossless compression techniques such as BMP, PNG and TIFF.

- **BMP**

The bitmapped graphics or the BMP format is used by the Microsoft Windows graphics subsystem or GDI as a simple graphic information representation format internally. As the popularity of the platform increases, thus increases the use of this file format. The BMP format does not include any compression technique, thus it remains lossless [2].

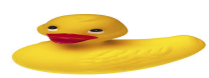

Figure-1. Size 6.27KB.

Figure-2. Size 9.47KB.

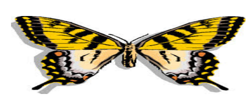

Figure-3. Size 18.5KB.

For the figures 1, 2 and 3 the Size during raw information storage is 6.27, 9.47 &18.5 and Size during information retrieval is 6.27, 9.47 and 18.5.

- **PNG**

The Portable Network Graphics or the PNG is a bitmap format, which uses a lossless data compress. The compression algorithm is called DEFLATE compression algorithm, which is a combination of LZ77 algorithm and Huffman Coding algorithm [2].
The LZ77 algorithm is defined as

$$\lceil \log_2(S) \rceil + \lceil \log_2(W) \rceil + \lceil \log_2(A) \rceil \quad (1)$$

Where,

S represents the size of the search buffer; W represents the size of the entire window and A represent the size of the alphabet

The compression algorithm results in 0.996 compression ratio.

Figure-4. Size 6.27KB.

Figure-5. Size 9.47KB.

Figure-6. Size 18.5KB.

For the Figures 4, 5 and 6 the Size during raw information storage is 6.27, 9.47 and 18.5 and Size during information retrieval is 6.20, 9.40 and 18.4.

- **TIFF**

The Tagged Image File format or TIFF is majorly intended to store image with dense information like photograph and line art. The major reason for popularity is to be used for high color depth in image manipulation tools or image processing tools for optical character recognition. The compression technique used for TIFF is LZW for majority of the cases [1, 2].
The algorithm is discussed below:

$$AvgL = L_1 * P(1) + L_2 * P(2) + L_3 P(3) .... + L_i * P(i) \quad (2)$$

Which summarize in

$$AvgL = \sum_{i=1}^{n} L_i * P(i) \quad (3)$$

Where
P = probability the input symbols
L1 = leaf which count the lower probability of the symbol
L2 = second lower probability symbol

The compression algorithm results in 0.6 compression ratio.

Figure-7. Size 7.13KB.

Figure-8. Size 8.99 KB.

Figure-9. Size 53.1KB.

For the Figures 7, 8 and 9 the Size during raw information storage is 7.13, 8.99 and 53.1and Size during information retrieval is 4.27, 5.40 and 31.5. Hence forth the list the most popular lossless compression techniques for image data is tested.

2.2 Lossy compression techniques

We consider few most popular lossy compression techniques such as JPEG and JPEG 2000.

- **JPEG**

The Joint Photographic Expert Group or JPEG is most popular compression technique for images to be transmitted over internet. The root cause of the popularity is the compression ratio can be adjusted during compression. Hence the loss of information can be controlled [3, 16].
The algorithm is discussed below:
Color space conversion and Down sampling

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299000 & 0.587000 & 0.114000 \\ -0.168736 & -0.331264 & 0.500002 \\ 0.500000 & -0.418688 & -0.081312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \quad (4)$$

The inverse transformation from YC_bC_r to RGB is

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.40210 \\ 1.0 & -0.34414 & -0.71414 \\ 1.0 & 1.77180 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ C_b - 128 \\ C_r - 128 \end{bmatrix} \quad (5)$$

$$F(u,v) = \frac{1}{4} C(u) C(v) \sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) \cos\left[\frac{\pi(2x+1)u}{16}\right] \cos\left[\frac{\pi(2y+1)v}{16}\right]$$
$$\text{for } u = 0,\ldots,7 \text{ and } v = 0,\ldots,7$$
$$\text{where } C(k) = \begin{cases} 1/\sqrt{2} & \text{for } k = 0 \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

$$\omega_{x,y}(u,v) = \frac{C(u)C(v)}{4} \cos\left[\frac{\pi(2x+1)u}{16}\right] \cos\left[\frac{\pi(2y+1)v}{16}\right] \quad (7)$$

The F (u, v) is called the DCT coefficient, and the DCT basis is

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

$$f(x,y)=\sum_{u=0}^{7}\sum_{v=0}^{7}F(u,v)\varpi_{x,y}(u,v) \quad \text{for } x=0,\ldots,7 \text{ and } y=0,\ldots,7 \tag{8}$$

Each DCT coefficient F (u, v) is divided by the corresponding quantizer step-size parameter Q (u, v) in the quantization matrix and rounded to the nearest integer as:

$$F_q(u,v)=Round\left(\frac{F(u,v)}{Q(u,v)}\right) \tag{9}$$

The compression algorithm results in 0.3 compression ratio; however this can be adjusted to improve resolution

Figure-10. Size 7.13KB.

Figure-11. Size 8.99 KB.

Figure-12. Size 53.1KB.

For the Figures 10, 11 and 12 the Size during raw information storage is 7.13, 8.99 and 53.1 and Size during information retrieval is 2.1, 2.6 and 15.93.

- **JPEG 2000**

The Joint Photographic Expert Group 2000 or JPEG 2000 is focused on wavelet based images processing and compression method. JPEG 2000 is popular as this algorithm provides more compression than the standard JPEG compression [3, 4].

The algorithm is discussed as follows [17]:

The Wavelet algorithm of the original signal X (z) goes through a low and high pass analysis FIR (finite impulse response) filter pair G (z) and H (z), and the results in both routes are sampled by a factor of 2. To reconstruct the original signal, the low and high pass coefficients γ(z) and λ(z) are upsampled by a factor of 2 and pass through another pair of synthesis FIR filters G'(z) and H'(z) as

$$\begin{aligned}G'(z)G(z^{-1})+H'(z)H(z^{-1})&=2\\ G'(z)G(-z^{-1})+H'(z)H(-z^{-1})&=0\end{aligned} \tag{10}$$

Let the *z*-transform of a FIR filter *H (z)* be represented by a Laurent polynomial or Laurent series,

$$H(z)=\sum_{k=p}^{q}h(k)z^{-k} \tag{11}$$

Degree of a Laurent polynomial *h* be defined as:

$$|h|=q-p \tag{12}$$

Laurent polynomial *r*(*z*) with |*r*(*z*)| < |*b*(*z*)| so that:

$$a(z)=b(z)q(z)+r(z) \tag{13}$$

Split the z-transform of the signal/filter in even and odd parts

$$H(z)=\sum_n h(n)z^{-n}\left\langle\begin{matrix}H_e(z)=\sum_n h(2n)z^{-n} & \text{Even Part}\\ H_o(z)=\sum_n h(2n+1)z^{-n} & \text{Odd Part}\end{matrix}\right. \tag{14}$$

Converted to and from the original z-transform with the following equations:

$$\begin{aligned}H(z)&=H_e(z^2)+z^{-1}H_o(z^2),\\ H_e(z)&=\frac{1}{2}\left[H(z^{1/2})+H(-z^{1/2})\right],\\ H_o(z)&=\frac{1}{2}z^{1/2}\left[H(z^{1/2})-H(-z^{1/2})\right].\end{aligned} \tag{15}$$

Rewriting the wavelet filtering and subsampling operation using the even/odd parts of the signal and filter as:

$$\begin{aligned}\gamma(z)&=G_e(z)X_e(z)+z^{-1}G_o(z)X_o(z),\\ \lambda(z)&=H_e(z)X_e(z)+z^{-1}H_o(z)X_o(z).\end{aligned} \tag{16}$$

Converting the formula to a matrix form, considering P(z) is a dual polyphase matrix,

$$\bar{P}(z)=\begin{bmatrix}G_e(z) & G_o(z)\\ H_e(z) & H_o(z)\end{bmatrix}. \tag{17}$$

The wavelet transform is invertible as two polyphase matrices are inverse of each other

$$P'(z)=P(z)^{-1}=\frac{1}{H_o(z)G_e(z)-H_e(z)G_o(z)}\begin{bmatrix}H_o(z) & -G_o(z)\\ -H_e(z) & G_e(z)\end{bmatrix}. \tag{18}$$

Also the inverse filters have the following simple relationship with the forward filter:

$$G'_e(z)=H_o(z),\qquad G'_o(z)=-H_e(z),\qquad H'_e(z)=-G_o(z),\qquad H'_o(z)=G_e(z), \tag{19}$$

VOL. 10, NO. 5, MARCH 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Starting with a complementary filter pair (g,h), let us assume that the degree of filter g is larger than that of filter h. We seek a new filter g^{new} which satisfies:

$$g(z) = h(z)t(z^2) + g^{new}(z). \quad (20)$$

where $t(z)$ is a Laurent polynomial

$$P(z) = \begin{bmatrix} H_e(z)t(z)+G_e^{new}(z) & H_o(z)t(z)+G_o^{new}(z) \\ H_e(z) & H_o(z) \end{bmatrix} = \begin{bmatrix} 1 & t(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} G_e^{new}(z) & G_o^{new}(z) \\ H_e(z) & H_o(z) \end{bmatrix} = \begin{bmatrix} 1 & t(z) \\ 0 & 1 \end{bmatrix} P^{new}(z) \quad (21)$$

The new filter g^{new} is complementary to filter h, as the polyphase matrix satisfies:

The lifting wavelet can be directly inversed as each stage matrix in can be directly inversed

$$P'(z) = P(z)^{-1} = \begin{bmatrix} 1/K_1 & \\ & 1/K_2 \end{bmatrix} \prod_{i=m}^{0} \left\{ \begin{bmatrix} 1 & 0 \\ -s_i(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -t_i(z) \\ 0 & 1 \end{bmatrix} \right\} \quad (22)$$

Figure-13. Size 7.13KB.

Figure-14. Size 8.99 KB.

Figure-15. Size 53.1KB.

The algorithm results into multiple compression factors, we assume 0.2 compression factor

For the Figures 13, 14 and 15 the Size during raw information storage is 7.13, 8.99 and 53.1and Size during information retrieval is 1.42, 1.79 and 10.62.

Hence forth the list the most popular lossy compression techniques for image data is tested.

3. CLOUD STORAGE ENVIRONMENTS AND PERFORMANCE COMPARISON FOR IMAGE DATA

We understand the maximized world of options for cloud storage [5] and selected few most popular for testing the data storage and retrieval for multiple image files load. The most popular cloud based storage solutions are DropBox based on Amazon Cloud Service, OneDrive (Formerly known as SkyDrive) based on Microsoft Azure and Google Drive based on Google App Engine cloud service [6, 8].

Table-1. Comparison of upload / download speed in seconds for drop box [7, 9].

Image type and size	Test 1			Test 2			Test 3			Avg.		
	Drop box	Google drive	One drive	Drop box	Google drive	One drive	Drop box	Google drive	One drive	Drop box	Google drive	One drive
10 MB BMP file	15	13	29	11	12	31	10	13	28	12	13	29
10 MB PNG file	15	13	29	11	12	31	10	13	28	12	13	29
10 MB TIFF file	15	13	29	11	12	31	10	13	28	12	13	29
10 MB JPEG file	15	13	29	11	12	31	10	13	28	12	13	29
10 MB JPEG 2000 file	15	13	29	11	12	31	10	13	28	12	13	29

Hence the test results show the type of image information file does not imply any difference on the upload and download speed on various storage solutions.

4. FRAMEWORK FOR STORAGE COMPARISON

The above test results is been generated using a custom build application, which compares and records the response time for upload and upload operation. The application also storages the complete statistics for all information exchange operations in a database for further analysis.

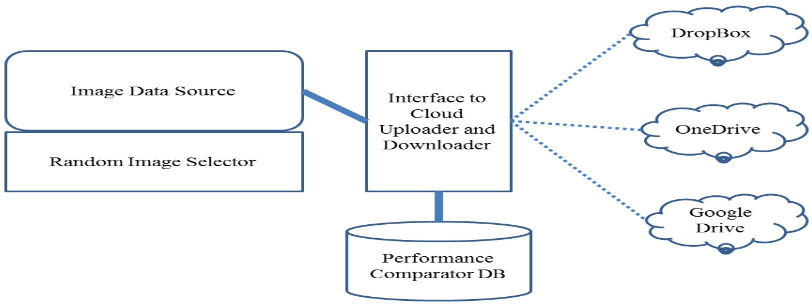

Figure-16. Performance evaluator application for cloud storage.

Here we describe the application with its components

- **Image data source and random image selector**

The solution first incorporates near about 2000 images which are stored in their respective file format and are to be selected randomly for upload and download test and the randomized java code is to select one single image from the image source and supply to uploader and downloader for testing. This component ensures multiple types for constant internet speed.

- **Interface to cloud upload and downloader and performance comparator database**

The interface holds the security details for connection and ensures the constant uploading and downloading of the images selected by Selector program component. This component is also responsible for recording the upload and download time and stores into the database and the database is connected to the Interface and configured locally to store the upload and download time statistics.

- **Cloud storage solutions**

Multiple subscription based cloud storages are used to test the performance in terms of seconds in time for upload and download operations.

5. CONCLUSION AND FUTURE WORK

The work has analyzed the most popular compression algorithms in the space of image compression and recorded the algorithm facts based on the compression ratio. This work has also proven that, the type of image information file types will not impact the speed for upload and download operation. In continuation, the work also results in a tool for comparing the performance of multiple cloud storage services or solutions.

We propose a future extension for the same work to result into an algorithm for automatic selection of image information file systems and respective storage structures based on the information to be transmitted over internet.

REFERENCES

[1] J. J. Ding and J. D. Huang. 2008. Image Compression by Segmentation and Boundary Description.

[2] R. C. Gonzalez and R. E. Woods. 2002. Digital Image Processing 2/E. Upper Saddle River, NJ: Prentice-Hall.

[3] W. B. Pennebaker, J. L. Mitchell. 1993. JPEG Still Image Data Compression Standard. Van Nostrand Reinhold.

[4] The JPEG web page: http://www.jpeg.org/.

[5] H. Wang, R. Shea, F. Wang, and J. Liu. 2012. On the Impact of Virtualization on Dropbox-like Cloud File Storage/Synchronization Services. In: Proceedings of the IEEE 20th International Workshop on Quality of Service, IWQoS '12, pages 11:1-11:9.

[6] A. Li, X. Yang, S. Kandula, and M. Zhang. 2010. CloudCmp: Comparing Public Cloud Providers. In: Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement, IMC'10, pages 1-14.

[7] Amazon. Cloud Drive v. 2.0.2013.841. http://www.amazon.com/gp/feature.html?docId=1000828861.

[8] N. Bermudez, S. Traverso, M. Mellia, and M. M. Munaf`o. 2013. Exploring the Cloud from Passive Measurements: the Amazon AWS case. In: The 32nd Annual IEEE International Conference on Computer Communications, INFOCOM'13.

[9] Drago, M. Mellia, M. M. Munaf`o, A. Sperotto, R. Sadre, and A. Pras. Inside Dropbox: Understanding Personal Cloud Storage Services. In: Proceedings of the 12th ACM Internet Measurement Conference, IMC'12, pages 481–494, 2012.