Healthcare Cloud Integration using Distributed Cloud Storage and Hybrid Image Compression

Sherif E. Hussein
Computer and Systems Department
Faculty of Engineering
Mansoura University, Egypt

Sherif M. Badr
Computer Science Department
Faculty of Computer Science
Modern Academy, Egypt

ABSTRACT

In recent years, the development and demand of multimedia product grows increasingly fast, contributing to insufficient bandwidth of network and storage of memory device. Therefore, the theory of data compression became more significant for reducing data redundancy to save more hardware space and transmission bandwidth. Cloud computing on the other hand; provides elastic services, high performance and scalable data storage to a large and everyday increasing number of healthcare users. Today, clouds are mainly used for handling highly intensive computing workloads and for providing very large data storage facilities. Both goals are combined with a third goal of potentially reducing healthcare data storage cost. In this research, distributed cloud storage that can interact with many cloud providers was used as a backend while hybrid image compression/decompression technique was used in the front end.

General Terms

Medical Imaging, Cloud Computing.

Keywords

Distributed cloud storage, image compression, healthcare data.

1. INTRODUCTION

Due to the wide adoption of modern medical imaging techniques in clinical diagnosis, like CT (Computed Tomography) and MRI (Magnetic Resonance Image), large amount of medical images are generated each day. In order to facilitate long-term storage and efficient transmission, compression techniques have to be applied to these data. Different from natural images, medical images generally have two special issues that should be noted in compression. First, medical images are sensitive to compression errors. Large distortion arising from lossy compression may invalidate their diagnostic values. Second, medical images, especially the monochromatic ones, usually have extended dynamic ranges. Each pixel typically contains 12 or 16 bits per channel, compared to the 8-bit-depth pixels of common natural images [1].

Lossless compression schemes provide reconstructed images identical to the original ones, but suffer from relative low compression ratios, typically between 2:1 and 4:1. Lossy compression schemes, on the other hand, can achieve much higher compression ratio by allowing some distortion in reconstruction. At high quality lossy levels (10:1–20:1), compression rates much greater than those obtained by lossless methods can be obtained while achieving visually indistinguishable results. That is, the human eye cannot detect a difference between the original image and the compressed-then-decompressed image with the lossy compression method. However, the lossy distortion level may have to be supervised by qualified physicians to avoid possible legal and diagnostic problems [2].

2. PREVIOUS WORK

There have been numerous compression research studies examining the use of compression as applied to medical images. Those studies can be categorized as focusing on just a lossless compression method or focusing on both lossless and lossy compression. Most have focused on lossless algorithms since the medical community has been reluctant to adopt lossy techniques owing to the legal and regulatory issues that are raised and the regulatory policies set by agencies such as the Food and Drug Administration, although this situation may start to change as more lossy research is performed [3].

Lossless image compression is typically performed in two steps, decorrelation and coding. Image decorrelation attempts to reduce the redundancy within the image then coding with a basis set. There are several common approaches that have been adopted in literature to perform this redundancy reduction step including differential pulse code modulation, hierarchical interpolation, and bit-plane encoding and multiplicative auto regression. Several popular approaches for encoding are Huffman, Lempel-Ziv, run-length, and arithmetic encoding.

In Huffman Coding, the characters in a data file are converted to a binary code. The most common characters in the input file (characters with higher probability) are assigned short binary codes and least common characters (with lower probabilities) are assigned longer binary codes. Codes can be of different lengths [4]. Lempel-Ziv-Welch uses a dictionary or code table done by constructing a "dictionary" of words or parts of words in a message, and then using pointers to the words in the dictionary. LZW is used to compress text, executable code, and similar data files to about one-half their original size. A higher compression of 1:5 is also possible as reported in [5] Run Length Encoding is data coding with frequently repeated characters. It is called run-length because a run is made for repeated bits and coded in lesser bits by only stating how many bits were there [6]. In Arithmetic Coding, message is encoded as a real number in an interval from 0 to 1. However, the whole codeword must be received to start decoding. If there is a corrupt bit in the codeword, the entire message could become corrupt. Besides, symbols to encode within a code word are limited [7].

The Huffman and Lempel-Ziv encoding methods were compared as applied to MRI images in [8] to show that Huffman encoding methods achieve higher compression than compression ratios resulting from using Lempel-Ziv encoding.

As mentioned earlier, lossless methods are preferred in the medical community. Of these methods, JPEG and wavelet are most popular. These two compression methods actually gained widespread acceptance as lossy methods. However, each can be made lossless which is the preferred style in medical imaging.

Chambolle et al. [9] showed that wavelet transform is a promising tool for image compression providing high rates of compression while maintaining good image quality. Iyriboz et al. [10] performed a quantitative comparison of three lossy compression methods (one wavelet and two JPEG) as applied to a variety of 12-bit medical images in conjunction with the Department of Radiology at the Hershey Medical Center (Hershey, PA). This work showed the quality of JPEG and wavelet-based compression.

Chen and Tseng [11] proposed a wavelet-based compression scheme with an adaptive prediction for medical images. They proposed as well the adaptive prediction approach to overcome the multicollinearity problem and employ three prediction equations for different wavelet subbands to achieve a more accurate prediction. The proposed WCAP method and four famous lossless compression methods: SPIHT, JPEG2000, CALIC, and SSM, were experimentally compared. Their experimental results showed that the proposed WCAP method almost achieved the highest compression rates for CT, MRI, and ultrasound images; moreover, WCAP had individually improved 20.7, 12.6, 8.02, and 3.47% compression rates with respect to SPIHT, JPEG2000, CALIC, and SSM methods in average.

With regard to clinically relevant region encoding, Chen et al. [12] made use of regions of interest using sub band analysis and synthesis or volumetric datasets using wavelets. They followed up this work using structure preserving adaptive quantization methods as a means of improving quality for compression rates in the regions of interest. But all of their effort was on lossy approaches. On the other hand, El-Sonbaty et al. [13] used enhanced fuzzy based models to extract ROI in medical imaging.

Sun et al. [14] proposed a value of interest VOI-based compression scheme for medical images with extended bit depth, which considered the user side display during the compression process. Given the recommended VOI mapping parameters in the display, the original image was decomposed into a VOI layer and an enhancement layer. The VOI layer image only contained information of the VOI pixels to reduce the transmission delay of these pixels. The enhancement layer image was encoded using Gray-Golomb codes to support VOI scalability. They applied pixel domain bit plane coding for the both layers, which enabled pixel level distortion estimation for arbitrary truncated bit-stream. Compared to JPEG2000, there scheme could significantly reduce the bandwidth cost and transmission delay, with very slight overall bit-rate increase.

Zukoski and Boult [15] proposed an expansion to the residual approach is still experimental, this paper has shown that the overall approach of clinically relevant regions has clearly demonstrated advantages over both traditional lossless compression and simple lossy compression, and that the residual approach holds some promise. A larger scale study in conjunction with Penn State University’s Hershey Medical Center is ongoing. This study will include a qualitative component to examine the effects of the hybrid compression methodology in a clinical setting.

Storm and Cosman [16] developed a region based coding approach. They discussed two approaches: one uses different compression methods in each region such as ‘contour-texture’ coding and sub band decomposition coding, and the other uses the same compression method in each region such as the discrete cosine transform but with varying compression quality in each region such as by using different quantization tables. They used two multiresolution coding schemes: wavelet zero tree coding and the S-transform, and considered only 8 bit images. In their implementation, the regions of interest were selected manually.

Goel and Chandra [17] proposed a secure technique for medical image storage with inter-pixel image encryption technique which can be embedded into the existing PACS (Picture Archiving and Communication System) to provide more strength with present features and zooming system which enables physicians to diagnose the diseases with patient records more accurately and on basis of this diagnosis further treatment can be advised. However, more focus need to be given into the zooming part using the existing PACS system.

Some compression schemes have been adopted by the Digital Imaging and Communications in Medicine (DICOM) standard, among which JPEG2000 is considered to have a few good properties. It provides both lossy and lossless compressions for images with bit depth no more than 16 bit per channel, and allows progressive transmission with SNR and spatial scalability. In particular, it supports ROI (region of interest) coding, so that image regions with diagnostic importance can have much higher quality than the other parts. The most popular compression algorithms in use today in the medical community are lossless JPEG (Joint Photographic Experts Group) and lossless wavelet. JPEG has been adopted by the DICOM group in their widely adopted image file format, but the wavelet compression algorithm is gaining ground. In fact, the DICOM Working Group added support for the JPEG 2000 standard into the DICOM format in November of 2001. It has also been adopted by ISO as a standard. JPEG 2000 is based on wavelet compression. The new compression approach, which will be proposed here utilize a hybrid lossy/lossless method, which can be made all lossy or all lossless.

3. THE PROPOSED MODEL

In this research, it was proposed a novel lossy and lossless compression scheme for medical images with extended bit depth. The method developed a region based coding approach. In the research implementation, the regions of interest were selected manually those regions have been compressed using a lossless scheme while the rest of the image was compressed using lossy compression.

3.1 Methodology

A typical 12-bit medical X-ray may be 2048 pixels by 2560 pixels in dimension. This translates to a file size of 10,485,760 bytes. A typical 16-bit mammogram image may be 4500 pixels by 4500 pixels in dimension for a file size of 40,500,000 (40 megabytes). This has consequences for disk storage and image transmission time. Even though disk storage has been increasing steadily, the volume of digital imagery produced by hospitals and their new filmless radiology departments has been increasing even faster. Even if there were infinite storage, there is still the problem of transmitting the images. Many hospitals have satellite centers or clinics in small towns and remote areas to make it convenient for patients who have a hard time traveling the distance to the hospital, especially for diagnostic procedures.

These hospitals make use of ‘teleradiology’ applications that allow the clinic staff to operate the clinic without the need for a radiologist to be present. Instead of a diagnostic radiologist, a technician or basic radiologist in the clinic can take the X-ray and send the image through a network connection to the hospital where the diagnostic radiologist can read the image and send back a diagnosis [18]. But there is a problem, especially in emergency situations where time is of the essence, because a 10 MB image will take approximately half an hour using a high-speed modem. Broadband connections such as T1 lines improve the situation, but many clinics are in such remote areas, that it is cost prohibitive to set up high-speed lines. DSL typically has very asymmetric performance with high-speed downlinks and much slower uplinks, and the uplinks are what would limit the satellite facility. Even at standard Cable/DSL uplink speeds, a single 10 MB radiograph will take over five minutes to be sent. While that may sound reasonable, keep in mind that the patient is often asked to remain in the imaging apparatus until the radiologist has confirmed that the data is sufficient. So compression is not just about the storage costs, it is also about transmission time, imaging apparatus utilization and convenience/comfort of the patient. Compression techniques can reduce file size and transmission time, thus improving overall care. Image compression techniques take advantage of redundancy that occurs. There are different types of redundancy which are spatial, temporal and spectral. The research presented here will focus on the first of these three types of redundancies although the techniques can be used in the others also.

3.1.1 Basic types of compression

In lossless compression data is compressed without any loss of data. It assumes you want to get everything back that you put in i.e., a perfect reproduction of the original from the compression can be reconstructed.

With lossy compression, it is assumed that some loss of information is acceptable. When you reconstruct the information from the compressed data, you get something close to but not exactly the same as the original. You can get more compression by allowing the algorithm to lose more information. Lossy compression algorithms usually have an adjustable parameter to control this compression vs. quality tradeoff [19]. The best example is a videoconference where there is an acceptable amount of frame loss in order to deliver the image in real time. People may appear jerky in their movements, but you still have a grasp for what is happening on the other end of the conference. In the case of graphics files, some resolution may be lost in order to create a smaller file. The loss may be in the form of color depth or graphic detail. For example, high-resolution details can be lost if a picture is going to be displayed on a low-resolution device. Loss is also acceptable in voice and audio compression, depending on the desired quality.

Huffman Encoding is used in this research due to its superior properties in medical image compression [7]. It is a general technique for coding symbols based on their statistical occurrence frequencies (probabilities). The pixels in the image are treated as symbols. The symbols that occur more frequently are assigned a smaller number of bits, while the symbols that occur less frequently are assigned a relatively larger number of bits. Huffman code is a prefix code. This means that the (binary) code of any symbol is not the prefix of the code of any other symbol.

3.1.2 The hybrid compression approach

A new hybrid medical image compression approach that treats significant regions and non-significant regions differently is proposed. The regions will classify areas of the image into those that are clinically relevant and those that are not clinically relevant. The physician currently working with the image to be compressed will choose the relevant regions from the image.

Lossless compression will be used in the clinically relevant regions and lossy compression will be used in areas that are not clinically relevant. Most lossy compression algorithms such as JPEG, utilize a compression amount parameter that defines the amount of compression, and hence degradation, used on the image. Since the clinically relevant regions, by definition, delineate the regions that the physicians care about, the amount of compression performed in the clinically non relevant regions can be made very high. These regions will basically be used to provide a frame of reference to the physician and so long as the physician can make out in general what these clinically non relevant regions are, then all is well.

In the proposed method significant regions were manually selected contrary to [15] in which a pre stored database of common film types such as X-rays of the head and chest can be stored.

One advantage of this method is that it does not matter what the underlying data type is. It can be 2D or it can be 3D. The point of interest is the level of compression data rate that can be achieved. It is possible with this technique to get very high compression rates with many regions of interest as will be discussed in the results section.

The front end has been developed using ImageJ which is a public domain, Java-based image processing program developed at the National Institutes of Health [20]. ImageJ was designed with an open architecture that provides extensibility via Java plugins and recordable macros. Custom acquisition, analysis and processing plugins can be developed using ImageJ's built-in editor and a Java compiler. User-written plugins make it possible to solve many image processing and analysis problems, from three-dimensional live-cell imaging [21], to radiological image processing [22], multiple imaging system data comparisons [23] and automated hematology systems [24]. ImageJ can read many image formats including TIFF, PNG, GIF, JPEG, BMP, DICOM, FITS, as well as raw formats. It supports image stacks, a series of images that share a single window, and it is multithreaded, so time-consuming operations can be performed in parallel on multi-CPU hardware. It can as well measure distances and angles create density histograms and line profile plots. The web application program supports any number of images simultaneously, limited only by available memory.

3.2 Compression results

To provide supporting evidence for this research, the idea of clinically relevant regions has been applied to several medical images and the file sizes were compared between the image entirely compressed lossly, and the image that used the combined lossless/lossy approach. The compression system was applied to 8 X-ray images taken from male and female adults. Each image used was represented at 8 bits per pixel (bpp).

The first image was captured directly from the patient using a high-resolution 12-bit digital X-ray scanner. The image was then down sampled to 8 bits. Figure 1 shows an example image with three clinically relevant regions defined on it. Those regions have been marked by a radiologist to be lossless compression. The rest of the image has been compressed lossy compression. A JPEG level of 50 (of 100) was chosen for the lossless compression.

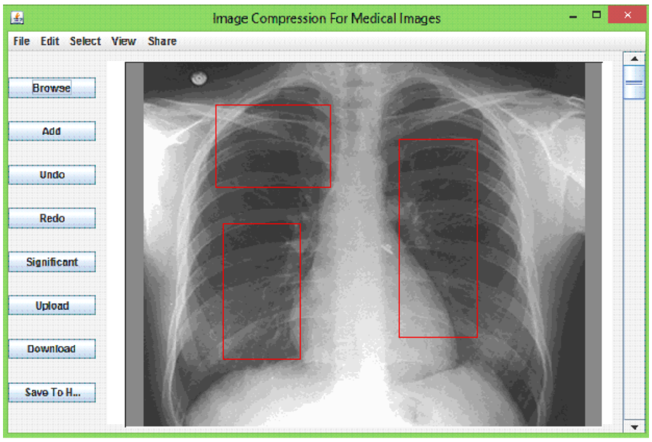

Fig. 1: The compression/decompression front end for the X-Ray image

The original file size of the uncompressed raw image is 5231556 bytes using 8 bits per pixel. With the entire image compressed using lossless JPEG2000, the compressed file size is 2972475 bytes. This gives a compression ratio of 1.76: 1 or 4.546 bpp.

Regions 1, 2 and 3 were compressed using lossless JPEG., The rest was compressed using lossy JPEG at a compression level of 50 (out of 100). The results are shown in Table 1.

So using the clinically relevant region approach, the compressed file size is 963576 bytes. This gave a compression ratio of 5.429:1 or 1.474 bpp. The clinically relevant region area took up almost 25 % of the entire image area. The compression ratio had tripled from the 1.76: 1 ratio for the pure lossless technique to 5.429:1 for the hybrid compression technique.

Table 1 Region compressions. Overall compression rate is 1.474 bpp, almost three times smaller than pure lossless

Region	Raw bytes	Type	Compressed
1	575471	Lossless	338512
2	418524	Lossless	246190
3	381903	Lossless	224648
4	3855658	Lossy	154226
Total	5231556	–	963576

For the chest X-rays, radiologists outlined significant areas of the lung areas to be compressed lossless, and the remaining area to be compressed loosely but at high quality at a level of 50 of 100. A collage of all 8 chest X-rays is shown in Figure 2.

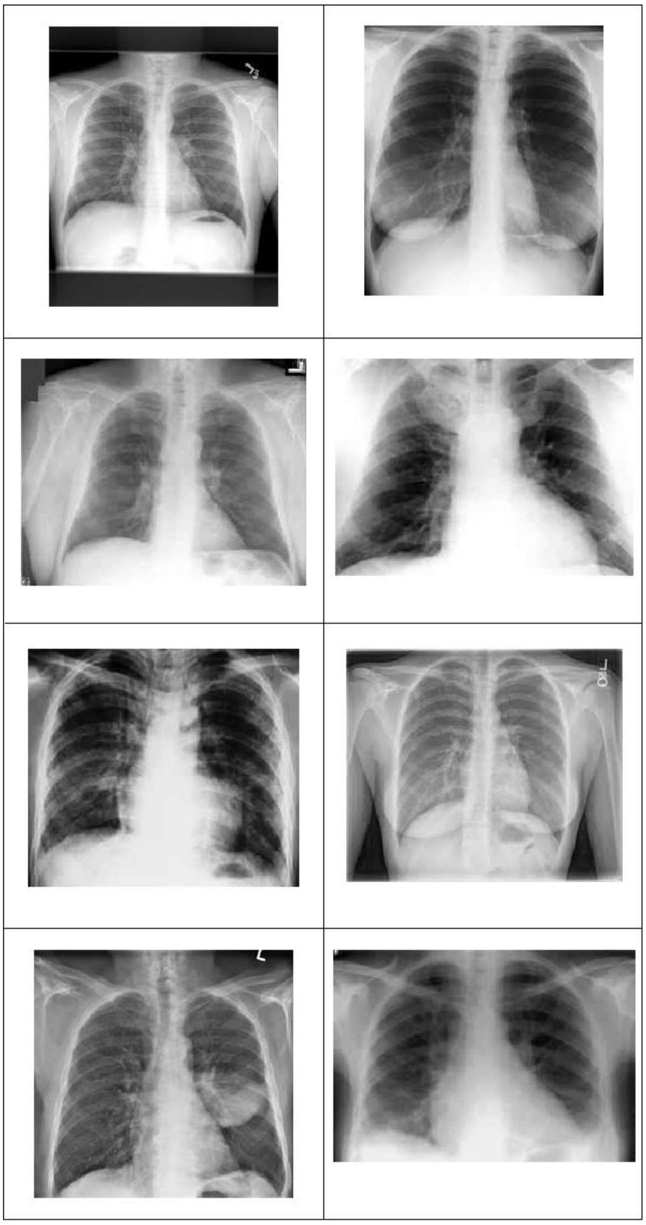

Fig. 2: Chest X-ray collage (8 bpp each)

Table 2 shows the resulting compression ratio from using the lossless only compression and from hybrid compression, as well as the bits per pixel resulting from the lossless-only and from the hybrid compression technique. The last column shows the savings resulting from using the hybrid compression methodology as compared to using just lossless compression. On an average, the hybrid compressed file size was 3.5 times smaller than its corresponding lossless-only compressed image. This translates to an average file size that is 29% smaller than the lossless-only compressed file size.

Table 2 Hybrid compression results

Image no.	Lossless (bpp)	Hybrid compress ratio	Lossless compress ratio	Hybrid (bpp)
1	4.7	5.2:1	1.7:1	1.6
2	4.7	5.4:1	1.7:1	1.5
3	5.4	4.3:1	1.5:1	1.8
4	4.8	4.4:1	1.7:1	1.8
5	4.6	7.5:1	1.7:1	1.1
6	5.1	4.7:1	1.6:1	1.8
7	4.6	10.4:1	1.7:1	0.8
8	4.7	4.2:1	1.7:1	1.9

4. DISTRIBUTED CLOUD STORAGE

Distributing data may be achieved by Distributed Systems or Distributed cloud storage.

The distributed system is a collection of independent computers interconnected via point to point communication lines. Each computer, known as a node in the network, has a processing capability, a data storage capability and is capable of operating autonomously in the system. The database is allocated across system nodes in units of distributed relations which allow a general manner of redundancy.

A distributed database is a collection of data which belong logically to the same system but are spread over the sites of a computer network. In other words: a distributed database system is characterized by the distribution of the system components of hardware, control, and data [26].

The only allocation constraint in that all data must be either locally or globally accessible from any system node. The distributed of data on the network is invisible to the user.

The Distributed Cloud computing has recently appeared as a new computing paradigm, which promises virtually unlimited resources. Customers rent resources based on the pay-as-you-go model and thus are charged only for what they use. Opposite to other service models, in-house PACS and application service provider PACS [25], cloud computing offers relatively lower cost, higher reliability and scalability.

4.1 Motivation

Cloud storage services differ in pricing schemes and performance characteristics. Some providers charge a flat monthly fee, others negotiate contracts with individual clients, and still others offer discounts for large volume or temporary promotional incentives or lower bandwidth rates for off-peak hours. Some providers may be desirable for geographic reasons, and others may offer extra features such as publishing files directly to the web or access via mountable file system. Changes in these features, or the emergence of new providers with more desirable characteristics, might persuade some clients to switch from one storage service to another. However, because of storage inertia, clients may not be free to choose the optimal vendor due to prohibitively high costs in switching from one provider to another. This puts the client at a disadvantage when a vendor raises prices or negotiates a new contract. A client at an old provider has to make an all-or-nothing decision about his storage service. He either moves all his data to a new provider and incurs the cost of switching to gain the benefit of lower prices, or stays at the old provider and saves the cost of switching at the risk of incurring higher costs on the long run. The problem, of course, is that the client cannot tell if and when his current provider will lower the pricing scheme to compete with the other providers. From the client’s point of view, a premature switch can be worse than no switch, and a late switch might have lost its value. Of course, this is not limited to price changes. Clients might be stuck with underperforming providers simply because switching to better providers is deemed too costly. One way to allow clients to be more agile in responding to provider changes is by making the switching and data placement decisions at a finer granularity than a global all-or-nothing. This can be achieved by striping the client’s data across multiple providers. For example, if a client spreads his data across several providers, taking advantage of a new fast provider entails switching only a fraction of the data, which could be more economically feasible [27].

In order to avoid vendor lock-in, data has to be hosted by multiple storage providers. However, despite being simple and reactive, storing full replicas of the same data is too costly [28], with the aid of erasure coding [29].

Given customer, such as data durability, data availability or independence from cloud providers to avoid vendors lock-in, it then becomes a non-trivial task to find the cloud storage provider or combinations of cloud storage providers that offer the best price to store users’ data. To make things worse, the ratio of read/write operations of a data object over a period of time affects the resulting charging for the customer, as providers implicitly promote certain access patterns with their pricing policies. Scalia software used in this research provides an engine that optimizes the placements of data chunks following the rules set by the data owner, while also taking into account the access patterns of the data in order to compute the cheapest provider set [30]. A default rule, rules per data object classes or rules per data object, can be defined in Scalia (e.g., using an API or a dashboard), so as to specify the availability, the durability, the geographical zone(s) and the lock-in factor of the data

4.2 Engine layer

The engine acts as a proxy between the client and the cloud storage providers, offering a unified API to all providers, including data storage to private resources. Mainly, it is responsible for storing the chunks of data to the best providers according to the optimization goals, and serving the data either directly from the cache or by reconstructing it using the chunks stored at the remote providers. The engine also tries to maintain the optimality of the chunk placement of an object, by periodically computing the best provider set using the data access statistics. The sampling period s is a time period where the statistics per object are collected and aggregated, typically 1 hour. Knowing the recent access history of a data permits to precisely adjust the set of providers, as could be reasonably supposed that the access pattern of the data in the near future will be similar to the current. Choosing a large decision period allows predicting the access pattern farer in the future, and thus permits to make better placement choices in the long run. However, imagine that the chunks of a data object were placed based on the assumption that the object would be stored for at least 6 months, and the object was in fact deleted after 1 week. The chosen placement would have been probably wrong, resulting in higher costs for the end user. Thus, the decision period has to be dynamically adjusted as it depends on the lifetime of the object, the burstiness of its access pattern and the resulting economic impact of the latter [31].

4.3 Active repair and cloud storage security

In the case of a transient failure of a cloud storage provider, Scalia may adopt two strategies to cope with the unavailability of providers: either do nothing and simply wait for the provider to recover, or move the chunks hosted at the faulty provider to another provider. However, the latter procedure comes at a relatively high cost: in order to move the chunk of the faulty provider, the data object needs to be reconstructed from the remaining chunks and split again into chunks. Depending on the available providers, the threshold m of the most cost effective providers set may be different. In that case, all chunks need to be re-written. If m is the same, then only the faulty chunk needs to be written, which corresponds to the cheapest case. When the cloud data security is considered, the security triad must be taken into account: confidentiality, integrity, and availability. In the cloud storage model, data is stored on multiple virtualized servers. Physically the resources will span multiple servers and can even span storage sites. Among the additional benefits of such generally low-cost services are the storage maintenance tasks (such as backup, replication, and disaster recovery), which the CSP performs. The most notable provider in this space is Amazon with its S3. A common aspect of many cloud-based storage offerings is the reliability and availability of the service. Replication of data is performed at a low level by such mechanisms as RAID or by a file system. One such file system is ZFS, which was designed by Sun Microsystems as both a file system and a volume manager. ZFS supports high storage capacities and performs numerous security relevant functions including copy-on-write cloning and continuous integrity checking along with automatic repair [18].

Figure 3 depicts a typical cloud storage gateway application as it is used to augment local storage by acting as an onsite secondary copy and as an intermediary to the CSP storage service.

5. CONCLUSION

In this research, a hybrid compression technique was presented that selectively compresses significant regions lossless compression using Huffman compression. The remaining of the image was compressed using lossy JPEG at a compression level of 50. The significant regions were manually selected with a radiologist. The compression results showed compression ratios in the range from 4 to 10 which are higher than those reported in literature. The compression algorithm was implemented in a web application front end to compress/decompress the medical images before to be sent to and after to be received from the back end.

The back end utilized Scalia which is a system that continuously optimizes the placement of data stored at multiple cloud providers, based on their access statistics. Scalia mediates data placement across multiple public cloud providers and private cloud resources. It helps the data owners to avoid vendor lockin and satisfy certain availability and durability constraints in a cost-effective way.

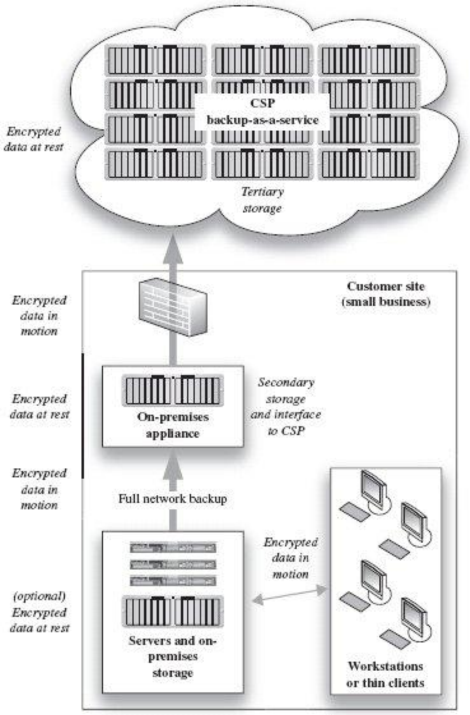

Fig. 3: Cloud storage gateway appliance

Those are key points for healthcare data storage which can increment in size astronomically with cost and safety constraints. This research combined a hybrid compression technique to decrease the required storage space with a distributed cloud storage layer to decrease the storage cost, enforce security measures and increase the storage performance.

References:

[1] Hong Zhu, Weizhen Sun, Minhua Wu, Guixia Guan, Yong Guan "Pre-Processing of X-Ray Medical Image Based on Improved Temporal Recursive Self-Adaptive Filter," Young Computer Scientists, International Conference for, November 2008, pp. 758-763.

[2] S. Anand, R. Shantha Selva Kumari, S. Jeeva, T. Thivya "Directionlet transform based sharpening and enhancement of mammographic X-ray images," Biomedical Signal Processing and Control, Volume 8, Issue 4, July 2013, 391-399.

[3] J.Janet, Divya Mohandass and S.Meenalosini .Lossless Compression Techniques for Medical Images In Telemedicine. InTechOpen, Published on: 2011-03-16. DOI: 10.5772/14399.

[4] Pablo Montero, Javier Taibo, Victor Gulías, Samuel Rivas, "Parallel Zigzag Scanning and Huffman Coding for a GPU-based MPEG-2 Encoder," ism, pp.97-104, 2010 IEEE International Symposium on Multimedia, 2010.

[5] Sergio De Agostino, "Lempel-Ziv Data Compression on Parallel and Distributed Systems," ccp, pp.193-202, 2011 First International Conference on Data Compression, Communications and Processing, 2011.

[6] Mustafa Safa Al-Wahaib, KokSheik Wong, "A Lossless Image Compression Algorithm Using Duplication Free Run-Length Coding," netapps, pp.245-250, 2010 Second

International Conference on Network Applications, Protocols and Services, 2010.

[7] I. Sodagar, B.-B. Chai, J. Wus, "A new error resilience technique for image compression using arithmetic coding," icassp, vol. 4, pp.2127-2130, Acoustics, Speech, and Signal Processing, 2000 Vol 4. 2000 IEEE International Conference on, 2000.

[8] Cohen, L.D. (1991) ‘On active contour models and balloons’, Computer Vision, Graphics, and Image Processing. Image Understanding, Vol. 53, No. 2, pp.211–218.

[9] Antonin Chambolle, Ronald A. DeVore, Nam-yong Lee, and Bradley J. Lucier. Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. IEEE Trans. Image Process., 7(3):319-335, 1998.

[10] Iyriboz, T.A., Zukoski, M.J., Hopper, K.D. and Stagg, P.L. (1999) ‘A comparison of wavelet and joint photographic experts group lossy compression methods applied to medical images’, Journal of Digital Imaging, May, Vol. 12, pp.14–17.

[11] Yao-Tien Chen, Din-Chang Tseng "Wavelet-based medical image compression with adaptive prediction," Computerized Medical Imaging and Graphics, Pages 1–8, Volume 31, Issue 1, January 2007.

[12] Chen, C.W., Zhang, Y.Q., Luo, J. and Parker, K.J. (1995) ‘Medical image compression with structure-preserving adaptive quantization’, Visual Communication and Image Processing ‘95, Vol. 2501, No. 2, pp.983–994.

[13] Yasser El-Sonbaty,Sherin M. Youssef,Karma M. Fathalla "Enhanced fuzzy-based models for ROI extraction in medical images," International Symposium on Signal Processing and Information Technology, December 2011, pp. 299-304

[14] Wen Sun , Yan Lu, Feng Wu, Shipeng Li "Level embedded medical image compression based on value of interest,"Proceedings of the 16th IEEE international conference on Image processing Pages 1749-1752, NJ, USA, 2009.

[15] Matthew J Zukoski; Terrance Boult; Tunç Iyriboz "A novel approach to medical image compression," International journal of bioinformatics research and applications 2(1):89-103, 2006.

[16] Storm, J. and Cosman, P.C. ‘Medical image compression with lossless regions of interest’, Signal Processing, pp.155–171, Vol. 59, No. 2, 1997.

[17] Amnesh Goel, Nidhi Chandra, "A Prototype Model for Secure Storage of Medical Images and Method for Detail Analysis of Patient Records with PACS," csnt, pp.167-170, 2012 International Conference on Communication Systems and Network Technologies, 11-13 May, 2012.

[18] Fatma E.-Z. A. Elgamal, Noha A. Hikal, F.E.Z. Abou-Chadi "Secure Medical Images Sharing over Cloud Computing environment," (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 4, No. 5, 2013

[19] Jing Tian, Li Chen "Image despeckling using a non-parametric statistical model of wavelet coefficients," Biomedical Signal Processing and Control, Volume 6, Issue 4, October 2011, Pages 432-437.

[20] Collins TJ (July 2007). "ImageJ for microscopy". BioTechniques 43 (1 Suppl): 25–30. doi:10.2144/000112517. PMID 17936939. Open Access logo PLoS transparent.svg.

[21] Eliceiri K, Rueden C (2005). "Tools for visualizing multidimensional images from living specimens". Photochem Photobiol 81 (5): 1116–22. doi:10.1562/2004-11-22-IR-377. PMID 15807634.

[22] Barboriak D, Padua A, York G, Macfall J (2005). "Creation of DICOM—Aware Applications Using ImageJ". J Digit Imaging 18 (2): 91–9. doi:10.1007/s10278-004-1879-4. PMC 3046706. PMID 15827831.

[23] Rajwa B, McNally H, Varadharajan P, Sturgis J, Robinson J (2004). "AFM/CLSM data visualization and comparison using an open-source toolkit". Microsc Res Tech 64 (2): 176–84. doi:10.1002/jemt.20067. PMID 15352089.

[24] Gering E, Atkinson C (2004). "A rapid method for counting nucleated erythrocytes on stained blood smears by digital image analysis". J Parasitol 90 (4): 879–81. doi:10.1645/GE-222R. PMID 15357090.

[25] Michael J. Ackerman, Terry S. Yoo "Open source software for medical image processing and visualization," Communications of the ACM, February 2005, pp. 55-59.

[26] Sherif M Badr “Secured Hierarchically Dependent Distributed Database Model Applied to Hospitals Information System (HIS)”, IJCA (0975 – 8887) Volume 66– No.22, March 2013.

[27] M. Vrable, S. Savage, and G. M. Voelker. Cumulus: Filesystem backup to the cloud. Trans. Storage, 5(4):1–28, 2009.RACS: A Case for Cloud Storage Diversity.

[28] Qinlu He, Zhanhuai Li, Xiao Zhang "Study on Cloud Storage System Based on Distributed Storage Systems," Computational and Information Sciences, International Conference on, pp 1332-1335, December 2010.

[29] Jing Han,Meina Song,Junde Song "A Novel Solution of Distributed Memory NoSQL Database for Cloud Computing," Computer and Information Science, ACIS International Conference on, pp. 351-355, May 2011.

[30] Thanasis G. Papaioannou,Nicolas Bonvin,Karl Aberer "Scalia: An adaptive scheme for efficient multi-cloud storage," 2012 SC - International Conference for High Performance Computing, Networking, Storage and Analysis" pp. 1-10, November 2012.

[31] Hussam Abu-Libdeh, Lonnie Princehouse, Hakim Weatherspoon "RACS: a case for cloud storage diversity," SoCC '10 Proceedings of the 1st ACM symposium on Cloud computing, New York, NY, USA, 2010, pp. 229-240, ISBN: 978-1-4503-0036-0.