ORIENTAL JOURNAL OF
COMPUTER SCIENCE & TECHNOLOGY
An International Open Free Access, Peer Reviewed Research Journal
Published By: **Oriental Scientific Publishing Co., India.**
www.computerscijournal.org

ISSN: 0974-6471
December 2015,
Vol. 8, No. (3):
Pgs. 216-221

Text Based Image Compression Using Hexadecimal Conversion

HEMA SURESH YARAGUNTI[1] and T. BHASKARA REDDY[2]

[1]School of Computer Science C.U.K, Kalaburgi, India.
[2]Department of Computer Science & Technology,S.K.U., Anantapur, India.

(Received: August 10, 2015; Accepted: October 30, 2015)

ABSTRACT

Easy and less-time-consuming transmission of high quality digital images requires the compression-decompression technique to be as simple as possible and to be completely lossless. Keeping this demand into mind, we proposed the lossless image compression technique i.e Text Based Image Compression using Hexadecimal Conversion (TBICH). The key idea is to increase redundancy of data presented in an image to get the better compression. We are concerned with lossless image compression in this paper .Our proposed technique is Text Based Image compression using Hexadecimal conversion (TBICH). Our approach works as follows: firstly an image is encoded and converts it into text file by using hexadecimal numbers and LZ77 text compression is applied on this encoded file to reduce size. While decompressing, decompress the encoded file and convert into image which is same as original image. The main objective of this proposed technique is to reduce the size of the image without loss and also to provide security.

Key words: LZ77, CR, TBICH.

INTRODUCTION

Image processing applications are drastically increasing over the years. Storage and transmission of digital image has become more of a necessity than luxury these days, hence the importance of Image compression. Image security has become most important aspect while transmission, exchanging and storage of an image. Most of compressions techniques are based on properties of inter pixel redundancy, pixel differencing, ordering and differencing etc. The famous LZ77 for lossless data compression used a somewhat different approach which is related to our task.

Image compression can be achieved via coding methods, compression methods or combination of both the methods[15]. Coding directly applied to the raw images treating them as a sequence of discrete numbers. Common coding methods include Arithmetic, Huffman, Lempel_ ziv_Welch(LZW)[18], and Run length. Spatial domain methods, which are a combination of spatial domain algorithms and coding methods, not only operate directly on the grey values in an image but also try to eliminate the spatial redundancy. In transform domain compression,the image is represented using an appropriate basis set ,and goal is to obtain a sparse coefficient matrix.DCT based compression and wavelet transform are examples of transform domain methods.

Three performance metrics are used to evaluate algorithms and choose the most suitable one:
1) Compression ratio,
2) Computational requirements, and
3) Memory requirements.

The computational needs of an algorithm is expressed, in terms of how many operations (additions/multiplications, etc.) are required to encode a pixel (byte). The third metric is the amount of memory or buffer required to carry out an algorithmic or MRI Medical imaging produces human body pictures in digital form. Since these imaging techniques produce prohibitive amounts of data, compression is necessary for storage and communication purposes. Many current compression schemes provide a very high compression rate but with considerable loss of quality. On the other hand, in some areas in medicine, it may be sufficient to maintain high image quality only in the region of interest, i.e., in diagnostically important regions.

So, image compression becomes a solution to many imaging applications that require a vast amount of data to represent the images, such as document imaging management systems, facsimile transmission, image archiving, remote sensing, medical imaging, entertainment, HDTV, broadcasting, education and video teleconferencing.

One major difficulty that faces lossless image compression is how to protect the quality of the image in a way that the decompressed image appears identical to the original one. In this paper we are concerned with lossless image compression. Based on hexadecimal conversion and LZ77 algorithm.

Hexadecimal numbers

Each hexadecimal digit[2,3] represents four binary digits (bits), and the primary use of hexadecimal notation is a human-friendly representation of binary-coded values in computing and digital electronics. One hexadecimal digit represents a nibble, which is half of an octet or byte (8 bits). For example, byte values can range from 0 to 255 (decimal), but may be more conveniently represented as two hexadecimal digits in the range 00 to FF. Hexadecimal is also commonly used to represent computer memory addresses[2,3]. There are many advantages for hexadecimal numbers.

Hex values are easier to copy and paste from image editors, normally easier to remember as it's only 6 digits and can easily become more compact, e.g. #fff vs rgb(255,255,255). Hex requires less character to express the same value. Hex is more webs friendly as well.

Encoding

In computers, encoding is the process of putting a sequence of characters (letters, numbers, punctuation, and certain symbols) into a specialized format for efficient transmission or storage[14,17]. Decoding is the opposite process the conversion of an encoded format back into the original sequence of characters. Encoding and decoding are used in data communications, networking, and storage. The term is especially applicable to radio (wireless) communications systems.

LZ77 (LEMPEL ZIV 1977)

Published in 1977, LZ77[4,5] is the algorithm that started it all. It introduced the concept of a 'sliding window'[7,11] for the first time which brought about significant improvements in compression ratio over more primitive algorithms. LZ77 maintains a dictionary using triples representing offset, run length, and a deviating character. The offset is how far from the start of the file a given phrase starts at, and the run length is how many characters past the offset is part of the phrase. The deviating character is just an indication that a new phrase was found, and that phrase is equal to the phrase from offset to offset length plus the deviating character[8,9]. The dictionary used changes dynamically based on the sliding window as the file is parsed.

The proposed method

The proposed method is an implementation of the loss- less image compression. The objective of the proposed method in this paper is to design an efficient and effective lossless image compression scheme. This section deals with the design of a lossless image compression method. The proposed method is based on encoding using hexadecimal conversion and LZ77 in order to improve the compres-sion ratio of the image comparing to other compression techniques.

The proposed methods is performed in by using following steps,

1. Digitization
2. Encoding
3. Compression

The proposed TBICH technique is one of the techniques where we are converting an image into text format and then lossless text compression techniques are applied on it to reduce the size. The below figures shows the model of proposed technique TBICH.

Image digitization

A digital image is a numeric representation (normally binary) of a two-dimensional image[1]. Depending on whether the image resolution is fixed, it may be of vector or raster type.

An image captured by a sensor is expressed as a continuous function f(x,y) of two co-ordinates in the plane. Image digitization[1] means that the function f(x,y) is sampled into a matrix with M rows and N columns.

Encoding

In this step, integers are converted into binary number and then convert these binary to hexadecimal numbers and stored in text format. This text format file is encoded file, which provides security[14,17]. Now this hexadecimal conversion contains numbers range from 0-16.As we have representing the 0-255 numbers into 0-16 range, redundancy will increase because repetitions only within 0-15 combinations and with this more redundancy will get better compression. The integer numbers takes 8bits space to store, after converting it into hexadecimal its value, which takes 16 bits to store. Due to this reason, the size of the encoded files much larger than original image. It is difficult to store as intermediate file due to time and space complexities. And it is not possible to transmit this heavy file from source to destination. If we want to transfer this need to reduce the size of the file.

Compression and decompression

As encoded file[14] is present in text format and it is best to apply the text compression technique[8,9,10,12] to reduce the size of a file and to get better compression. The lossless compression

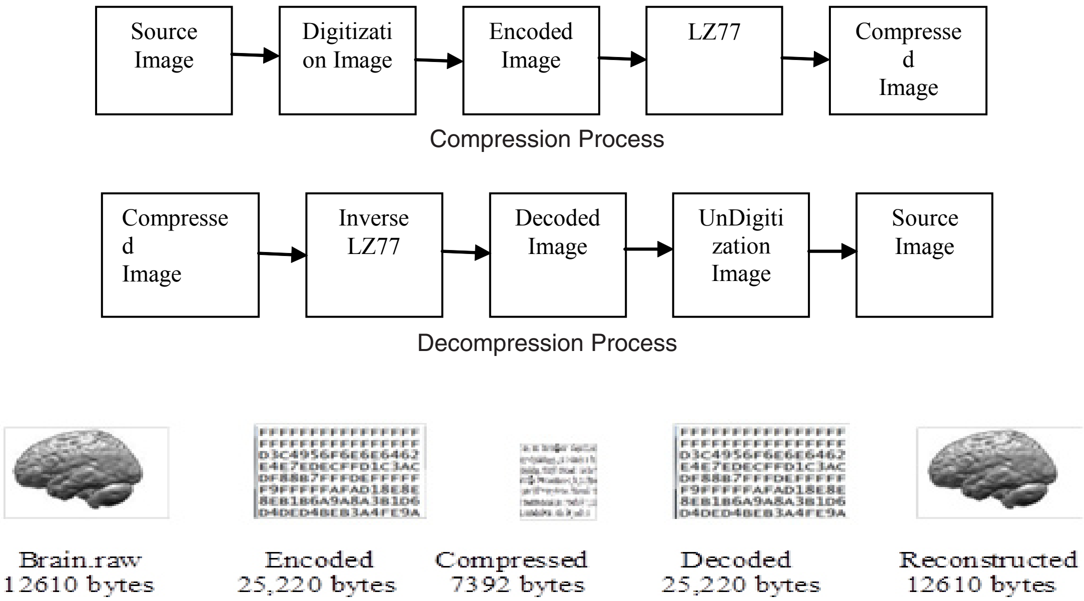
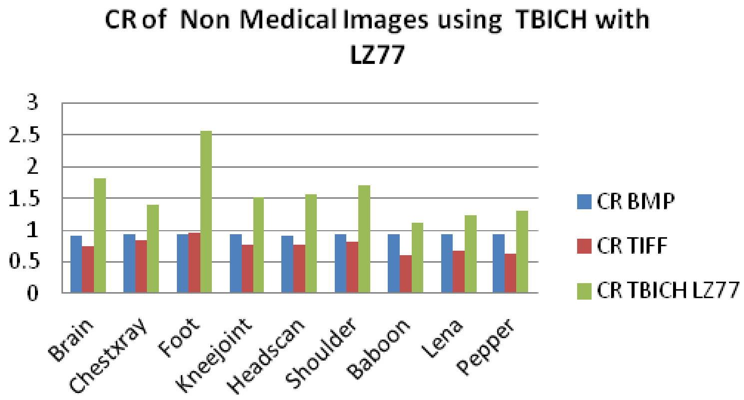


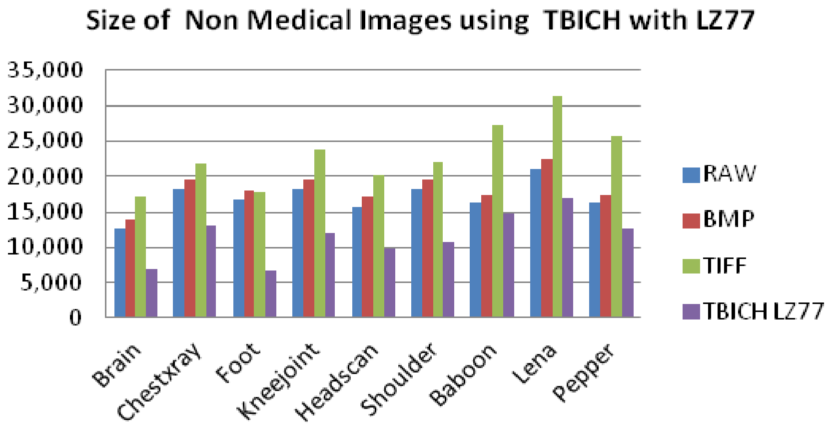
Fig: Flow of Text based image compression technique

Sizes and CR of Images using TBICH with LZ77

Images	RAW (bytes)	BMP (bytes)	CR BMP	TIFF (bytes)	CR TIFF	TBICH Encoded (bytes)	TBICH LZ77 (bytes)	CR TBICH LZ77
Brain	12,610	13,884	0.90824	17,068	0.738809	25,220	6,909	1.825156
Chestxray	18,225	19,440	0.9375	21,800	0.836009	36,450	13,033	1.398373
Foot	16,740	17,944	0.932902	17,756	0.94278	33,480	6,552	2.554945
Kneejoint	18,225	19,440	0.9375	23,852	0.764087	36,450	11,986	1.520524
Headscan	15,625	17,080	0.914813	20,260	0.771224	31,250	9,955	1.569563
Shoulder	18,225	19,440	0.9375	22,076	0.825557	36,450	10,736	1.69756
Baboon	16,384	17,464	0.938158	27,268	0.600851	32,768	14,716	1.113346
Lena	21,025	22,540	0.932786	31,244	0.672929	42,050	16,958	1.239828
Pepper	16,384	17,464	0.938158	25,852	0.633761	32,768	12,557	1.30477



Graph on CR of different coding technique for Non medical images and TBICH with LZ77



Graph on Sizes of different coding technique for Non medical images and TBICH with LZ77

technique LZ77[11,12,13,15] is applied on encoded file to reduce the size .The below figure shows the size of source image, encoded file and compressed file. It clearly shows that size of an image is reduced.

In this proposed technique, consider the sample raw image converting it to text formats and applying the data compression technique to the reduce size of encoded file. Final compressed file size is less than the original image. In decompression, compressed file is converted into encoded file and again encoded file converted into image. The below Figure shows proposed lossless compression technique.

Performance analysis

The proposed TBICH technique is applied on each sample medical and non medical images and compression rate calculated. The below table depict number of bytes required to store an image and compression rate of different coding techniques and proposed TBICH for medical and non medical images. Graph is drawn on the basis of these sizes of different compression techniques and on the basis of compression ratio. This table is clearly shows that TBICH technique is produces the better results than other techniques.

CONCLUSION

The proposed TBICH technique is purely lossless technique and showing better results. This technique suitable for medical and satellite images.

REFERENCES

1. http://en.wikipedia.org/wiki/Digitizing, http://graficaobscura.com/ matrix/ index.html
2. http://en.wikipedia.org/wiki/Hexadecimal, http://www.linfo.org/hexadecimal.html
3. "Influence of hexadecimal character and decimal character for power waveform data compression" by Dahai Zhang, Yanqui Bi, Jianguo Zhao, ISBN-978-1-4244-4934-7, IEEE. In Sustainable Power Generation and Supply, 2009 SUPERGEN '09, International conference.
4. http://en.wikipedia.org/wiki/LZ77_and_LZ78, http://wiki.tcl.tk/12368
5. "Data Compression Using Long Common Strings " by Jon Bentley Bell Labs, Hanover, New Hampshire
6. " A Comparative Study Of Text Compression Algorithms " International Journal of Wisdom Based Computing, Vol. 1 (3), December 2011, Senthil Shanmugasundaram, Robert Lourdusamy
7. "High Performance DEFLATE Compression on Intel® Architecture Processors"- November 2011, Vinodh Gopal, Jim Guilford, Wajdi Feghal i, Erdinc Ozturk, Gil Wolrich, IA Architects Intel Corporation
8. "The data compression program for Gzip" version 1.6 28 May 2013
9. International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 5– No.4, March 2013 – www.ijais.org 51 " Hardware Implementation of LZMA Data Compression Algorithm" by E.Jebamalar Leavline, D.Asir Antony Gnana Singh
10. International Journal of Computer Applications (0975 – 8887), **80**(4) (2013) "Semi-Adaptive Substitution Coder for Lossless Text Compression" Rexline S J, Robert L, Trujilla Lobo
11. Dictionary Based Pre-processing Methods in Text Compression" - A Survey Rexline S.J, Robert L *International Journal of Wisdom Based Computing,* (2); (2011)
12. "New Strategy of Lossy Text Compression", Al-Dubaee, India Integrated Intelligent Computing (ICIIC), First International Conference on 5-7 Aug. 2010 E-ISBN :978-0-7695-4152-5, Print ISBN:978-1-4244-7963-4, INSPEC Accession Number:11530473 Publisher:IEEE
13. "An enhanced scheme for lossless compression of short text for resource constrained devices" Published in: Computer and Information Technology (ICCIT), 2011 14th International Conference on 22-24 Dec. 2011, Print ISBN:978-1-61284-907-2,

INSPEC Accession Number:12592220, Publisher:IEEE , Islam, M.R. Ahsan Rajon.

14. "Text Compression and Encryption through Smart Devices for Mobile Communication" Pizzolante, R. Published in: Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2013 Seventh International Conference on 3-5 July 2013, Page(s):672 – 677 INSPEC Accession Number:13794400. Publisher: IEEE.
15. "Lossless Image Compression Technique Using Combination Methods Journal of Software Engineering and Applications, 2012, 5, 752-763.A. Alarabeyyat1, S. Al-Hashemi1, T. Khdour1, M. Hjouj Btoush1, S. Bani-Ahmad1, R. Al-Hashemi2
16. S Rigler, W Bishop and A Kennings. FPGA-Based Lossless Data Compression using Huffman and LZ77 Algorithms. Proceedings of Canadian Conference on Electrical and Computer Engineering, (2007).
17. Ashwini M. Deshpande, Mangesh S. Deshpande and Devendra N. Kayatanavar. FPGA Implementation of AES encryption and decryption. Proceedings of International Conference on Control, Automation, Communication and Energy Conservation (2009).
18. David Salomon. Data Compression: The Complete Reference. Second Edition, Springer New York, Inc. (2000).
19. Simon Haykin. Communication Systems. 4th Edition, John Wiley and Sons, (2001).