International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 12, December - 2017

A Comparative Study of Image Format Compression Deflate Algorithms

Pandi Barita Nauli Simangunsong
Amik Stiekom of North Sumatera
Medan, Indonesia

Abstract **: Image is very often used for personal important, groups and crowds with the help of handpone or dslr camera, there is often a problem with the image storage because of the size of diverse images, so need a place to store the image it takes memory to save that image, the more the image will be the greater the required memorytherefore a necessary in way to overcome these problems with compression techniques, deflate algorithm can solve the problem, the kind of image extension that is best to be compressed with deflate method is bmp extension (Bitmap).**

Keywords* : *Image Compression, Image Format, Deflate Algorithm.

I. INTRODUCTION

Image is an object that is very real or in accordance with the original, images that have great resolution must have great resolution too, the resolution is directly proportional to the size of the image, so the storage or memory needed will be greater, to solve this problem need to use techniques to overcome them. Many image file formats used such as jpg, gif and tiff format, bmp, and the size of each format are also different then it takes a compression technique to overcome the problem of the image.

Storage media really need to be used to store photographic images, more and more photographic images are stored in the storage media then the memory will be required greater. Therefore the techniques needed to overcome it, so that images that have large size can still be saved even if the storage media is limited. As for ways that can be used to overcome them, required storage of data on storage media, storing data in storage media needs to be applied by compressing data to produce smaller data. Image compression is used so that images can be utilized with the aim of reducing the redundancy of the data in the image so images can be stored efficiently.

Compression is a technique used to overcome the problem of excessive photographic photo size, so the previous image has a large image capacity to be used as a new image with a small size to store very large image data in memory, can be stored in small capacity. storage media, compression is an appropriate technique for reducing image size. The deflate algorithm is one of the best ways to solve the problem of too large a photographic image size.

The compressed deflate algorithm is derived from a series of blocks, based on the data entered into each of the blocks to be arranged in sequence. the block to be compressed will consist of combinations of lz77 algorithms and huffman algorithms [12].

II. LITERATURE REVIEW

A. *Image Compression*

Compression is a technique to reduce the size of the image, with the aim to save momori to save the image and transfer process also more smoothly therefore the bandwidth required is not too large. compression aims to reduce the size of the image without reducing the quality of the image. a small size gauge is widely used for medical images, to be stored in storage space[7]. According to the compression definition is the technique of minimizing the storage space required to be able to store data with an uncertain amount. it is used to save memory such as bandwidth, cost and time required to transmit data to other places. Lossless technique is required in medical field because lossless technique is very suitable, different from losy technique because the data when compression is missing [4].

B. *Lossless And Losy*

Compression techniques are always used there are two, namely lossless compression or losy compression is always used to reduce the size of the image. Lossless or lossy techniques will be used for compression of images that have a large size. Decompression process is required to restore the image of the original form [13].

Compression losy is a compression that is no longer similar to the original data. Losy compression of many important data is removed, so the data is no longer in accordance with the original data and a lot of information is not the same so that data can not be "perceived" by humans [9].

The lossless compression technique is data when compressed equally or in accordance with bit-for-fit which means data is similar to the original data. Lossles means "fixed data is appropriate", data that has been compressed very efficiently stored in a state already compressed because the data will still be able to survive in memory [9].

C. File Format

The image can be selected as desired, such as: Tiff, Jpeg, Rawfle, if the desired raw format, can also choose between 8-bit or 16-bit, the corresponding results from the camera, digital camera there are options to choose low quality to high quality, to a negative color, quality choices can affect the editing process, the process of printing to the process of processing the image [5].

D. *Deflate Algorithm*

Deflate algorithm is applied to each text block. Deflate algorithm is a compression algorithm that has lossless properties, deflate algorithm is kombinsi lz77 algorithm and huffman algorithm. The huffman tree will not rely on any previous blocks and the next block. The Lz77 algorithm has duplicate characters that exist in each previous block until new data up to 32 KB before [10].

Hash tables use duplicate strings for the search process. the number of characters has a length of 3 parts that will enter into the hash table. Calculation of hash table index is made as much as 3 bytes next. No hash table empty, all characters will be compared with input characters and the most matching data will be selected [12].

1. Algoritma Lempel Zip 77

Lz77 algorithm is a compression algorithm that uses a dictionary to compare incoming bytes, but the data is not genuine. But encoding is made only one, all the encoded data is created exactly the same [12]. Lz77 algorithm compression process data will take a long time which can cause the missing character, lz77 algorithm deficiency is [2]:

a. Algorithm takes a long time.

b. Lz77 algorithm takes longer time during compression so as to cause data there is damaged during compression process.

2. Algoritma *Huffman*

The arrangement of the tree from the merge from the initial node to the end of the root tree node. the huffman algorithm combines two tree nodes, from the lower frequencies to the uppermost frequencies. The number of frequencies of both nodes is obtained from the interior node [12]. The huffman algorithm uses a statistical encoding that can minimize the number of bits required for a series of symbols. The huffman algorithm has a purpose by making many variations of symbols. A code that has a short frequency will often be used whereas a long frequency code is rarely used, which means that code that has a long frequency will appear longer to the string [1]. The huffman algorithm is well translated therefore has rules to adhere to, meaning the first code is formed by frequency and can not be a prefix code in another code, so the code following the rule will form a binary tree, which is often heard with the term huffman tree [7].

III. Result And Discusion

The following sample images to be compressed using the deflate method according to the deflate algorithm rules:

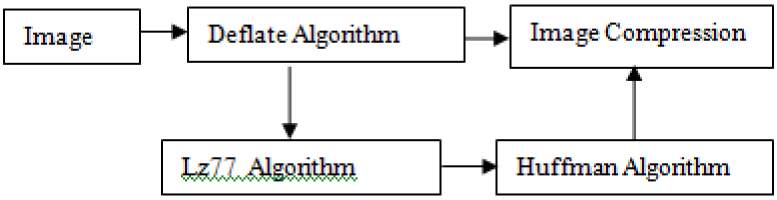
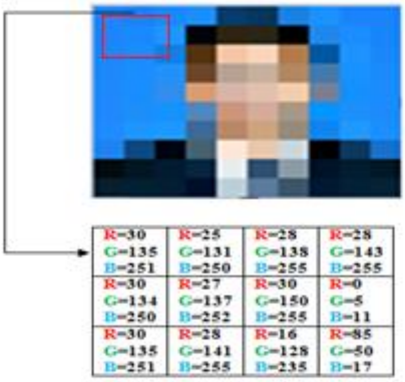


Figure 1 : Compression Proses Deflate Algorithm

The following sample image to be dikomresi using deflate algorithm is:



R=30
G=135
B=251
	R=25
	G=131
	B=250
		R=28
		G=138
		B=255
			R=28
			G=143
			B=255

R=30
G=134
B=250
	R=27
	G=137
	B=252
		R=30
		G=150
		B=255
			R=0
			G=5
			B=11

R=30
G=135
B=251
	R=28
	G=141
	B=255
		R=16
		G=128
		B=235
			R=85
			G=50
			B=17

Figure 2: Image Sample Comprassion

The principle of the deflate method is the combination of the lz77 algorithm and the huffman algorithm, the first compression of a photographic image with the lz77 algorithm and the lz77 algorithm output consisting of (Index, Length, Next Symbol).

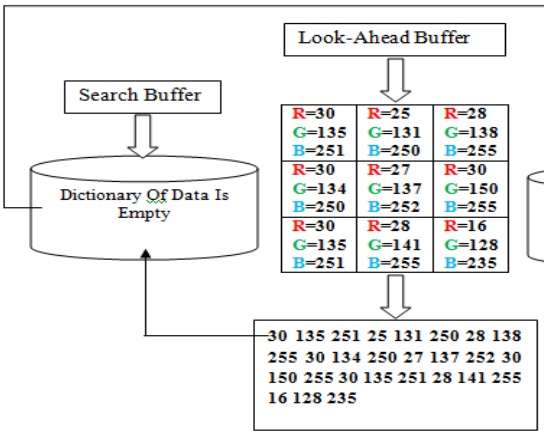


Figure 3: Lz77 Algorithm

In the picture above is known that the character 30 is not found in the search buffer then the index is 0 and the length is also 0 and the next character to be entered in the search buffer is character 30.

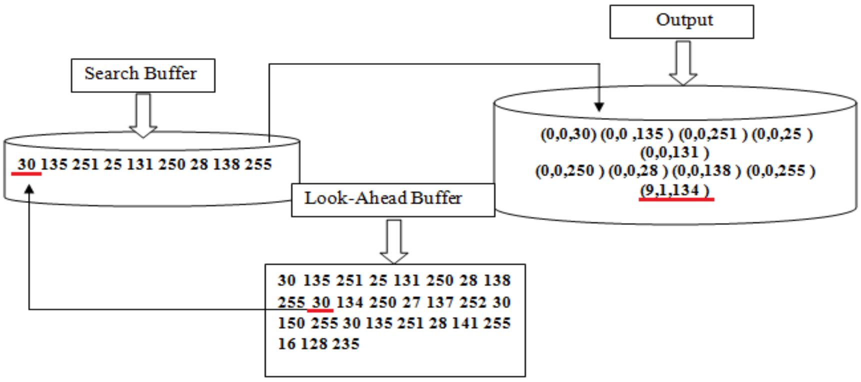


Figure 4: Compression Lz77 Algorithm

In the image above the next character to be input is the 30 character in the look-ahead buffer will be input to the search buffer because the 30 character is in the index to Nine and the same character is only one and the next symbol is 134 then the output writing is (9,1,134) so on until the process is complete.

Output Algoritma LZ77
(0,0,30) (0,0,135) (0,0,251) (0,0,25) (0,0,131) (0,0,250) (0,0,28) (0,0,138) (0,0,255) (9,1,134) (6,1,27) (0,0,137) (0,0,252) (6,1,150) (9,2,135) (0,0,251) (15,1,141) (6,1,16) (0,0,128) (0,0,235)

Image size after compressed using LZ77 algorithm is 60 bytes. The output of LZ77 will be compressed again using the huffman tree algorithm.

The next step of the huffman algorithm process is:

1. Sort the value of the output on the lz77 algorithm based on the frequency of occurrence of the smallest frequency to the largest frequency.

Table 1: Frecuency Of Pixel Value

Character	Frequency
0	28
1	5
6	3
9	2
135	2
251	2
2	1
15	1
16	1
25	1
27	1
28	1
30	1
128	1
131	1
134	1
137	1
138	1
141	1
150	1
235	1
250	1
252	1
255	1

2. Join the two frequency occurrences of the smallest frequency to the largest frequency that has been sorted.

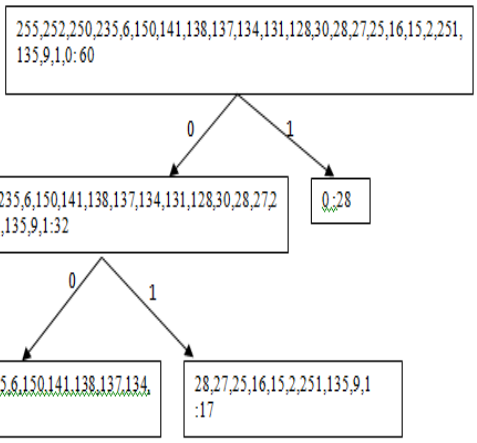


Figure 5: Huffman Compression Process

3. Repeat step number two until the remaining one binary tree.
4. Numbering on the binary tree, the left side of the tree is labeled 0 and the right side of the tree is labeled 1.
5. Browse the binary tree from root to leaf.

Table 2: Huffman Compression Results

Character	Amount	Bit Code	Image Size
0	28	0	1*28=28
1	5	1110	4*5=20
6	3	1000	4*3=12
9	2	10110	5*2=10
135	2	00110	5*2=10
251	2	11010	5*2=10
2	1	101010	6*1=6
15	1	001010	6*1=6
16	1	110010	6*1=6
25	1	010010	6*1=6
27	1	100010	6*1=6
28	1	000010	6*1=6
30	1	111100	6*1=6
128	1	011100	6*1=6
131	1	101100	6*1=6
134	1	001100	6*1=6
137	1	110100	6*1=6
138	1	010100	6*1=6
141	1	100100	6*1=6
150	1	000100	6*1=6
235	1	110000	6*1=6
250	1	010000	6*1=6
252	1	100000	6*1=6
255	1	000000	6*1=6

The result of image compression using deflate method is the size after compressed to 198 bits, the previous size is 216 bits then deflate method yield ratio of 8.4%, that means deflate method save memory of 8.4%. The result of comparison of deflate algorithm show that deflate algorithm efficient use against some file formats in the image.

Table 3: Comparative of Image Format

File Name	Before Compression / Kb	After Compression / Kb	Rasio (%)
One.bmp	23347.2	22323.2	4%
Two.bmp	1044.48	866	17%
Three.bmp	23347.2	22323.2	4%
Four.bmp	23347.2	22323.2	4%
Pandi smg.png	501	500	-0,2 %
Jellyfish.png	1607.68	1607	0%
Abc.tif	551936	550912	0,18%
Abcd.tif	690176	686080	0.59%
Chrysanthemum.gif	247	247	0%
Snapshot - 1.gif	49.6	50	-0.80%
Img_1291.jpg	1884.16	1884.16	0%
Img_1292.jpg	2539	2539	0%
Penguins.jpg	759	759	0%

The graph below shows the ratio of the compression deflate algorithm

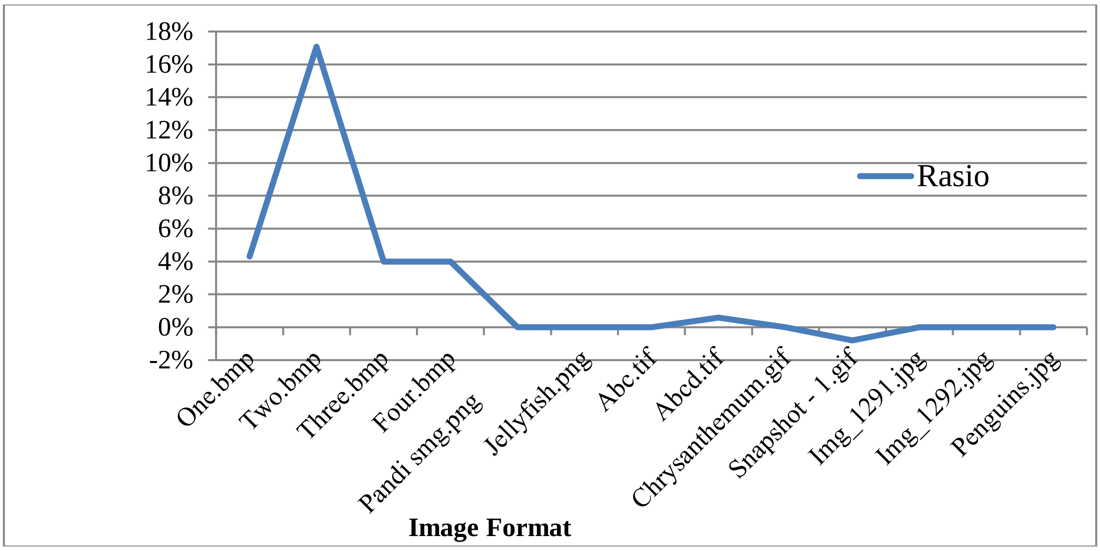


Figure 6: Chart showing compression deflate algorithm

X-axis → Format File
Y-axis → Compression Ratio

IV. CONCLUSIONS

Deflate algorithm has lossless deflate algorithm technique done with two stages, first stage compressed with LZ77 algorithm then compressed again with huffman algorithm, Image compression result image with extension * Bmp and * Tif much better to be compressed when compared to image file with extension * jpg, * gif, * png. When the compression process occurs the information from the image is not lost because deflate algorithm is lossless and for the process of returning the original image form with decompression technique.

V. REFERENCES

[1] A.A.Shaikh, P.P.Gadekar, 2015, Huffman Coding Technique for Image Compression Vol-IV, Issue-IV, April 2015.

[2] Aslam Khan, Sanjaya Mishra,2014,Image Compression using Growing Self Organizing Map Algorithm Vol.14, No.11, November 2014.

[3] Bora Park, Antonio Salvodi,Paolo Gubian 2008. Data Extraction From Damage Compressed File For Computer Forensic Purposes Vol.1, No.14, October 2008 pp.89-102.

[4] Christy Sumitha Vincant, Janet, 2016, An Enhanced N-Pattern Hybrid Technique for Medical Images in Telemedicine ,2016 p. 305-313.

[5] Dan Simon, 2004. Digital Photography Bible:Desktop Edition. Canada: WileyPublishing,Inc.

[6] Gonzalez R.C, Woods R.E.2008.Digital Image Processing:3rd Edition.Usa:Pearson Prentice Hall.

[7] Kavinder, Chopra Vinay, 2013, DICOM Image Compression using Huffman Coding Technique with Vector Quantization Vol.4, No.3, March 2013.

[8] Nilam P. Shaha, M. S. Kumbhar,2013. Image Compression Using Huffman Coding Vol 1, No.8, 2013 pp.119-126

[9] Mamta Sharma. 2010. Compression Using Huffman Coding Vol.10, No.5, May 2010 pp.133-141.

[10] M Sundaresan, E Devika. 2012, Efficient Integrated Coding for Compound Image Comprassion Vol.2, No.2, July-September 2012.

[11] M Sundaresan, E Devika. 2013, Lossy and Lossless Compression using various Algorithms Vol.65, No.20, March 2013.

[12] Savan Oswal, Singh Anjali, Kirthi Kumari, 2016. Deflate Compression Algorithm Vol.4, Issue 1, Januari 2016 pp. 430-436.

[13] S. Smyrna Grace, T.Nalini,A. Pravin Kumar, 2012, Secure And Compressed Secret Writing Using DES and DEFLATE Algorithm Vol.9 No.3, Issue 2, March 2012