# Analyzing the Bitmap Image File and Extracting Different Bit Patterns for Compressing the Image File using Distinct Colour Codes

Apurv K. Pandya
Research Scholar,
Department of Computer Science
Saurashtra University, Rajkot, Gujarat, India

C. K. Kumbharana, PhD
Professor, Head & Guide,
Department of Computer Science
Saurashtra University Rajkot, Gujarat, India

## ABSTRACT
Graphics image files can be stored in different file formats. One of the most general form is bitmap image format. Bitmap file format is an uncompressed form of image consisting of matrix of pixel constructing the image. BMP (24bit) represents each pixel in set of 24 bits comprising of RGB(Red, Green, Blue). Where BMP(32bit) represents one additional byte of alpha for transparency/opaque. Researcher identified redundancy of colors and made model to compress BMP(24bit) file. A model is developed to list all unique color of the image and store only index of these unique color instead of the whole 3 RGB color. The compression performance is based on number of unique colors and width-height of the image. In this paper researcher has made an effort to extract such bit patterns representing the pixel and then using such different bit patterns determines the total number of colors represented in image. Once the number of color is found it is possible to compress the file by grouping similar colors with exact pixel positions. Paper is prepared based on the research to analyze the image file and then compressing it. The model demonstrated here for 24bit file and at last concluded with tradeoff, improvements and limitation.

## Keywords
Image Processing, Bitmap format, Graphics file format, Digital Image Processing, Image Compression, Color code, RGB Color model, Bit Matrix.

## 1. INTRODUCTION
In the world of computer, digital image processing is a wide area of research in which an image is analyzed and processed to achieve the desired goal like extracting portion, transformation, mirror image, rotating, scaling and storing. Storing an image file requires different techniques for different formats. The simplest way to store the image file in form bitmap file. The file format is consisted of m X n matrix of bits having 0 and 1 in uncompressed form. One drawback of such file is the storage requirements. As it is the original format with all detail of an image stored for each pixel of the image it requires more space to store. For example as the experimented images used for this paper are of 400 X 300 of pixels with each pixel is represented through 24 bits of information. Pixels are processed in sequential manner and extracted the bit patterns for each of the pixels. Once the bit forms are extracted it can be compressed to optimally utilize the storage space. There are number of image file formats and various ways to compress an image file. In this paper 24bit bitmap image file format is used to compress the file using color coding scheme.

Image conversion and compression of a file has two major aspects, lossless and lossy compression format with no loss and some undistinguishable loss in original image respectively. In lossless form an image can be decompressed in its original form but in lossy format it is not possible due to loss of information to achieve the desired compression rate. [1][2][3]

## 2. GRAPHICS IMAGE FILE FORMATS
Various graphics file formats and compression algorithms are available. Each of them can be used based on the needs of an application and the availability of resources to use them. The most popular forms of graphics file formats and compressions algorithms are, BMP, GIF, JPEG, PNG, PSD, TIFF. Each format is described here briefly[4][5].

### 2.1 BMP
BMP format stands for Bitmap file. It stores the pixel values in form of bit matrix. It is an incomprssed file format.

### 2.2 GIF
GIF stands for Graphic Interchange Format. It uses 256 colors occuping only one byte for pixel and suitable for small image files like logos. It has also the capablities of having animation. In this format 24-bit of color depth is possible. GIF is a lossless image file format but not suitable for photopraphs and color gradient images.

### 2.3 JPEG
JPEG stands for Joint Photographic Expert Group format the compresses the image with 24 bits of colour depth. It is lossy compression but loss of information can be ignored compared to comression ratio which further can be adjusted. JPEG format is one of the widely form of image compression.

### 2.4 PNG
PNG stands for Portable Network Graphics which is an open source compression technique. It supports 8 bit pallete colors and 24 bit true color images. It is designd for online image viewing applications used with internet .

### 2.5 PSD
PSD stands for Photoshop Document designed by Adobe systems. PSD format is one of the oldest image format.

### 2.6 TIFF
TIFF stands for Tagged Image File Format. It is useful for raster graphics images which are scanned and processed. It is a lossless technique.

Many other formats and compression exist but only few are introduced.

## 3. BITMAP IMAGE FILE FORMAT

Bitmap image file format is stored as first row of image pixel data is stored in BMP file upside down to meet windows coordinate system. A BMP file has following four parts.[6][7]

(1) File header

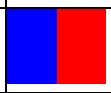
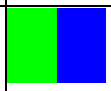
(2) Bitmap Header

(3) Color table

(4) Image data

File header occupies 14 bytes. These 14 bytes are divided into five segments containing the details of file type, file size, reserved bytes and pointer to image data of BMP file. Bitmap Header occupies 40 bytes containing detailed information of bitmap image. Color table contains the color information of image for bitmap indexing. Last part stores the image data row by row representing pixel values in binary form

**Table 1. BMP Color and size**

| Sr | File Name, Width X Height, Size | Number of Colors | Color |
|---|---|---|---|
| 1 |  blue.bmp, 400X300, 360054 bytes | 1 | blue(255) |
| 2 |  bluered.bmp 400X300, 360054 bytes | 2 | blue(255)<br>red(255) |
| 3 |  blueredshaded.bmp 400X300, 360054 bytes | 1 | red(128), blue(128) |
| 4 |  green.bmp 400X300, 360054bytes | 1 | green(255) |
| 5 |  greenblue.bmp ,400X300, 360054 bytes | 2 | green(255)<br>blue(255) |
| 6 |  greenblueshaded.bmp 400X300, 360054 bytes | 1 | green(128), blue(128) |
| 7 |  red.bmp 400X300, 360054 bytes | 1 | red(255) |
| 8 |  redgreen.bmp 400X300, 360054 bytes | 2 | red(255)<br>green(255) |
| 9 |  redgreenblue.bmp 400X300, 360054 bytes | 3 | red(255)<br>green(255)<br>blue(255) |
| 10 |  redgreenblueshaded.bmp 400X300, 360054 bytes | 1 | red(128), green(128), blue(128) |
| 11 |  redgreenshaded.bmp 400X300, 360054 bytes | 1 | red(128), green(128) |

For a monochrome image a bit value may be either 0 or 1 representing black and white. For gray scale image a byte is used to obtain different shades of black and white image. To store a color image it uses RGB color model as every color is constructed through combination of basic three colors RED, GREEN and BLUE. Each color value of Red, Green and Blue is stored in separate byte [8]. So total 2 ^ 24 combination resulting 16777216 different colors can be stored. One more byte is used to store transparency information and is known as alpha channel. If it's all bits are 0 then it represents entirely transparent and if it's all bits are 1 then it represents entirely solid or opaque. In general its values vary between these two lower and upper bound values. Bitmap files are uncompressed files. Researchers have taken width:400 pixels, height:300 pixels and color model: 24bits.[8]

So size of BMP images$= 400 \times 300 \times 3 + 54$

$= 360054$ Bytes
$= 351$ KB.

There are 11 images created using red, green, blue and combination of these colors to study bit pattern of bit map picture, which are shown in the table-1. Here number shown in bracket with color name is the intensity of the color in the range from 0 to 255. Bit pattern of these images are shown in table 2.

**Table 2. BMP Color bit pattern**

| File Name | Color | RED | GREEN | BLUE |
|---|---|---|---|---|
| blue.bmp | blue(255) | 00000000 | 00000000 | 11111111 |
| Blue-red.bmp | blue(255) | 00000000 | 00000000 | 11111111 |
| | red(255) | 11111111 | 00000000 | 00000000 |
| Blue-red-shaded.bmp | red(128), blue(128) | 10000000 | 00000000 | 10000000 |
| green.bmp | green(255) | 00000000 | 11111111 | 00000000 |
| Green-blue.bmp | green(255) | 00000000 | 11111111 | 00000000 |
| | blue(255) | 00000000 | 00000000 | 11111111 |
| Green-Blue-shaded.bmp | green(128), blue(128) | 00000000 | 10000000 | 10000000 |
| red.bmp | red(255) | 11111111 | 00000000 | 00000000 |
| Red-green.bmp | red(255) | 11111111 | 00000000 | 00000000 |
| | green(255) | 00000000 | 11111111 | 00000000 |
| Red-green-blue.bmp | red(255) | 11111111 | 00000000 | 00000000 |
| | green(255) | 00000000 | 11111111 | 00000000 |
| | blue(255) | 00000000 | 00000000 | 11111111 |
| Red-green-blue-shaded.bmp | red(128), green(128), blue(128) | 10000000 | 10000000 | 10000000 |
| Red-green-shaded.bmp | red(128), green(128) | 10000000 | 10000000 | 00000000 |

Analysis of color bit pattern of above referred 11 files shows that size of all file remains same irrespective of number of colors used i.e. one, two or three.

## 4. A MODEL FOR BMP IMAGE COMPRESSION

Researcher has derived a model to compress the details of bitmap file. The model analyzed the given file and extract bit

information from the file which is used to determine the color patterns used for compression.

Description for Figure 1 : 24-bit BMP Image compression model:

Each process is of above model is described in detail. It can be implemented in suitable programming language.

1. First an Image file is read using file reading facilty of programing language construct.
2. Second step checks color of each pixel one by one and adds into list, if it does not already exist in the list. Thus in the second step list of unique colors is prepared.
3. Create matrix of same width X height of original image. Replace each 3 byte pixel with unique color index of1 or 2 byte.
4. Store width, height, list of unique colors and pixel matrix into file.

Following Figure 1 depicts the model to compress 24bit BMP.

The model stores width, height, unique color count in addition to image matrix and color index. Data and file structure for the compressed image is given below.

**Table 3 Compressed Image data and file structure of AKP**

| **Compressed Image file header information** | **Width (W)** | **Height (H)** | **Unique color count (C)** | **Index (I)** | **Index Data (IDX)** | **Image Data (IMG)** |
|---|---|---|---|---|---|---|
| Size (9 Bytes) | 2 Bytes | 2 Bytes | 4 Bytes | 1 Byte | C*3 Bytes | W*H*I Bytes |
| possible value | 0 to 65535 | 0 to 65535 | 0 to 16777216 | 1, 2 or 3 | C Colors, each of 3 Bytes (RGB) | Index value for each pixel |

The proposed compressed file (AKP) has header information of 9 (NINE) bytes, comprising of width, height, unique color count and index.

Width(2 Bytes): the width of image is stored using two bytes which can be from 0 to 65535.

Height(2 Bytes): the height of image is stored using two bytes which can be from 0 to 65535.

Unique color count( 4 Bytes): the unique color count holds number of unique colors of the image, which may range from 0 to 16777216.

Index(1 Byte) : the index holds 1, 2 or 3. If unique color count is 256 or less, index of color can be stored into one byte only. So index will hold ONE. If unique color count is 65536 or less, index of color requires TWO bytes. So index will hold TWO. If unique color count is 16777216 or less, index of color requires THREE bytes. So index will hold THREE.

The index data (IDX) is list of unique colors each of 3 bytes (RGB). Image data will hold index of color for each pixel. This image data array will be of size width X height X index.

## 5. OUTCOME AND EXPERIMENTING WITH DIFFERENT BITMAP IMAGES

Experiment is done with many images including handmade shown in table-1. Some metadata like width, height, unique color count and Index size are required to be stored. This header information takes 09 bytes.

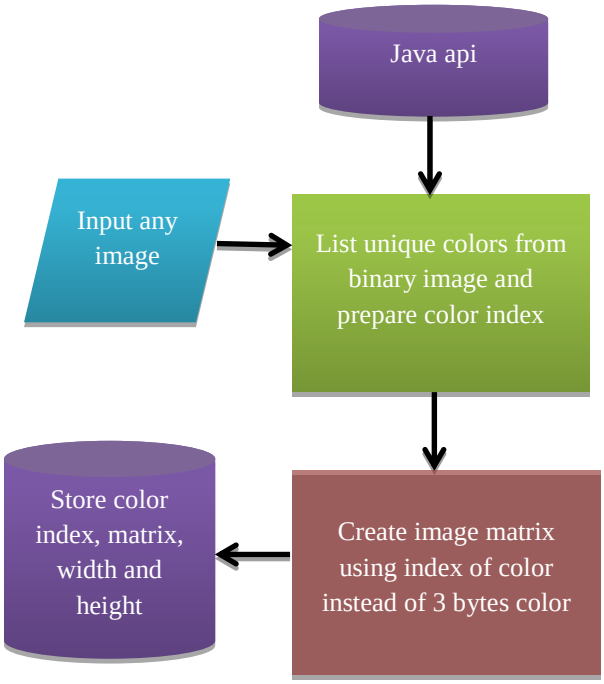



**Fig 1 : 24-bit BMP Image compression model**

**Table 4 Compressed Image data and file structure of AKP**

| **Number of Colors** | **Number of Bytes required for storing color index into matrix** | **Number of bytes of index (Number of colors * 3 [RGB])** | **Number of bytes for Image Matrix 400 x 300 x B** | **Compressed file Header size (width, height, index size)** | **Total bytes required for Compressed file (C+D+E)** | **BMP Size (Bytes) for Matrix 400 X 300** |
|---|---|---|---|---|---|---|
| **A** | **B** | **C** | **D** | **E** | **F** | **G** |
| 1 | 1 | 3 | 120000 | 9 | 120012 | 360054 |
| 2 | 1 | 6 | 120000 | 9 | 120015 | 360054 |
| 256 | 1 | 768 | 120000 | 9 | 120777 | 360054 |
| 257 | 2 | 771 | 240000 | 9 | 240780 | 360054 |
| 40015 | 2 | 120045 | 240000 | 9 | 360054 | 360054 |
| 40016 | 2 | 120048 | 240000 | 9 | 360057 | 360054 |
| 65536 | 2 | 196608 | 240000 | 9 | 436617 | 360054 |
| 65537 | 3 | 196611 | 360000 | 9 | 556620 | 360054 |
| 16777215 | 3 | 50331645 | 360000 | 9 | 50691654 | 360054 |
| 16777216 | 3 | 50331648 | 360000 | 9 | 50691657 | 360054 |

## 6. CONCLUSION

The compression model makes file size around 33% where number of unique colors are <=256 which occupies one byte for color index. The compression model compresses file sizes, till numbers of colors are < 40015. The compression model reaches at break-even point when number of colors reaches to 40015. If compression is attempted for file having more than 40015 colors, then it will increase the file size instead of compression. The scope of future work is to test the model for images with more variety of colors and width-height for better implementation.

## 7. REFERENCES

[1] Data Compression - Lossless and Lossy Techniques, Ruchi Gupta, Mukesh Kumar, Rohit Bathla IJAIEM July 2016 ISSN 23194847

[2] A survey on Lossless and lossy Data Compression Methods.- P. Kavitha – International Journal Of Computer Science and Engineering Technology IJCSET Volume 7 No, 3 March 2016

[3] A review of Image Compression Techniques – Rajandeep Kaur and Pooja Choudhary International Journal of Computer Applications (IJCA) VOL.142 No.1, May 2016

[4] A New Algorithm for Data Compression Optimization – I Made Agus Dwi Suirjaya International Journal of Advance Computer Science and Applications (IJACSA)iVol.3, No.8, 2012

[5] Designing and developing a model for converting image formats using Java API for comparative study of different image formats - International Journal of Scientific and Research Publications, Volume 4, Issue 7, July 2014

[6] Comparison of different Image compression formats – ECE 533 Project Report Paula Aguilera

[7] BMP to JPEG – The Conversion Process – Anjali Anand, Dr.Himanshu Aggarwal Journal of Global Research in Computer Science Volume 2 No, 6 June 2011

[8] Lossless Digital Compression Method For Bitmap Images – Dr. T. Meyyappan, SM. Thamarai, N.M. Jeya Nachiaban - The International Journal of Multimedia and Its Applications – (IJMA) Volume 3, No.4, November 2011.