INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

Department of Electronics & Electrical Communication Engineering
M.Tech. First Year

Vision and Intelligent Systems
(VIS)
EC69211– Image Processing Laboratory

Experiment No.1

BMP File Format

Submitted by

Suraj Kumar(22EC65R14)

Ayush Jangid(22EC65R24)

Contents

Introduction

This experiment aims to read a Bitmap image file, for manipulating the colour channel of the corn.bmp file, and modifying the R or G or B any one other channel to zero, and then write it to the disc as an image. This experiment was carried out with just the use of basic python constructs like read and write function to better understand how image files are generated, modified, and acted upon.

The BMP file format is a raster graphics image file format used to store bitmap digital pictures. It is also known as a bitmap image file, device independent bitmap (DIB) file format, or simply a bitmap. A single bit or a set of bits defines each pixel in a bitmap picture. As a result, it's known as a bitmap or a map of bits and pixels. Bitmap pictures are ideal for storage and image processing because they can store raw, uncompressed data. The BMP file format is capable of storing two-dimensional digital images both monochrome and colour, in various colour depths, and optionally with data compression, alpha channels, and colour profiles. Microsoft has defined a particular representation of colour bitmaps of different colour depths, as an aid to exchanging bitmaps between devices and applications with a variety of internal representations. They called these device-independent bitmaps or DIBs, and the file format for them is called DIB file format or BMP image file format.

BitMap File Format:

A BMP file format contains different sections that contain information about header, color pallet, and actual pixel data.

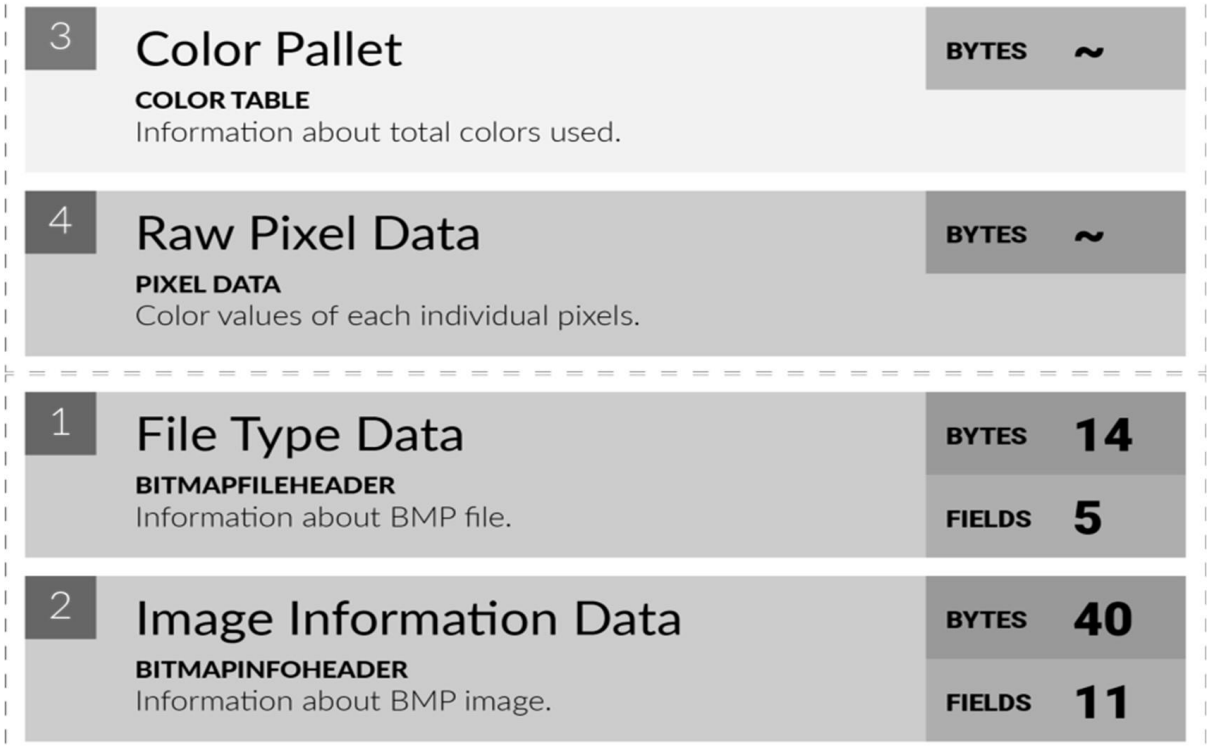

Figure 1: Sections of Bitmap File Format

Bitmap FileHeader:

This block is a BMP Header labeled as BITMAPFILEHEADER (the name comes from C++ struct in Windows OS). This is the starting point of the BMP file and has a 14 bytes width. This header contains a total of 5 fields of variable byte width. These are mentioned in the below table.

Bitmap File Header			
Offset (hex)	Offset (dec)	Size (bytes)	Purpose
00	0	2	The header field used to identify the BMP and DIB file is *0x42 0x4D* in hexadecimal, same as BM in ASCII. The following entries are possible: • **BM** Windows 3.1x, 95, NT, ... etc. • **BA** OS/2 struct bitmap array • **CI** OS/2 struct color icon • **CP** OS/2 const color pointer • **IC** OS/2 struct icon • **PT** OS/2 pointer
02	2	4	The size of the BMP file in bytes
06	6	2	Reserved; actual value depends on the application that creates the image
08	8	2	Reserved; actual value depends on the application that creates the image
0A	10	4	The offset, i.e. starting address, of the byte where the bitmap image data (pixel array) can be found.

Table 1: Description of Bitmap file header contents

Bitmap InfoHeader:

This header is 40-bytes wide and contains a total of 11 fields of variable byte widths. These are mentioned in the below table.

Bitmap Information Header Windows BITMAPINFOHEADER			
Offset (hex)	Offset (dec)	Size (bytes)	Purpose
0E	14	4	the size of this header (40 bytes)
12	18	4	the bitmap width in pixels (signed integer)
16	22	4	the bitmap height in pixels (signed integer)
1A	26	2	the number of color planes (must be 1)
1C	28	2	the number of bits per pixel, which is the color depth of the image. Typical values are 1, 4, 8, 16, 24 and 32.
1E	30	4	the compression method being used. See the next table for a list of possible values
22	34	4	the image size. This is the size of the raw bitmap data; a dummy 0 can be given for BI_ RGB bitmaps.
26	38	4	the horizontal resolution of the image. (pixel per meter, signed integer)
2A	42	4	the vertical resolution of the image. (pixel per meter, signed integer)
2E	46	4	the number of colors in the color palette, or 0 to default to 2n
32	50	4	the number of important colors used, or 0 when every color is important; generally ignored

Table 2: Description of Bitmap info header contents

Color Pallet:

This block contains the list of colors to be used by a pixel. This is an indexed table with the index starting from 0. However, this block is mandatory when BitsPerPixel is less than or equal to 8, hence this block is semi-optional. When the BitsPerPixel is 16, 24, or 32, the color value of a pixel is calculated from the combination of individual Blue, Green, and Red values defined by the pixel.

Raw Pixel Data:

This block contains binary numbers dedicated to representing the unique color values of each pixel. Depending on the bpp of the BMP image, a byte can contain color values of multiple pixels or multiple bytes can be used to represent the color value of a single pixel.

Algorithm

1. We take the name of the input file that is to be read. If the file is not in bmp format then print error message. i.e. Error!!! The file is not a BMP file
2. We then read the bitmap header data that is present at the starting of the BMP file and display the data of the header.
3. Depending on the information in the header we will allocate the size to the array and then loading the pixel data into the pixel array.
4. We get to know the number of bits per pixel from the Header Structure.
5. Since it is an 8-bit image, we allocate 256 x 4 size to the extra variable to store the color table data.
6. Then we call the write function and while writing the data from extra we make one of the channels say Blue = 0 and store the intensities of other channels as it is.
7. Finally, we write the data in the final output image “Corn_Blue_0”.

Fig 2: Input corn Image

Fig3:Output image(blue = 0)

Figure 4: Representing Green=0 and Red= 0 respectively

Discussion

1. While reading the 24-bit coloured image the offset value is 54 but when the 8-bit image is read its offset value is 1078. In an 8-bit coloured image, the colour table consists of the data which has size 256 x 4 = 1024. Hence offset value will be (54 + 1024) i.e. 1078.
2. BMP file format is well defined, the file format consists of a file header, information header, and pixel array. BMP file header gives important information such as height, width, size, offset, and the number of bits per pixel.
3.The size of the images can be calculated as:
Size= Header Size + Channels*Height*Width
This holds true in the case of RGB i.e. 54+3*256*256 = 192 KB which is almost equal to the size of the image, and in the case of output grayscale image, the calculated size is 1078 + 256*256 = 65KB which is almost equal to the size of the output image.
5. To set the intensity zero of any of the color in the corn image we have assigned value 0 to a particular column of a color table and increased our pointer by 4 to reach the next element of the same column.

References:

- Wikipedia page on Bitmap file format

- https://en.wikipedia.org/wiki/BMP_file_format