Windows Bitmap Library Documentation

Release 0.1.1

Jakub Przywóski

September 06, 2015

Contents

CHAPTER 1

Contents:

1.1 Introduction

This is a simple library for importing and manipulating windows bitmap files.

Currently only 24 bit uncompressed bitmaps are supported.

I developed this library for the purpose of teaching myself some image processing methods. So, the API will most likely change in the future as I add new stuff.

There is a lot of space for code optimiztion - most of the algorithms are trivially parallel - excellent use case for things like pthreads, openMP, CUDA or x86 vector instructions.

1.2 Quickstart

These sample programs illustrate how to access and manipulate pixel data in the bitmap:

1.2.1 Example 1

Brightness adjustment:

```c
#include <libwinbmp.c>
#include <stdio.h>
#define STEP 64

int main(void)
{
    bmp_t *bmp;
    bmp = bmp_load("/home/jakub/bmp-parser/lena.bmp");
    unsigned int row_size = bmp->info.width * 3;
    unsigned int x;
    unsigned int y;
    int d;

    for (y = 0; y < bmp->info.height; y++) {
        for (x = 0; x < row_size; x++) {
            d = (int)bmp->data[y][x] + STEP;
            if (d > 255) {
                d = 255;
            } else if (d < 0) {
                d = 0;
            }
            bmp->data[y][x] = (unsigned char)d;
        }
    }

    bmp_write(bmp,"/home/jakub/bmp-parser/frob.bmp");
    bmp_destroy(bmp);
    return 0;
}
```

1.2.2 Example 2

Averaging two images:

```c
#include <libwinbmp.c>
#include <stdio.h>


int main(void)
{
    bmp_t *bmp;
    bmp_t *other;
    bmp = bmp_load("/home/jakub/bmp-parser/lena.bmp");
    other = bmp_load("/home/jakub/bmp-parser/beach.bmp");

    bmp_average(bmp, other);

    bmp_write(bmp,"/home/jakub/bmp-parser/frob.bmp");
    bmp_destroy(bmp);
    bmp_destroy(other);
    return 0;
}
```

1.3 libwinbmp.h

1.3.1 Structures

'bmp_file_header_t'_ Bitmap file info.

'bmp_bitmap_info_header_t'_ Bitmap data info.

'bmp_t'_ Bitmap structure.

1.3.2 Utility Functions

'bmp_t *bmp_load(const char *path)'_ Loads bitmap file from the path into bmp_t structure.

'int bmp_write(bmp_t *bmp, const char *path)'_ Writes in-memory bitmap to a file.

'void bmp_destroy(bmp_t *bmp)'_ Deallocates memory taken up by the bitmap.

'unsigned int get_row_size(bmp_t *bmp)'_ Calculates row size including 4-byte alignment padding.

'unsigned int get_pixel_array_size(bmp_t *bmp)'_ Calculates pixel array size including 4-byte alignment padding.

1.3.3 Image Functions

Just a bunch of simple functions.

Histogram

'bmp_t *bmp_brightness(bmp_t *bmp, int step)'_ Adjusts the brightness of the image.

'bmp_t *bmp_invert(bmp_t *bmp)'_ Inverts the color values.

'bmp_t *bmp_grayscale(bmp_t *bmp)'_ Converts the image into grayscale.

'bmp_t *bmp_remove_channel(bmp_t *bmp, const char channel)'_ Removes selected rgb channel.

‘bmp_t *bmp_swap_channel(bmp_t *bmp, const char channel, const char other)‘_ Swaps two channels.

Image Arithmetic

‘bmp_t *bmp_add(bmp_t *bmp, const bmp_t *other)‘_ Adds two bitmaps.

‘bmp_t *bmp_subtract(bmp_t *bmp, const bmp_t *other)‘_ Subtracts two bitmaps.

‘bmp_t *bmp_difference(bmp_t *bmp, const bmp_t *other)‘_ Subtracts two bitmaps (absolute pixel distance is returned).

‘bmp_t *bmp_multiply(bmp_t *bmp, const bmp_t *other)‘_ Multiplies two bitmaps.

‘bmp_t *bmp_average(bmp_t *bmp, const bmp_t *other)‘_ Returns minimum of two pixels.

‘bmp_t *bmp_min(bmp_t *bmp, const bmp_t *other)‘_ Returns maximum of two pixels.

Convolution Filters

‘bmp_t *bmp_blur(bmp_t *bmp)‘_ Blurs the bitmap.

‘bmp_t *bmp_edges(bmp_t *bmp)‘_ Detects the edges.

‘bmp_t *bmp_sharpen(bmp_t *bmp)‘_ Sharpens the image.

‘bmp_t *bmp_emboss(bmp_t *bmp)‘_ Creates emboss effect.

‘bmp_t *bmp_mean(bmp_t *bmp)‘_ Mean blur filter.

Drawing

‘unsigned char *bmp_get_pixel(bmp_t *bmp, const unsigned int x, const unsigned int y)‘_ Returns the blue-green-red pixel values at the specified point.

‘void bmp_set_pixel(bmp_t *bmp, const unsigned int x, const unsigned int y, const unsigned int rgb)‘_ Sets the pixel at the specified point.

‘bmp_t *bmp_line(bmp_t *bmp, const int x0, const int y0, const int x1, const int y1, const int rgb)‘_ Draws a line.

CHAPTER 2

Indices and tables

- genindex
- modindex
- search