

Przetwarzanie i Kompresja Obrazów. Wstęp

Aleksander Denisiuk (denisjuk@pja.edu.pl)
Polsko-Japońska Akademia Technik Komputerowych
Wydział Informatyki w Gdańsku
ul. Brzezi 55, 80-045 Gdańsk

27 lutego 2016

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Najnowsza wersja tego dokumentu dostępna jest pod adresem
<http://users.pja.edu.pl/~denisjuk/>

Wprowadzenie

Wiadomości
wstępne

Biblioteki

Libpng

Libjpeg

Case study:
Wizualizacja

Wprowadzenie

Wprowadzenie

Wiadomości
wstępne

Biblioteki

Libpng

Libjpeg

Case study:
Wizualizacja

■ Modele obrazów graficznych

- ☐ grafika wektorowa
- ☐ grafika rastrowa

■ Modele barw

- ☐ RGB
- ☐ CMY
- ☐ YUV
- ☐ YCrBb
- ☐ CIE-*

■ Formaty plików graficznych

- ☐ PNG
- ☐ JPEG

Wprowadzenie

Wiadomości
wstępne

Biblioteki

Libpng

Libjpeg

Case study:
Wizualizacja

- **libpng**
- **jpeglib**
- **ImageMagick**
- **Opt**
- Paczki Debiana:
 - ☐ **libpng-dev**
 - ☐ **libjpeg-dev**
 - ☐ **imagemagick**
 - ☐ **opt**

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

Libpng

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

- Jest oficjalną legalną **wzorcową** biblioteką do obsługi PNG
- Ma zaimplementowane prawie wszystkie opcje formatu PNG
- Zgodna z ANSI C (C89)
- Istnieje od roku 1995
 - najnowsza stabilna wersja 1.6.21
 - 16 stycznia 2016
- Używa zlib w wersji 1.0.4 i wzwyż (zalecana jest wersja 1.2.5)
- Wolna licencja **libpng license**
 - dostępna w pliku **libpng.h**
 - certyfikowana przez OSI
 - można wyświetlić w kodzie

```
printf("%s", png_get_copyright(NULL));
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

- W kodzie (C, C++):
`#include <png.h>`
- W opcjach konsolidatora (linkera):
`-lpng`

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

■ png_struct

- ☐ struktura wewnętrzna
- ☐ zawsze jest przekazywana jako pierwszy argument do wszystkich funkcji `libpng`

■ png_info

- ☐ zawiera informacje o pliku PNG
- nie powinno się mieć (od wersji 1.5.0 nie ma) bezpośredniego dostępu do tych struktur
 - ☐ koresztać z licznych funkcji `png_get_xxx()` oraz `png_set_xxx()`

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

■ Otwieranie pliku

```
FILE *fp = fopen(file_name, "rb");  
if (!fp) return (ERROR);
```

■ Sprawdzanie nagłówka PNG

```
png_byte header[8];  
png_size_t number = 8;  
fread(header, 1, number, fp);  
is_png = !png_sig_cmp(header, 0, number);  
if (!is_png) return (NOT_PNG);
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

■ png_struct

```
png_structp png_ptr = png_create_read_struct  
    (PNG_LIBPNG_VER_STRING,  
     (png_voidp)user_error_ptr,  
     user_error_fn,  
     user_warning_fn);
```

```
if (!png_ptr) return (ERROR);
```

■ możliwe jest wykorzystanie domyślnych callbacków

```
png_structp png_ptr = png_create_read_struct  
    (PNG_LIBPNG_VER_STRING,  
     NULL,  
     NULL,  
     NULL);  
if (!png_ptr) return (ERROR);
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

■ png_info

```
png_infop info_ptr
    = png_create_info_struct(png_ptr);
if (!info_ptr){
    png_destroy_read_struct(&png_ptr,
        (png_infopp)NULL, (png_infopp)NULL);
    return (ERROR);
}
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
if(setjmp(png_jmpbuf(png_ptr))){  
    png_destroy_read_struct(&png_ptr,  
                           &info_ptr, NULL);  
    fclose(fp);  
    return (ERROR);  
}
```

- Trzecim argumentem, zamiast `NULL`, może być `&end_info`

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_init_io(png_ptr, fp);  
png_set_sig_bytes(png_ptr, number);  
png_read_info(png_ptr, info_ptr);
```

Otrzymanie parametrów pliku

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_uint_32 width, height;  
int color_type, bit_depth;  
png_uint_32 rowbytes;  
png_get_IHDR(png_ptr, info_ptr,  
             &width, &height,  
             &bit_depth, &color_type,  
             &interlace_type,  
             &compression_type, &filter_method);  
rowbytes=png_get_rowbytes(png_ptr,info_ptr);
```

- Trzy ostatnie argumenty mogą być **NULL**
- **width, height** — szerokość i wysokość obrazu w pikselach (do 2^{31})
- Są również funkcje do pobrania pojedynczych wartości

png_get_IHDR: głębia i paleta kolorów

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

- `color_type`, `bit_depth` — typ palety kolorów i odpowiednia głębia kolorów dla każdego kanału
 - ☐ `PNG_COLOR_TYPE_GRAY` — (bit depth: 1, 2, 4, 8, 16)
 - ☐ `PNG_COLOR_TYPE_GRAY_ALPHA` — (bit depth: 8, 16)
 - ☐ `PNG_COLOR_TYPE_PALETTE` — (bit depth: 1, 2, 4, 8)
 - ☐ `PNG_COLOR_TYPE_RGB` — (bit depth: 8, 16)
 - ☐ `PNG_COLOR_TYPE_RGB_ALPHA` — (bit depth: 8, 16)
- Dla innej głębii kolorów:
 - ☐ `PNG_COLOR_MASK_COLOR`
 - ☐ `PNG_COLOR_MASK_ALPHA`
 - ☐ `PNG_COLOR_MASK_PALETTE`

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

■ `interlace_type`

- ☐ `PNG_INTERLACE_NONE`
- ☐ `PNG_INTERLACE_ADAM7`

■ `compression_type`

- ☐ `PNG_COMPRESSION_TYPE_BASE`

■ `filter_method`

- ☐ `PNG_FILTER_TYPE_BASE`
- ☐ `PNG_INTRAPIXEL_DIFFERENCING`

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_bytep * row_pointers
    = (png_bytep*) malloc(sizeof(png_bytep) * height);
for (y=0; y<height; y++)
    row_pointers[y]
        = (png_byte*) malloc(rowbytes);
png_read_image(png_ptr, row_pointers);
```

Zwolnienie pamięci, zamykanie pliku

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_destroy_read_struct(&png_ptr,  
                        &info_ptr, NULL);  
  
fclose(fp);
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
if (color_type != PNG_COLOR_TYPE_RGBA)
    return (ERROR)

for (y=0; y<height; y++) {
    png_byte* row = row_pointers[y];
    for (x=0; x<width; x++) {
        png_byte* ptr = &(row[x*4]);
        printf("Pixel at position [ %d - %d ] \
            has the following RGBA values: \
            %d - %d - %d - %d\n",
            x, y, ptr[0], ptr[1], ptr[2], ptr[3]);

        ptr[0] = 0;
        ptr[1] = ptr[2];
    }
}
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
FILE *fp = fopen(file_name, "wb");  
if (!fp) return (ERROR);
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

■ png_struct

```
png_structp png_ptr = png_create_write_struct  
    (PNG_LIBPNG_VER_STRING,  
     (png_voidp)user_error_ptr,  
     user_error_fn,  
     user_warning_fn);  
if (!png_ptr) return (ERROR);
```

■ możliwe jest wykorzystanie domyślnych callbacków

```
png_structp png_ptr = png_create_write_struct  
    (PNG_LIBPNG_VER_STRING,  
     NULL,  
     NULL,  
     NULL);  
if (!png_ptr) return (ERROR);
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

■ png_info

```
png_infop info_ptr
    = png_create_info_struct(png_ptr);
if (!info_ptr){
    png_destroy_write_struct(&png_ptr,
                           (png_infopp)NULL);
    return (ERROR);
}
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
if(setjmp(png_jmpbuf(png_ptr))){\n    png_destroy_write_struct(&png_ptr,\n                            &info_ptr);\n    fclose(fp);\n    return (ERROR);\n}
```


Ustawianie parametrów obrazu

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_init_io(png_ptr, fp);  
png_set_IHDR(png_ptr, info_ptr,  
             width, height,  
             bit_depth, color_type,  
             interlace_type,  
             compression_type, filter_method);
```

- Sens parametrów jak w funkcji `png_get_IHDR`, ale
 - ☐ `compression_type` powinno być `PNG_COMPRESSION_TYPE_DEFAULT`
 - ☐ `filter_method` powinno być `PNG_FILTER_TYPE_DEFAULT`
 - ☐ może być `PNG_INTRAPIXEL_DIFFERENCING`

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_write_info(png_ptr, info_ptr);
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_write_image(png_ptr, row_pointers);
```

- `row_pointers` jest strukturą, utworzoną tak jak przy wczytywaniu obrazka.

Końcówka, zwolnienie pamięci, zamykanie pliku

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
png_write_end(png_ptr, NULL);

for (y=0; y<height; y++)
    free(row_pointers[y]);
free(row_pointers);
png_destroy_write_struct(&png_ptr,
                        &info_ptr);

fclose(fp);
```

Wprowadzenie

Libpng

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Libjpeg

Case study:
Wizualizacja

```
int main(int argc, char **argv){  
    if (argc != 3) return (ERROR);  
  
    read_png_file(argv[1]);  
    process_file();  
    write_png_file(argv[2]);  
  
    return 0;  
}
```

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

Libjpeg

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:
Wizualizacja

- Rozwijana przez Independent JPEG Group
 - nie są związani z ISO/IEC JTC1/SC29/WG1 standards committee (aka JPEG)
- Ma zaimplementowane prawie wszystkie opcje formatu JPEG
- Zgodna z ANSI C
- Pierwsze wydanie było 7 października 1991
 - najnowsza stabilna wersja 9b
 - 17 stycznia 2016
- Wolna licencja

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

- W kodzie (C, C++):
`#include <jpeglib.h>`
- W opcjach konsolidatora (linkera):
`-ljpeg`

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

- `jpeg_decompress_struct` JPEG decompression object
 - ☐ zawiera informacje o pliku
 - ☐ jest inicjalizowany po wczytaniu nagłówka
- `jpeg_compress_struct` JPEG compression object
 - ☐ zawiera informacje o parametrach kompresji
- `jpeg_error_mgr` Error handler structure
 - ☐ zawiera informacje o callbacku

- Inicjalizacja struktur

```
struct jpeg_decompress_struct cinfo;
struct jpeg_error_mgr jerr;
. . . . .
cinfo.err = jpeg_std_error(&jerr);
jpeg_create_decompress(&cinfo);
```

■ Otwieranie pliku

```
FILE *infile = fopen(file_name, "rb");
if (!infile) return (ERROR);
jpeg_stdio_src(&cinfo, infile);
```

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

```
jpeg_read_header(&cinfo, TRUE);  
jpeg_start_decompress(&cinfo);
```

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

■ Z pól struktury `jpeg_decompress_struct`

```
JDIMENSION width = cinfo.output_width;  
JDIMENSION height = cinfo.output_height;  
int num_components = cinfo.out_color_components;  
J_COLOR_SPACE color_space  
                = cinfo.out_color_space;
```

■ `num_components`: 1 lub 3

■ `J_COLOR_SPACE`

- ☐ `JCS_UNKNOWN`
- ☐ `JCS_GRAYSCALE`
- ☐ `JCS_RGB`
- ☐ `JCS_YCbCr`
- ☐ `JCS_CMYK`
- ☐ `JCS_YCCK`

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Case study:
Wizualizacja

```
JSAMPARRAY row_pointers = (JSAMPARRAY)
    malloc(sizeof(j_common_ptr) * height);
for (y=0; y<height; y++)
    row_pointers[y] = (JSAMPROW) malloc(rowbytes);

y=0;
JSAMPARRAY tmp = row_pointers;
while( cinfo.output_scanline < cinfo.image_height ){
    y = jpeg_read_scanlines( &cinfo, tmp, 1 );
    tmp +=y;
}
```

Końcówka Zwolnienie pamięci, zamykanie pliku

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

```
jpeg_finish_decompress(&cinfo);  
jpeg_destroy_decompress(&cinfo);  
free(row_pointer[0]);  
fclose(infile);
```

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:
Wizualizacja

```
if (color_space != JCS_RGB) return;

for (y=0; y<height; y++) {
    JSAMPROW row = row_pointers[y];
    for (x=0; x<width; x++) {
        JSAMPROW ptr = &(row[x*3]);
        printf("Pixel at position [ %d - %d ] \
            has the following RGB values: \
            %d - %d - %d\n",
            x, y, ptr[0], ptr[1], ptr[2]);

        ptr[0] = 0;
        ptr[1] = ptr[2];
    }
}
```

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

```
FILE *outfile = fopen(filename, "wb");  
if (!outfile) return (ERROR);
```


Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

```
struct jpeg_compress_struct cinfo;
```

```
struct jpeg_error_mgr jerr;
```

```
cinfo.err = jpeg_std_error( &jerr );
```

```
jpeg_create_compress(&cinfo);
```

```
jpeg_stdio_dest(&cinfo, outfile);
```

Ustawianie parametrów obrazu

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetywarzanie

Zapisywanie

main

Case study:

Wizualizacja

```
cinfo.image_width = width;
cinfo.image_height = height;
cinfo.input_components = num_components;
cinfo.in_color_space = color_space;

int quality = 75;
jpeg_set_defaults( &cinfo );
jpeg_set_quality (&cinfo, quality, TRUE);
```

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:
Wizualizacja

```
tmp = row_pointers;
while( cinfo.next_scanline < cinfo.image_height){
    y = jpeg_write_scanlines( &cinfo, tmp, 1 );
    tmp +=y;
}
```

- `row_pointers` jest strukturą, utworzoną tak jak przy wczytywaniu obrazka.

Końcówka, zamykanie pliku, zwolnienie pamięci

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:

Wizualizacja

```
jpeg_finish_compress( &cinfo );  
jpeg_destroy_compress( &cinfo );  
fclose( outfile );
```

```
for (y=0; y<height; y++)  
    free(row_pointers[y]);  
free(row_pointers);
```

Wprowadzenie

Libpng

Libjpeg

Wprowadzenie

Struktury

Wczytywanie

Przetwarzanie

Zapisywanie

main

Case study:
Wizualizacja

```
int main(int argc, char **argv){  
    if (argc != 3) return (ERROR);  
  
    read_jpeg_file(argv[1]);  
    process_file();  
    write_jpeg_file(argv[2]);  
    return 0;  
}
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

Case study: Wizualizacja

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

- W wyniku numerycznych testów powstała tablica danych (liczb rzeczywistych). Trzeba te dane wizualizować
- Rozwiązanie:
 - utworzmy plik png: każdy piksel reprezentuje element tablicy poprzez skalę szarości:
 - najmniejsza wartość — piksel czarny
 - największa wartość — piksel biały
 - Najmniejsza i największa wartości nie muszą się pokrywać z najmniejszą i największą wartościami pośród danych
 - Napiszmy prosty program w C++

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

- Plik wejściowy ma strukturę
 - ☐ w pierwszej linijce podana jest szerokość
 - ☐ w drugiej — wysokość
 - ☐ dalej jest tablica danych `double` w postaci binarnej

`%Nx: 128`

`%Ny: 128`

`.....>?sd<,+\nma????`

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
#include <iostream>
```

```
#include <fstream>
```

```
#include <cstdlib>
```

```
#include <cmath>
```

```
#include <opt.h>
```

```
#include <png.h>
```

```
#include "input.h"
```

```
using namespace std;
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
#ifndef INPUT
```

```
#define INPUT
```

```
#include <fstream>
```

```
using namespace std;
```

```
void readSize(ifstream & in, int & P, int & Q);
```

```
#endif
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
#include <cstring>
```

```
#include <iostream>
```

```
#include <cstdlib>
```

```
#include "input.h"
```

```
void readSize(istream & in, int &Nx, int &Ny){  
    char s[256];  
    in.getline(s,256);  
    if(strncmp(s, "%Nx: ", 4)==0){  
        Nx=atoi(s+5);  
    }  
    else cerr<<"Bad file format?? (first line)"  
        <<endl;
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
in.getline(s,256);  
if(strncmp(s, "%Ny: ", 4)==0){  
    Ny=atoi(s+5);  
}  
else cerr<<"Bad file format?? (second line)"  
    <<endl;  
return;  
} //Koniec funkcji
```

Opcje wiersza poleceń. Wartości domyślne

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
double Max=2.5;  
double Min=-1.5;  
char * inPixelFile=(char*)"test.pixels";  
char * outPNGFile=(char*)"test.png";
```

Opcje wiersza poleceń. Moduł opt

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
int main(int argc, char **argv){
    // Options
    optrega(&Max, OPT_DOUBLE, 'M', (char*) "max",
            (char*) "Maximum data value");
    optrega(&Min, OPT_DOUBLE, 'm', (char*) "min",
            (char*) "Minimum data value");
    optrega(&inPixelFormat, OPT_STRING, 'i',
            (char*) "in-pixel-file",
            (char*) "Input Pixels File");
    optrega(&outPNGFile, OPT_STRING, 'o',
            (char*) "out-png-file",
            (char*) "Output PNG File");
    opt(&argc,&argv);
    opt_free();
}
```

Wczytywanie rozmiaru tablicy

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
// Reading size
ifstream in(inPixelFormat, ifstream::binary);
int Nx, Ny;
readSize(in, Nx, Ny);
cout<<"Max: "<<Max<<endl<<"Min: "<<Min<<endl;
cout<<"Nx: "<<Nx<<endl<<"Ny: "<<Ny<<endl
  <<"in pixels file: "<<inPixelFormat<<endl
  <<"out PNG file: "<<outPNGFile<<endl;
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
cout<<"reading pixels data from "  
      <<inPixelFile<<"..."<<endl;  
int Size=Nx*Ny;  
double* R=new double[Size];  
in.read((char*)R,Size*sizeof(double));  
in.close();
```


Otwieramy plik wyjściowy

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
cout<<"outputting PNG data to "  
      <<outPNGFile<<"..."<<endl;  
  
FILE *fp = fopen(outPNGFile, "wb");  
if (!fp){  
    cerr<<"Error opening file "<<outPNGFile<<endl;  
    return 1;  
}
```

Utworzenie struktury dla zapisu

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
png_structp png_ptr = png_create_write_struct(  
    PNG_LIBPNG_VER_STRING, NULL, NULL, NULL);  
if (!png_ptr) return (2);  
png_infop info_ptr  
    = png_create_info_struct(png_ptr);  
if (!info_ptr){  
    png_destroy_write_struct(&png_ptr,  
        (png_infopp) NULL);  
    return (3);  
}
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
png_init_io(png_ptr, fp);
```

```
png_set_IHDR(png_ptr, info_ptr,  
             Nx, Ny, 8,  
             PNG_COLOR_TYPE_GRAY,  
             PNG_INTERLACE_NONE,  
             PNG_COMPRESSION_TYPE_DEFAULT,  
             PNG_FILTER_TYPE_DEFAULT);
```

```
cout<<"\twriting header ..."<<endl;  
png_write_info(png_ptr, info_ptr);
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
cout<<"\twriting image data ..."<<endl;
png_bytep row_pointer=new png_byte[Nx];
for (int j=0;j<Ny;j++){
    for(int i=0;i<Nx;i++){
        int val=(R[Nx*j+i]-Min)/(Max-Min)*256+0.5;
        if (val>255) val=255;
        else if (val<0) val=0;
        row_pointer[i]=(png_byte)val;
    }
    png_write_row(png_ptr, row_pointer);
}
delete row_pointer;
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
png_write_end(png_ptr, NULL);  
fclose(fp);  
png_destroy_write_struct(&png_ptr,  
                        &info_ptr);  
  
return 0;  
}
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
GCC = g++
LIBS = -lpng -lm -lopt
SOURCES = $(wildcard *.cpp)
OBJECTS := $(SOURCES:.cpp=.o)
EXECUTABLE = pixels2png

all: $(SOURCES) $(EXECUTABLE)

$(EXECUTABLE): $(OBJECTS)
    $(GCC) $(OBJECTS) -o $@ $(LIBS)

clean:
    rm -f $(EXECUTABLE) $(OBJECTS)
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

\$ make

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
$ ./pixels2png -i test2.pixels -o test2.png \
-m -0.5 -M 1.5
```

```
Max: 1.5
```

```
Min: -0.5
```

```
Nx: 128
```

```
Ny: 128
```

```
in pixels file: test2.pixels
```

```
out PNG file: test2.png
```

```
reading pixels data from test2.pixels...
```

```
outputting PNG data to test2.png...
```

```
    writing header ...
```

```
    writing image data ...
```


Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
$ ./pixels2png --help
```

```
Usage: pixels2png [options]
```

```
To invoke the menu, type:
```

```
        pixels2png --menu
```

```
The options are:
```

```
-M, --max          <DOUBLE>
```

Maximum data value

```
-m, --min          <DOUBLE>
```

Minimum data value

```
-i, --in-pixel-file <STRING>
```

Input Pixels File

```
-o, --out-png-file  <STRING>
```

Output PNG File

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

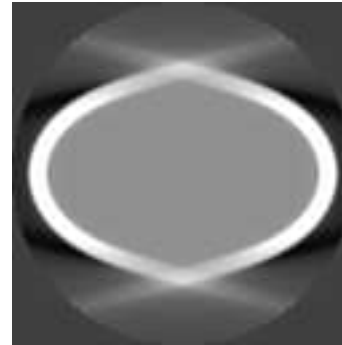
Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa



Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

■ ImageMagick

- ☐ legalny, wieloplatformowy, ma API dla wielu języków programowania (w tym PHP)
- ☐ www.imagemagick.org/



```
convert -scale 1600% test2.png test2x1600.png
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

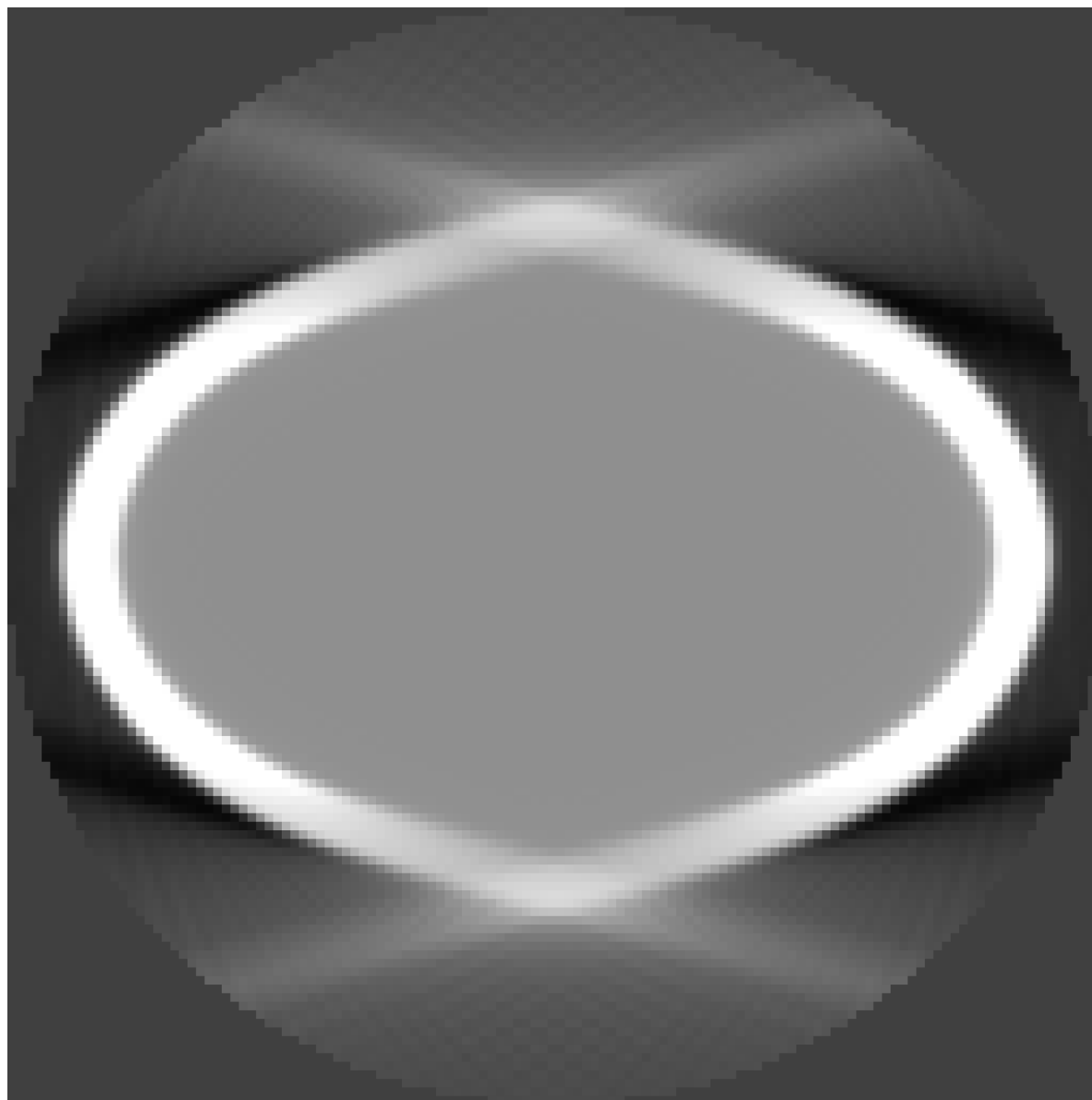
Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa



Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
montage -pointsize 72 -title 'test 2 results' \  
-geometry +16+16 test2x1600.png \  
test2x1600title.png
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

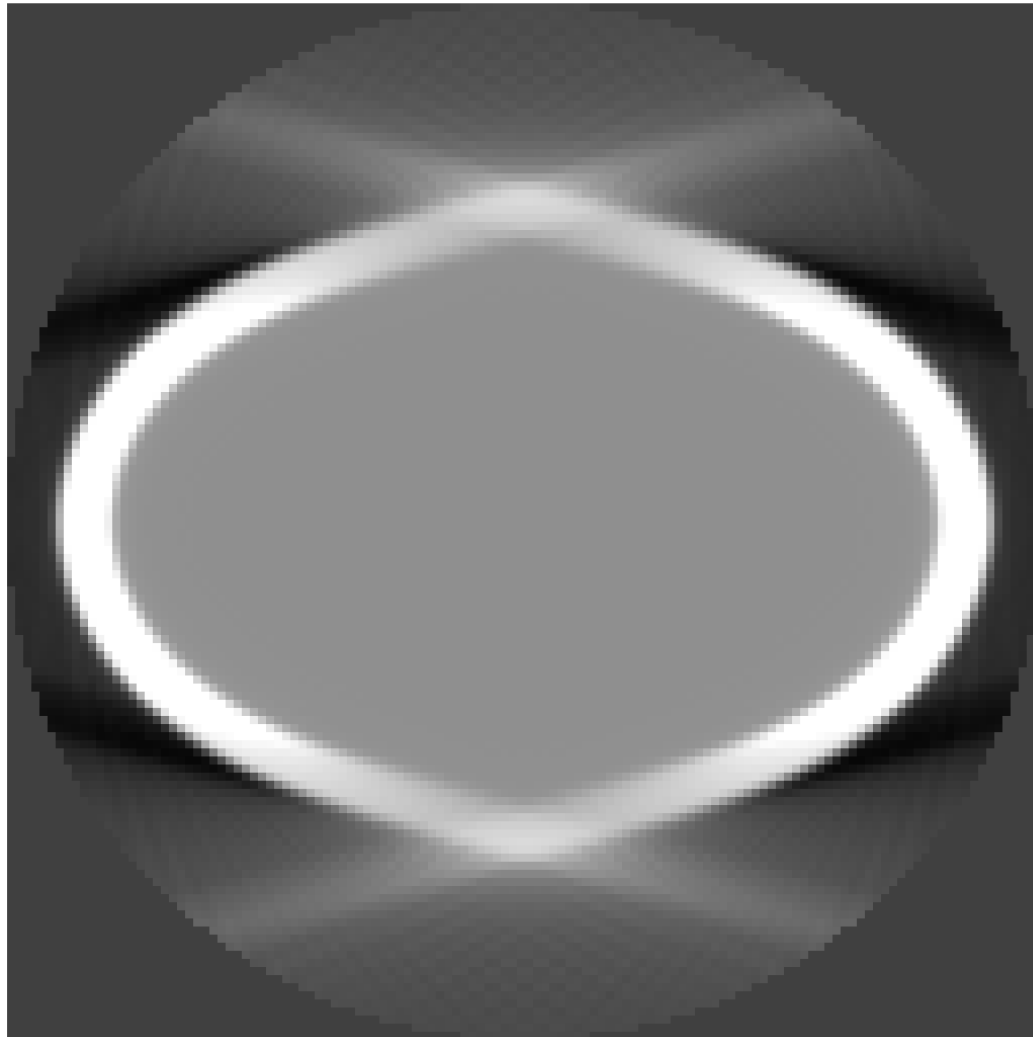
Makefile

Użycie

ImageMagick

Makefile dwa

test 2 results



WprowadzenieLibpngLibjpegCase study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

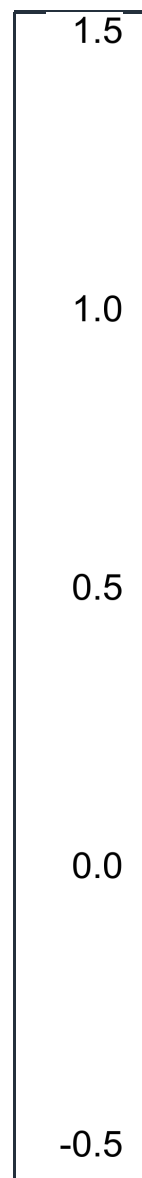
Użycie

ImageMagick

Makefile dwa

```
convert -size 150x2272 -background white \  
-fill black -interline-spacing 470 \  
-gravity east -pointsize 72 \  
label:'1.5\n1.0\n0.5\n0.0\n-0.5' scale.png
```

<u>Wprowadzenie</u>
<u>Libpng</u>
<u>Libjpeg</u>
<u>Case study: Wizualizacja</u>
Zagadnienie
Implementacja
Makefile
Użycie
ImageMagick
Makefile dwa



Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
convert -size 160x2272 gradient:white \  
gradient.png
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa



Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
montage -tile 3x1 \  
test2x1600title.png \  
-label ' ' scale.png \  
gradient.png \  
-geometry +2+0 \  
wynik.png
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

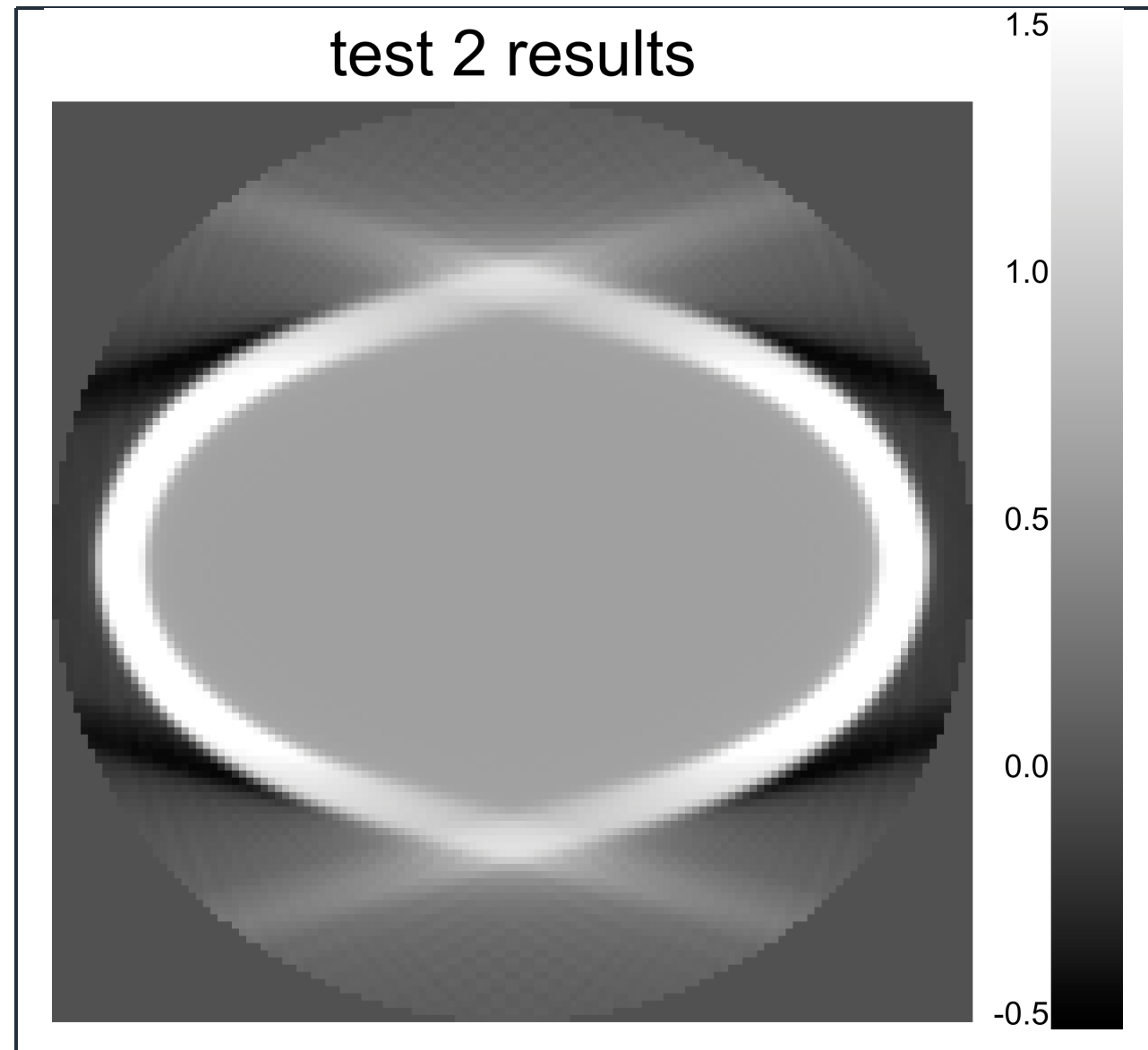
Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa



Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

■ Zmienne

```
PNGDIR = Output
```

```
MAX = 1.50
```

```
MIN = -0.50
```

```
MED = $(shell echo "scale=2;\n      (${MAX}+${MIN})*0.5" | bc )
```

```
Q1 = $(shell echo "scale=2;\n      (${MAX}+${MIN})*0.25" | bc )
```

```
Q3 = $(shell echo "scale=2;\n      (${MAX}+${MIN})*0.75" | bc )
```

Wprowadzenie
Libpng
Libjpeg
Case study: Wizualizacja
Zagadnienie
Implementacja
Makefile
Użycie
ImageMagick
Makefile dwa

```
all: $(SOURCES) $(EXECUTABLE) images
images: $(patsubst %.pixels, %.png,\
    $(addprefix $(PNGDIR)/, $(wildcard *.pixels)))
$(PNGDIR)/%.png: %x1600title.png gradient.png scale.png
    montage -tile 3x1 $< -label ' '\
    scale.png gradient.png -geometry +2+0 $@
gradient.png:
    convert -size 160x2272 \
    gradient:white gradient.png
scale.png:
    convert -size 150x2272 -background white\
    -fill black -interline-spacing 470\
    -gravity east -pointsize 72 \
    label: '$(MAX)\n$(Q3)\n$(MED)\n$(Q3)\n$(MIN)'\
    scale.png
```

Wprowadzenie

Libpng

Libjpeg

Case study:
Wizualizacja

Zagadnienie

Implementacja

Makefile

Użycie

ImageMagick

Makefile dwa

```
%x1600title.png: %x1600.png
montage -pointsize 72 -title \
'test 2 results' -geometry +16+16 $< $@
```

```
%x1600.png: %.png
convert -scale 1600% $< $@
```

```
%.png: %.pixels
./$(EXECUTABLE) -i $< -o $@ \
-M $(MAX) -m $(MIN)
```