

Formatos de imágenes

Herramientas de programación para
procesamiento de señales

Índice

- Resumen de diferentes formatos de imágenes
- Formato ***PGM***
 - Tipos
 - Encabezado
- Formato ***PPM***
- Operación simple → aplicar umbral
- Apéndice → librerías ***magick*** y ***opencv***

Formatos de imágenes

- Maneras estándar de almacenar imágenes digitales
- Para diferentes tipos de imágenes:
 - Rasterizadas → JPEG, TIFF, RAW, PGM, PNG, BMP
 - HDR → RGBE, IFF-RGFX
 - Vectoriales → CGM, SVG, AI, CDR
 - Stereo → MPO, PNS, JPS
- Con y sin pérdida
- Texto ó binario

PGM – Portable Gray Map

<http://netpbm.sourceforge.net/doc/pgm.html>

- Formato de gráficos simple en escala de grises
- 8 bits/píxel si **max** ≤ 255
- 16 bits/píxel si **255** < **max** < **65536**
- Encabezado

P2

→ “magic number” (espacio o nueva línea)

128 128

→ #columnas (espacio) #filas (espacio o nueva línea)

255

→ máximo valor de gris (espacio o nueva línea)

- Datos separados por espacio (de izquierda a derecha y de arriba a abajo)

PPM – Portable Pixel Map

<http://netpbm.sourceforge.net/doc/ppm.html>

- Formato de gráficos simple en color
- 24 bits/píxel si **max** ≤ 255 (8 bits/canal)
- 48 bits/píxel si **255** < **max** < **65536** (16 bits/canal)
- Encabezado

P3

→ “magic number” (espacio o nueva línea)

128 128

→ #columnas (espacio) #filas (espacio o nueva línea)

255

→ máximo valor (espacio o nueva línea)

- Datos separados por espacio (de izquierda a derecha y de arriba a abajo, en orden RGB)

Ejemplo #1

P3

3 2

255

255 0 0

255 255 255

0 0 255

255 0 0

255 255 255

0 0 255

Ejemplo #1

P3

3 2

255

255 0 0

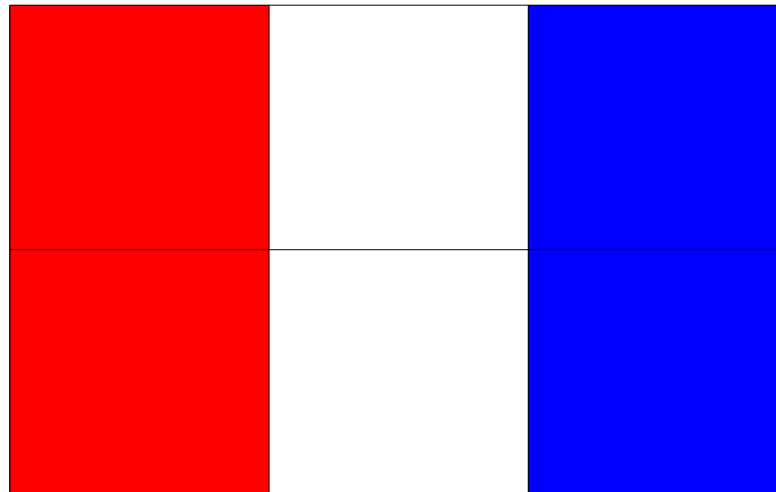
255 255 255

0 0 255

255 0 0

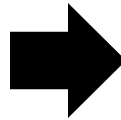
255 255 255

0 0 255



Umbral (*Thresholding*)

- Prender píxeles cuyo valor es mayor o igual a cierto umbral, y apagar los otros.



Apéndice

magick & opencv

Herramientas de programación para
procesamiento de señales

GraphicsMagick

<http://www.graphicsmagick.org/>

- Colección de herramientas y librerías para leer, escribir y manipular imágenes
- Soporte para más de 88 formatos
- Portable (Unix, OSX, Win)
- Multi-thread (usando OpenMP)
- Funciones para varios lenguajes → C, C++, Perl, PHP, Python, etc...

Ejemplo #2 (parte 1)

```
#include <magick/api.h>

int main(int argc, char** argv)
{
    ExceptionInfo exception;
    Image *image;
    ImageInfo *image_info;
    InitializeMagick(".");
    GetExceptionInfo(&exception);
    image_info=CloneImageInfo((ImageInfo *) NULL);
    (void) strcpy(image_info->filename, argv[1]);
    image=ReadImage(image_info,&exception);

    if (image == (Image *) NULL)
        MagickError(exception.severity,...
            exception.reason,exception.description);

    int COLS = image -> columns;
    int ROWS = image -> rows;
```

Ejemplo #2 (parte 2)

```
//output image
ImageInfo *image_info_out=NULL;
image_info_out=CloneImageInfo((ImageInfo *) NULL);
image_info_out->depth=8;
Image *image_out=NULL;
image_out=AllocateImage(image_info_out);

if (image == (Image *) NULL)
    MagickError(ResourceLimitError,
        "Unable to save image",
        "Memory allocation failed");

image_info_out=CloneImageInfo((ImageInfo *) NULL);
strcpy( image_info_out->filename, argv[2] );
image_out -> rows = image->rows;
image_out -> columns = image->columns;
image_out->depth=8;
```

Ejemplo #2 (parte 3)

```
PixelPacket *q;

for (int i=0; i < ROWS; i++ )
    for (int j=0; j < COLS; j++){
        int offset=COLS*i+j;
        int r=(int)rint(GetOnePixel(image,j,i).red);
        int g=(int)rint(GetOnePixel(image,j,i).green);
        int b=(int)rint(GetOnePixel(image,j,i).blue);

        q=SetImagePixels(image_out,j,i,1,1);
        if (q == (PixelPacket *) NULL){
            break;
        }

        q -> red = ( (unsigned) ( r ) );
        q -> green = ( (unsigned) ( g ) );
        q -> blue = ( (unsigned) ( b ) );
        q++;
    }
SyncImagePixels(image_out);
```

Ejemplo #2 (parte 4)

```
WriteImage(image_info_out,image_out);

DestroyImage(image_out);
DestroyImageInfo(image_info_out);

DestroyExceptionInfo(&exception);
DestroyImage(image);
DestroyImageInfo(image_info);
DestroyMagick();
}
```

OpenCV

<http://opencv.willowgarage.com/>

- Librería de funciones de programación para visión por computadora en tiempo real
- Portable (Unix, OSX, Win, Android)
- Funciones para varios lenguajes → C, C++, Python (Proximamente Java)

Ejemplo #3

```
#include "opencv2/core/core_c.h"
#include "opencv2/highgui/highgui_c.h"
#include "opencv2/imgproc/imgproc_c.h"

int main ( int argc, char **argv )
{
    IplImage *in = cvLoadImage(argv[1], CV_LOAD_IMAGE_COLOR);
    IplImage *out = cvCreateImage(cvGetSize(in),
                                in->depth, in->nChannels);
    cvThreshold(in, out, 128, 255, CV_THRESH_BINARY);

    cvSaveImage(argv[2], out, NULL);

    cvReleaseImage(&in);
    cvReleaseImage(&out);

    return 0;
}
```