

# Tarea 1

Angel Manrique Pozos Flores; N.C M07211505

*Instituto Tecnológico Nacional de México, Blvd. Industrial, Mesa de Otay, 22430 Tijuana, B.C., Mexico.*

(Dated: 2 de Febrero del 2016)

El presente trabajo habla sobre como empezar a utilizar las librerías de Vision de OpenCV y el software CodeBlocks, utilizando una pequeña imagen de 16x16 pixeles, haciéndole un procesamiento para obtener sus componentes RGB y sus matrices de valores correspondientes.

## I. INTRODUCCIÓN

El procesamiento de imágenes es de gran utilidad en la mayoría de las áreas de investigación ya que todo lo que captamos en el mundo la gran mayoría de los datos provienen de nuestros sentidos en especial la vista.

Por ello el estudio de la visión computacional es de gran importancia, el presente trabajo se hace un acercamiento a esta área, utilizando las librerías de visión open source OpenCV y el software libre CodeBlocks el cual se configuró para que pudiera ser capaz de reconocer las librerías de OpenCV.

## II. CONSTRUCCIÓN

Se utilizó una figura de 16x16 pixeles de tamaño que contenga información en RGB (Red, Green, Blue) la cual se muestra a continuación.



Figura 1. Imagen de 16x16 pixeles con zoom para poder ser apreciada, la cual contiene sus correspondientes de rojo, verde y azul así como su correspondiente en escala de grises.

Ya que se requiere descomponer la imagen en sus correspondientes valores de RGB se desarrolló un código en C++ con las librerías de visión de OpenCV para la realización de la práctica, el siguiente código se muestra a continuación.

```
#include <stdlib.h>
#include <cv.hpp>
#include <cxcore.hpp>
#include <highgui.h>
#include <iostream>
```

```
using namespace std;
using namespace cv;
```

```
int main()
{
```

```
Mat img = imread("1.jpg",
    CV_LOAD_IMAGE_COLOR);
namedWindow("original image",1);
imshow("original image", img);
```

```
if(img.empty())
{
    return -1;
}
```

```
Mat gray;
cvtColor(img, gray, CV_BGR2GRAY);
namedWindow("gray image",
    CV_WINDOW_AUTOSIZE);
imshow("gray image", gray);
```

```
vector<Mat> rgbChannels(3);
split(img, rgbChannels);
Mat Values;
Mat fin_img;
```

```
Values = Mat::zeros(Size(img.cols,
    img.rows), CV_8UC1);
```

```
{
vector<Mat> channels;
channels.push_back(Values);
channels.push_back(Values);
channels.push_back(rgbChannels[2]);
```

```
merge(channels, fin_img);
namedWindow("RED", 1);
imshow("RED", fin_img);
}
```

```
{
vector<Mat> channels;
channels.push_back(Values);
channels.push_back(rgbChannels[1]);
channels.push_back(Values);
```

```
merge(channels, fin_img);
namedWindow("GREEN", 1);
imshow("GREEN", fin_img);
}
```

```

{
vector<Mat> channels;
channels.push_back(rgbChannels[0]);
channels.push_back(Values);
channels.push_back(Values);

merge(channels, fin_img);
namedWindow("BLUE", 1);
imshow("BLUE", fin_img);
}

Mat RED = rgbChannels[2];
Mat GREEN = rgbChannels[1];
Mat BLUE = rgbChannels[0];

cout << "RGB MATRIX = " << endl << "
" << img << endl << endl;
cout << "RED MATRIX = " << endl << "
" << RED << endl << endl;
cout << "GREEN MATRIX = " << endl << "
" << GREEN << endl << endl;
cout << "BLUE MATRIX = " << endl << "
" << BLUE << endl << endl;

waitKey(0);
return 0;
}

```

Las imagenes obtenidas de RGB son las siguientes.

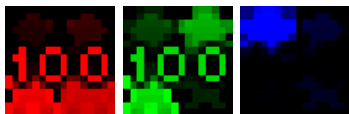


Figura 2. Correspondientes para el color rojo, verde y azul.

Asi como su correspondiente matriz de elementos para la imagen original.

y su matriz de elementos para la escala de grises.

y las matrices para los layers de RGB de nuestra imagen serian.

### III. CONCLUSIONES

La importancia de aprender a manejar imagenes de forma matricial es de gran importancia ya que de esta forma se puede tener mayor control sobre las operaciones que se desean realizar sobre la imagen, ahora sabemos gracias a la practica realizada que una imagen comunmente tendra tres componentes principales en RGB asi como un correspondiente en escalas de grises

esta se maneja en valores que van desde el 0 hasta el 255 la siguiente practica tendra como objetivo trabajar la imagen en esta forma, por ello era de vital importancia

```

unit done
opencv support available
RGB MATRIX =
[128, 0, 0, 128, 0, 0, 223, 16, 0, 255, 31, 0, 255, 31, 0, 223
, 32, 0, 128, 0, 0, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 137
, 0, 0, 152, 0, 0, 152, 0, 0, 102, 0, 0, 0, 0, 0, 0, 0
128, 0, 0, 128, 0, 0, 255, 30, 0, 255, 77, 44, 255, 77, 44, 255
30, 0, 128, 0, 0, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 148
0, 56, 182, 56, 56, 182, 56, 0, 145, 0, 0, 0, 0, 0, 0, 0
255, 33, 0, 255, 33, 0, 255, 69, 35, 255, 69, 35, 255, 69, 35, 255
69, 35, 255, 86, 53, 255, 29, 0, 0, 136, 0, 72, 185, 72, 54, 175
52, 94, 175, 94, 94, 175, 94, 52, 175, 92, 0, 139, 0, 0, 142, 0
255, 31, 0, 255, 64, 30, 255, 64, 30, 255, 64, 30, 255, 64, 30, 255
64, 30, 255, 64, 30, 255, 41, 0, 0, 155, 0, 51, 189, 50, 52, 189
52, 51, 189, 50, 51, 189, 50, 51, 189, 50, 50, 189, 50, 0, 134, 0
128, 0, 0, 255, 35, 0, 255, 61, 29, 255, 61, 29, 255, 61, 29, 255
61, 29, 255, 30, 0, 128, 0, 0, 0, 0, 0, 0, 0, 127, 0, 47, 162
47, 47, 162, 49, 49, 162, 47, 47, 162, 49, 0, 129, 0, 0, 0, 0
223, 32, 0, 108, 13, 0, 31, 7, 3, 123, 28, 13, 255, 60, 27, 123
28, 13, 23, 3, 0, 0, 0, 0, 0, 1, 0, 123, 0, 45, 156
45, 21, 75, 21, 5, 18, 5, 5, 18, 5, 0, 49, 0, 0, 83, 0
36, 4, 0, 31, 7, 3, 0, 216, 255, 26, 3, 0, 108, 13, 0, 31
6, 3, 0, 216, 255, 0, 216, 255, 0, 2, 0, 22, 72, 22, 20, 71
20, 0, 10, 0, 0, 0, 216, 255, 0, 216, 255, 5, 18, 5, 0, 11, 0
16, 2, 0, 0, 216, 255, 0, 216, 255, 5, 0, 0, 7, 0, 0, 0
216, 255, 30, 6, 3, 19, 2, 0, 0, 216, 255, 4, 17, 4, 5, 17
5, 0, 216, 255, 0, 2, 0, 5, 17, 5, 0, 216, 255, 0, 7, 0
0, 0, 0, 0, 0, 0, 216, 255, 0, 21, 21, 0, 21, 0, 21, 0
216, 255, 0, 0, 0, 0, 0, 0, 216, 255, 0, 0, 0, 0, 0, 0
1, 0, 216, 255, 0, 0, 18, 0, 0, 0, 0, 216, 255, 0, 0, 0
0, 0, 0, 0, 0, 0, 216, 255, 1, 31, 31, 1, 31, 31, 0
216, 255, 0, 0, 0, 0, 0, 0, 0, 216, 255, 0, 0, 0, 0, 0
9, 0, 216, 255, 5, 5, 26, 0, 0, 9, 0, 216, 255, 0, 0, 0
0, 255, 255, 0, 26, 26, 0, 216, 255, 1, 29, 31, 4, 117, 123, 1
29, 31, 0, 216, 255, 0, 216, 255, 0, 6, 24, 24, 107, 16, 16
106, 4, 4, 23, 0, 216, 255, 0, 216, 255, 0, 0, 21, 0, 0, 41
0, 253, 255, 3, 113, 121, 0, 28, 30, 3, 113, 121, 7, 235, 251, 3
113, 121, 0, 28, 30, 4, 31, 31, 27, 27, 107, 34, 34, 215, 34, 34
215, 16, 16, 103, 4, 4, 26, 4, 4, 26, 16, 16, 103, 0, 0, 213
0, 0, 0, 0, 255, 255, 5, 224, 240, 5, 224, 240, 5, 224, 240, 5
224, 240, 0, 255, 255, 0, 0, 0, 0, 0, 131, 0, 0, 212, 33, 33
206, 33, 33, 206, 33, 33, 206, 33, 33, 206, 0, 0, 212, 0, 0, 131
0, 0, 0, 0, 253, 255, 4, 213, 229, 4, 213, 229, 4, 213, 229, 4
213, 229, 0, 255, 255, 0, 0, 0, 0, 198, 0, 0, 214, 32, 32
201, 32, 32, 201, 32, 32, 201, 32, 32, 201, 0, 0, 213, 0, 0, 198
0, 255, 255, 4, 211, 226, 4, 210, 224, 0, 245, 255, 0, 253, 255, 2
206, 221, 2, 206, 221, 0, 255, 255, 0, 0, 213, 31, 31, 198, 31, 31
198, 0, 0, 213, 0, 0, 214, 31, 31, 194, 31, 31, 194, 0, 0, 210
0, 255, 255, 0, 183, 198, 0, 183, 210, 0, 255, 255, 0, 255, 255, 0
193, 210, 0, 183, 198, 0, 255, 255, 0, 0, 214, 0, 0, 178, 0, 0
185, 0, 0, 211, 0, 0, 213, 0, 0, 185, 0, 0, 178, 0, 0, 213]

```

Figura 3. Matriz de elementos para la imagen original.

```

GRAY MATRIX =
[ 15, 15, 35, 47, 47, 44, 15, 15, 0, 0, 80, 89, 89, 60, 0, 0
15, 15, 47, 87, 87, 47, 15, 15, 0, 0, 87, 130, 130, 85, 0, 0
48, 48, 80, 80, 80, 80, 96, 46, 80, 138, 124, 125, 125, 124, 82, 83
47, 76, 76, 76, 76, 76, 76, 53, 91, 120, 121, 120, 120, 120, 79
15, 50, 74, 74, 74, 74, 47, 15, 0, 75, 115, 115, 115, 115, 76, 0
44, 20, 9, 34, 72, 34, 4, 0, 1, 72, 110, 53, 13, 13, 26, 50
6, 9, 203, 5, 20, 8, 203, 203, 1, 51, 50, 6, 203, 203, 13, 6
3, 203, 203, 1, 1, 203, 8, 3, 203, 12, 12, 203, 1, 12, 203, 4
0, 0, 203, 19, 19, 203, 0, 0, 203, 0, 0, 203, 5, 0, 203, 0
0, 0, 203, 28, 28, 203, 0, 0, 203, 0, 3, 203, 11, 3, 203, 0
226, 23, 203, 26, 106, 26, 203, 203, 2, 49, 43, 11, 203, 203, 6, 12
225, 103, 25, 103, 214, 103, 25, 28, 51, 88, 88, 42, 11, 11, 42, 64
0, 226, 204, 204, 204, 204, 226, 0, 57, 63, 85, 85, 85, 63, 57
0, 225, 194, 194, 194, 194, 226, 0, 59, 64, 83, 83, 83, 64, 59
226, 192, 191, 220, 225, 187, 187, 226, 64, 81, 80, 64, 80, 80, 63
226, 167, 176, 226, 226, 176, 167, 226, 64, 53, 95, 63, 64, 95, 53, 64]

```

Figura 4. Matriz de elementos para la imagen en escala de grises.

obtener un buen resultado en esta.

### IV. BIBLIOGRAFÍA

- OpenCV Tutorials, load and display a image.  
(<http://tinyurl.com/o6wh3cr>)

```

RED MATRIX =
[ 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
;
0, 0, 0, 44, 44, 0, 0, 0, 0, 0, 0, 55, 55, 0, 0, 0, 0;
0, 0, 35, 35, 35, 35, 55, 0, 0, 72, 52, 54, 54, 52, 0, 0, 0;
0, 30, 30, 30, 30, 30, 0, 0, 50, 52, 50, 50, 50, 50, 0, 0, 0;
0, 0, 23, 23, 23, 23, 0, 0, 0, 0, 47, 49, 47, 49, 0, 0, 0;
0, 0, 3, 13, 27, 13, 0, 0, 0, 0, 46, 21, 5, 5, 0, 0, 0;
0, 3, 255, 0, 0, 3, 255, 255, 0, 22, 20, 0, 255, 255, 5, 0, 0;
0, 255, 255, 0, 0, 255, 3, 0, 255, 4, 5, 255, 0, 5, 255, 0, 0;
0, 0, 255, 21, 21, 255, 0, 0, 255, 0, 1, 255, 18, 0, 255, 0, 0;
0, 0, 255, 31, 31, 255, 0, 0, 255, 0, 3, 255, 26, 3, 255, 0, 0;
255, 26, 255, 31, 123, 31, 255, 255, 6, 107, 106, 26, 255, 255, 21, 41, 0;
255, 121, 30, 121, 251, 121, 30, 31, 107, 215, 215, 103, 26, 26, 103, 213, 0;
0, 255, 240, 240, 240, 240, 255, 0, 191, 212, 206, 206, 206, 206, 212, 191, 0;
0, 255, 229, 229, 229, 229, 255, 0, 198, 214, 201, 201, 201, 201, 213, 198, 0;
255, 226, 224, 255, 255, 221, 221, 255, 213, 198, 196, 213, 214, 194, 194, 210, 0;
255, 198, 210, 255, 255, 210, 198, 255, 214, 178, 185, 211, 213, 185, 178, 213]

GREEN MATRIX =
[ 0, 0, 0, 16, 31, 31, 32, 0, 0, 0, 0, 137, 152, 152, 102, 0, 0
;
0, 0, 30, 77, 77, 30, 0, 0, 0, 0, 148, 182, 182, 145, 0, 0, 0;
33, 33, 69, 69, 69, 69, 86, 29, 136, 185, 175, 175, 175, 175, 139, 142, 0;
31, 64, 64, 64, 64, 64, 41, 155, 169, 169, 169, 169, 169, 169, 134, 0;
0, 35, 61, 61, 61, 61, 30, 0, 0, 127, 162, 162, 162, 162, 129, 0, 0;
32, 13, 7, 28, 60, 28, 3, 0, 1, 123, 156, 75, 18, 18, 45, 85, 0;
4, 7, 216, 3, 13, 6, 216, 216, 2, 72, 71, 10, 216, 216, 18, 11, 0;
2, 216, 216, 0, 0, 216, 6, 2, 216, 17, 17, 216, 2, 17, 216, 7, 0;
0, 0, 216, 21, 21, 216, 0, 0, 216, 0, 0, 216, 0, 0, 216, 0, 0;
0, 0, 216, 31, 31, 216, 0, 0, 216, 0, 0, 216, 5, 0, 216, 0, 0;
255, 26, 216, 29, 117, 29, 216, 216, 0, 24, 16, 4, 216, 216, 0, 0, 0;
253, 113, 28, 113, 235, 113, 28, 31, 27, 34, 34, 16, 4, 4, 16, 0, 0;
0, 255, 224, 224, 224, 224, 255, 0, 0, 0, 33, 33, 33, 33, 0, 0, 0;
0, 253, 213, 213, 213, 213, 255, 0, 0, 0, 32, 32, 32, 32, 0, 0, 0;
255, 211, 210, 245, 253, 206, 206, 255, 0, 31, 31, 0, 0, 31, 31, 0, 0;
255, 183, 133, 255, 255, 133, 183, 255, 0, 0, 0, 0, 0, 0, 0, 0, 0]

BLUE MATRIX =
[128, 128, 223, 255, 255, 223, 128, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0
;
128, 128, 255, 255, 255, 255, 128, 128, 0, 0, 0, 56, 56, 0, 0, 0, 0;
255, 255, 255, 255, 255, 255, 255, 255, 0, 72, 54, 54, 54, 52, 0, 0, 0;
255, 255, 255, 255, 255, 255, 255, 255, 0, 51, 52, 51, 51, 51, 50, 0, 0;
128, 255, 255, 255, 255, 255, 255, 128, 0, 0, 0, 47, 47, 49, 47, 0, 0, 0;
223, 108, 31, 123, 255, 123, 23, 0, 0, 0, 45, 21, 5, 5, 0, 0, 0;
36, 31, 0, 26, 108, 31, 0, 0, 0, 22, 20, 0, 0, 0, 5, 0, 0;
16, 0, 0, 5, 7, 0, 30, 19, 0, 4, 5, 0, 0, 5, 0, 0, 0;
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;
0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 5, 0, 0, 0;
0, 0, 0, 1, 4, 1, 0, 0, 0, 24, 16, 4, 0, 0, 0, 0, 0;
0, 3, 0, 3, 7, 3, 0, 4, 27, 34, 34, 16, 4, 4, 16, 0, 0;
0, 0, 5, 5, 5, 5, 0, 0, 0, 0, 33, 33, 33, 33, 0, 0, 0;
0, 0, 4, 4, 4, 4, 0, 0, 0, 0, 32, 32, 32, 32, 0, 0, 0;
0, 4, 4, 0, 0, 2, 2, 0, 0, 31, 31, 0, 0, 31, 31, 0, 0;
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

```

Figura 5. Matriz de elementos para las imágenes correspondientes a los elementos RGB de nuestra imagen original.