

Nombre: Braulio Castro

Codigo :

Repositorio= <https://github.com/braulio1996/ExamenFinalIA2>

```
int * Historgrama(Mat imagen) {
    //histograma de la imagen
    int * histo = new int[256];
    for (int i = 0; i < 256; i++)
        histo[i] = 0;
    Vec3b pixel;
    for (int i = 0; i < imagen.rows; i++) {
        for (int j = 0; j < imagen.cols; j++) {
            pixel = imagen.at<Vec3b>(i, j);
            histo[pixel[0]]++;
        }
    }
    return histo;
}

double dEuclidea(int m1[256], int m2[256]) {
    double suma = 0.0;
    for (int i = 0; i < 7; i++) {
        suma += (m1[i] - m2[i]) * (m1[i] - m2[i]);
    }
    return sqrt(suma);
}

void comparar() {
    int* histoImg [2];
    Mat imagen [2];
    double distancia = 0.0;
    double distanciaMenor = 9999999999999999.9;
    string numero;
    string nombre = ".\\CORPUS-IMAGENES-1\\image_00";
    for (int i = 1; i < 57; i++) {
        numero = i < 10 ? "0" + to_string(i) : to_string(i);
        cout << nombre + numero + ".jpg" << endl;
```

```
47 void comparar() {
48     int* histoImg [2];
49     Mat imagen [2];
50     double distancia = 0.0;
51     double distanciaMenor = 9999999999999999.9;
52     string numero;
53     string nombre = ".\\CORPUS-IMAGENES-1\\image_00";
54     for (int i = 1; i < 57; i++) {
55         numero = i < 10 ? "0" + to_string(i) : to_string(i);
56         cout << nombre + numero + ".jpg" << endl;
57         if (i % 2 == 0) {
58             imagen[1] = imread(nombre + numero + ".jpg");
59             histoImg[0] = Historgrama(imagen[0]);
60             histoImg[1] = Historgrama(imagen[1]);
61             distancia = dEuclidea(histoImg[0], histoImg[1]);
62             if (distancia < distanciaMenor)
63                 distanciaMenor = distancia;
64             cout << distancia << endl;
65         }
66         else {
67             imagen[0] = imread(nombre + numero + ".jpg");
68         }
69     }
70     cout << "Distancia Azul" << distanciaMenor << endl;
71 }
72 int main (int argc, char *argv[]){
73     comparar();
74 }
75
76
77
```

```
#include <opencv2/videoio/videoio.hpp>

using namespace std;
using namespace cv; // Espacio de nombres de OpenCV (Contiene funciones y definiciones de varios elementos de OpenCV)

int main (int argc, char *argv[]){
    Mat img = imread("image_0033.jpg", IMREAD_GRAYSCALE);
    int pixel;
    double promedio;
    for(int i=0; i < img.rows; i++){
        for(int j=0; j < img.cols; j++){
            pixel = img.at<uchar>(i,j);
            promedio+=pixel;
        }
    }
    cout <<"Intensidad GRIS : "<< promedio/(img.rows*img.cols) << endl;

    uchar imagen[4][4] = {{123,97,10,91},{89,47,19,101},{7,3,250,20},{13,1,0,200}};
    Mat mat(4,4,CV_8UC1,&imagen);
    Mat binaria;
    threshold(mat,binaria,40,1,THRESH_BINARY);
    Moments momentos = moments(binaria,true);
    cout <<"MOMENTO m01= " << momentos.m01 << endl;

}
```

Resultados:

```
Distancia Azul 10.72
braulio@brau:~/IA2$
```

```
braulio@brau:~/IA2/CORPUS-IMAGENES-2$ make run
./vision.bin
Intensidad GRIS :-2.26016e+280
MOMENTO m01= 8
braulio@brau:~/IA2/CORPUS-IMAGENES-2$
```