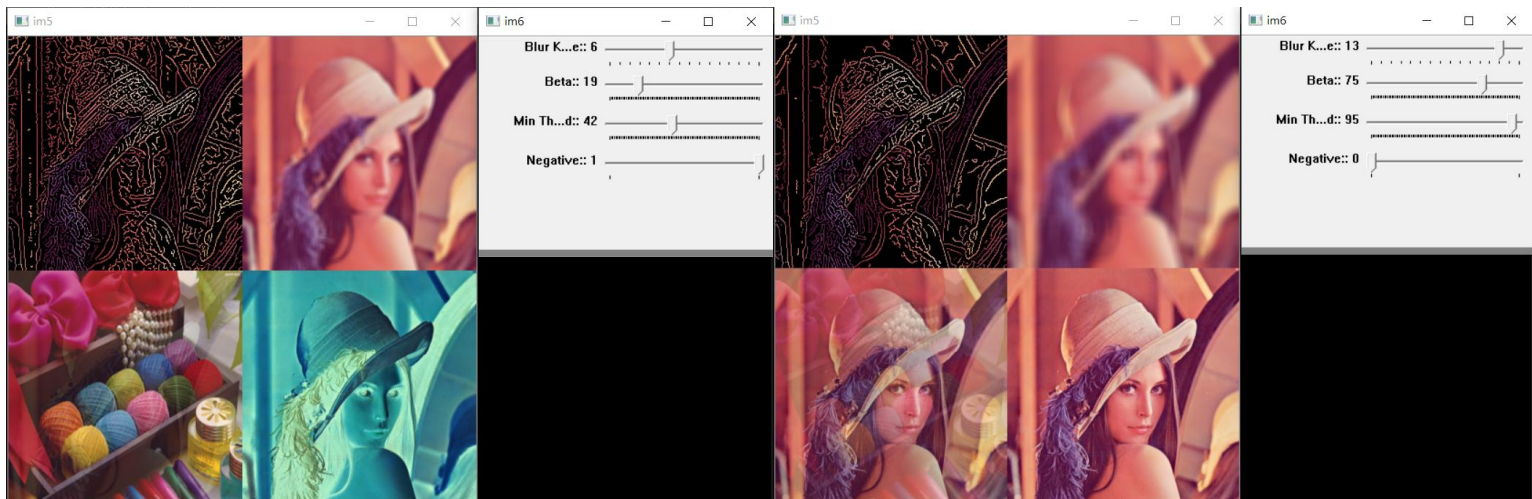
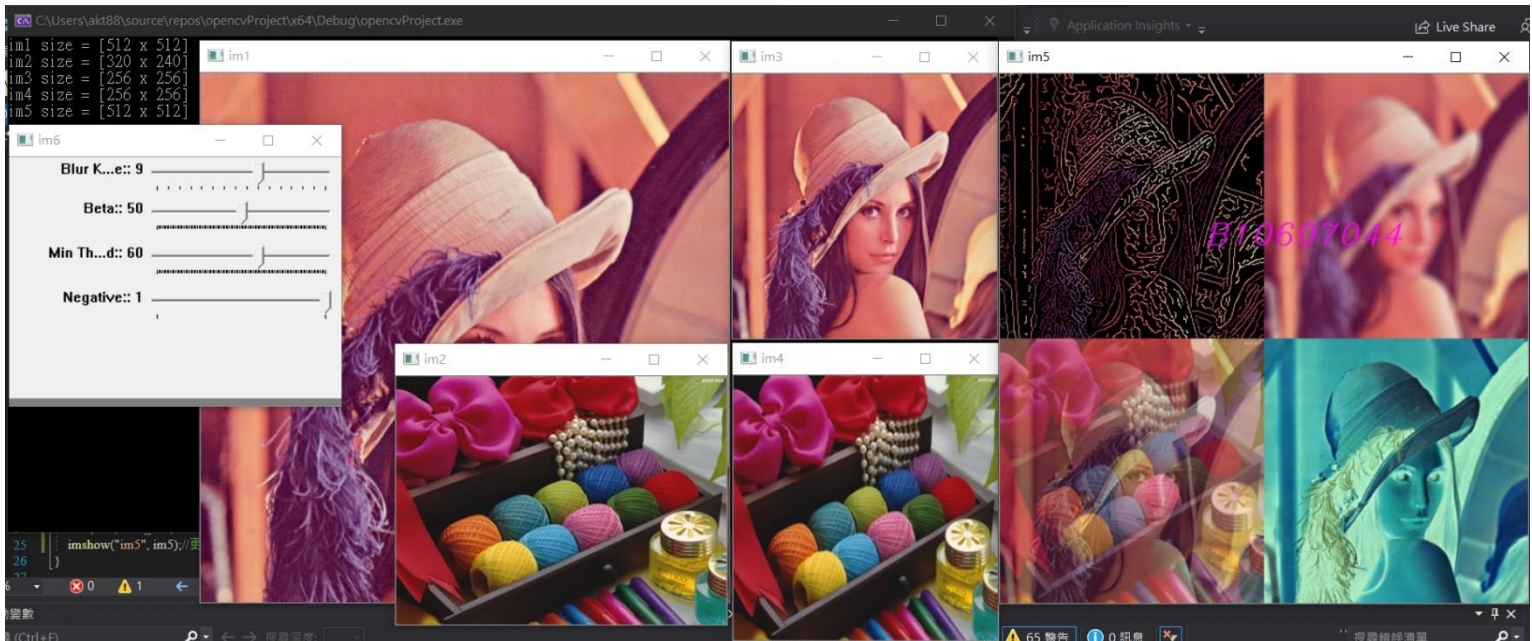


台科大 109 學年度 色彩及影像實作技術

作業三： OpenCV 基本影像處理

B10607044 潘乃聿



程式碼註解

```
#include<iostream>

#include "opencv2/opencv.hpp"
#include <opencv2/highgui/highgui.hpp>
#include <opencv2/imgproc/imgproc.hpp>

using namespace std;
using namespace cv;

Mat im3;
Mat im4;
Mat im5;
Mat ROI_LU;//Canny處理
Mat ROI_RU;//Blur處理
Mat ROI_LD;//權重合成處理
Mat ROI_RD;//反白
bool textflag = 0;//判斷是否有上浮水印 有才能結束
int blur_ksize = 3;//Blur處理初始Kernel Size
int Beta = 50;//權重合成處理初始權重值50%
int lowThreshold = 40;//Canny處理初始低閾值
int negative=1;//反白Flag

static void BlurImg(int, void*) {//Blur處理
    textflag = 0;//有改就會把浮水印用掉
    if (blur_ksize < 1) blur_ksize = 1;//保護不能為0
    blur(im3, ROI_RU, Size(blur_ksize, blur_ksize));//Blur處理
    imshow("im5", im5);//更改後成像
}

static void AddWeight(int, void*) {//權重合成處理
    textflag = 0;//有改就會把浮水印用掉
    addWeighted(im3, (double)Beta/100, im4, 1 - (double)Beta/100, 0.0, ROI_LD);//權重合成處理
    imshow("im5", im5);//更改後成像
}

static void Negative(int, void*) {//反白
    textflag = 0;//有改就會把浮水印用掉
    if (negative) bitwise_not(im3, ROI_RD);//Flag開啟為反白影像
    else im3.copyTo(ROI_RD);//Flag關閉為正常影像
```

```

        imshow("im5", im5); // 更改後成像
    }

static void CannyThreshold(int, void*) // Canny處理
{
    textflag = 0; // 有改就會把浮水印用掉
    Mat detected_edges, im3_gray;
    cvtColor(im3, im3_gray, COLOR_BGR2GRAY); // 將影像改成灰階
    blur(im3_gray, detected_edges, Size(3, 3)); // 降噪
    Canny(detected_edges, detected_edges, lowThreshold, lowThreshold * 3, 5); // Canny檢測
    ROI_LU = Scalar::all(0);
    im3.copyTo(ROI_LU, detected_edges); // 利用檢測結果當遮罩
    imshow("im5", im5); // 更改後成像
}

static void onMouse(int event, int x, int y, int flags, void*) // 滑鼠事件
{
    switch (event) {

        case EVENT_LBUTTONDOWN: // 左鍵按下
            Mat im5_temp;
            im5.copyTo(im5_temp); // 複製im5
            putText(im5_temp, "B10607044", Point(x, y), FONT_HERSHEY_COMPLEX |
FONT_ITALIC, 1, CV_RGB(255, 0, 255), 1, LINE_AA); // 寫學號
            imshow("im5", im5_temp); // 有學號影像
            imwrite("B10607044.jpg", im5_temp); // 寫入檔案
            textflag = 1; // 確認有寫學號
            break;
        }
    }

int main(int, char** argv)
{
    Mat im1 = imread(argv[1], 1); // 讀影像1
    Mat im2 = imread(argv[2], 1); // 讀影像2

    resize(im1, im3, Size(), 0.5, 0.5); // 將im3等於im1高寬各一半影像

```

```
resize(im2, im4, im3.size(), 1, 1); //將im4等於im2但高寬等於im3的影像
```

```
im5.create(im1.size[0], im1.size[1], CV_8UC3); //建立im5高寬跟im1一樣
```

```
ROI_LU = im5(Rect(0, 0, im1.cols / 2, im1.rows / 2)); //im5左上部分(Canny)
```

```
ROI_RU = im5(Rect(im1.cols / 2, 0, im1.cols / 2, im1.rows / 2)); //im5右上部分(Blur)
```

```
ROI_LD = im5(Rect(0, im1.rows / 2, im1.cols / 2, im1.rows / 2)); //im5左下部分(AddWeight)
```

```
ROI_RD = im5(Rect(im1.cols / 2, im1.rows / 2, im1.cols / 2, im1.rows / 2)); //im5右下部分(Negative)
```

```
namedWindow("im1"); //建立視窗
```

```
moveWindow("im1", 100, 100); //移動視窗
```

```
imshow("im1", im1); //成像
```

```
namedWindow("im2"); //建立視窗
```

```
moveWindow("im2", 120, 250); //移動視窗
```

```
imshow("im2", im2); //成像
```

```
namedWindow("im3"); //建立視窗
```

```
moveWindow("im3", 420, 100); //移動視窗
```

```
imshow("im3", im3); //成像
```

```
namedWindow("im4"); //建立視窗
```

```
moveWindow("im4", 500, 250); //移動視窗
```

```
imshow("im4", im4); //成像
```

```
namedWindow("im5"); //建立視窗(4格視窗)
```

```
namedWindow("im6"); //建立視窗(參數視窗)
```

```
moveWindow("im5", 800, 200); //移動視窗
```

```
createTrackbar("Blur Kernel Size:", "im6", &blur_ksize, 15, BlurImg); //建立Blur Kernel Size的
```

Trackbar

```
createTrackbar("Beta:", "im6", &Beta, 100, AddWeight); //建立Beta的Trackbar
```

```
createTrackbar("Min Threshold:", "im6", &lowThreshold, 100, CannyThreshold); //建立Min
```

Threshold的Trackbar

```
createTrackbar("Negative:", "im6", &negative, 1, Negative); //建立Negative Flag的Trackbar
```

```
BlurImg(0, 0); //執行Blur處理
```

```
AddWeight(0, 0); //執行AddWeight處理
```

```
CannyThreshold(0, 0); //執行Canny處理
```

```
Negative(0, 0); //執行Negative處理
```

```
moveWindow("im5", 700, 150); //移動視窗
```

```
imshow("im5", im5); //成像4格影像
```

```
setMouseCallback("im5", onMouse); //讀滑鼠事件
```

```
cout << "im1 size = " << im1.size() << endl; //輸出im1影像大小
cout << "im2 size = " << im2.size() << endl; //輸出im2影像大小
cout << "im3 size = " << im3.size() << endl; //輸出im3影像大小
cout << "im4 size = " << im4.size() << endl; //輸出im4影像大小
cout << "im5 size = " << im5.size() << endl; //輸出im5影像大小
while (1) { //確認條件
    if (waitKey() == 27 && textflag == 1) break; //確定有浮水印以及按ESC結束程式
}

destroyAllWindows(); //關閉所有視窗
return 0;
}
```