高等電腦視覺

作業#(01)

姓名： 吳佩霖

學號： 111c52016

指導老師： 張陽郎

|  |
| --- |
| **程式執行說明** |
| 1.  首先選擇要執行C++版本或openCV版本 |
| 2.  選擇要執行的題號 |

|  |
| --- |
| 3.  執行完成後會看到”file has been saved 檔名.bmp successful!”的提示，檔案則儲存於HW1的資料夾中。 |
| 4.  若是執行openCV的版本則是會跳出結果的視窗，並如同C++版本一樣有”file has been saved 檔名.bmp successful!”的提示，即為儲存完成。 |

|  |  |
| --- | --- |
| 1 | |
| C++ | OpenCV |
| HW1\_C++\_1.bmp | HW1opencv\_1.bmp |
|  |  |
| Discussion \_1 | |
| 由於bmp有標頭檔在讀取上會不如raw的直觀簡單，因此讀取的時候需要特別留意標頭與三通道的的部分。 在openCv則沒有這個問題，直接使用imread跟imwrite就可以直接讀寫。 | |

|  |  |
| --- | --- |
| 2 | |
| C++ | OpenCV |
| HW1\_C++\_2.bmp | HW1opencv\_2.bmp |
|  |  |
| Discussion \_2 | |
| 負片的做法基本上就是由255扣掉數值，在這邊C++版本有三通道的關係，因此需要扣除3次。  OpenCV的部分直接使用bitwise\_not進行運算，即可得出結果。 | |

|  |  |
| --- | --- |
| **3** | |
| C++ | OpenCV |
| HW1\_C++\_3.bmp | HW1opencv\_3.bmp |
|  |  |
| Discussion \_3 | |
| C++版本的旋轉需要經過計算，在取分割內容的部分則跟openCV 版本無太大差異。  openCV版本的旋轉可由cv::rotate完成，之後按照想要的分割內容格是進行取值即完成。 | |

|  |  |
| --- | --- |
| **Bonus1&2** | |
| C++ | OpenCV |
| HW1\_C++\_bonus1.bmp  (1024\*1024) | HW1opencv\_bonus1.bmp  (1024\*1024) |
|  |  |
| HW1\_C++\_bonus2.bmp  (256\*256) | HW1opencv\_bonus2.bmp  (256\*256) |
|  |  |
| Discussion \_3 | |
| C++使用了nearest\_neighbor進行了放大縮小，在並排上以肉眼觀察openCV是使用線性插值，輸出的圖片多少有點不同，但差異並無太大。 | |

|  |  |
| --- | --- |
| **Bonus3&4** | |
| C++ | OpenCV |
| HW1\_C++\_bonus3.bmp  (1024\*1024) | HW1opencv\_bonus3.bmp  (1024\*1024) |
|  |  |
| HW1\_C++\_bonus4.bmp  (256\*256) | HW1opencv\_bonus4.bmp  (256\*256) |
|  |  |
| Discussion \_3 | |
| 一樣使用前面題目所得到的結果進行放大與縮小。 | |