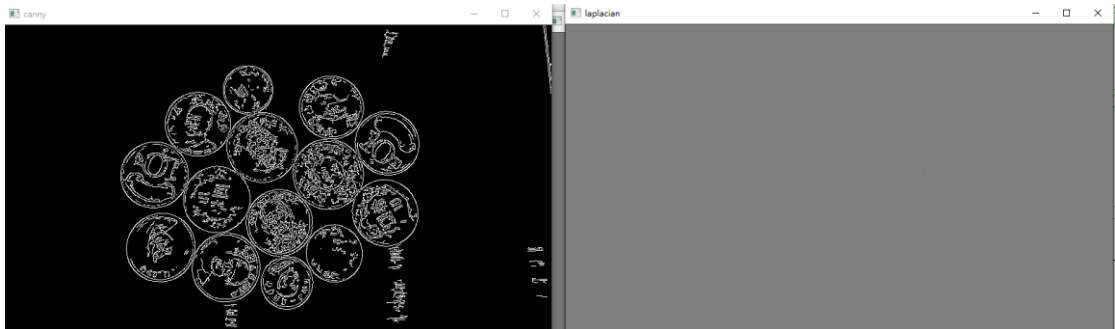


- `cv2.connectedComponents(image[, connectivity[, ltype]])`：請測試各項參數的影響並寫下心得。
 - `num_labels` 有多少區塊
 - `labels` 透過矩陣表示區塊
 - `stats` 是一個陣列，依序是 `x` 座標、`y` 座標、長、寬、面積，面積以**像素**為單位
 - `centroids` 區塊的中心點

在研究此語法中，發現一個最重要的問題是 `stats` 的面積是以像素來計算面積的，如果區塊中間沒有連通的話，`stats` 的面積是會扣除她的！因此在小組題目一辨識硬幣中不可以直接用 `stats[4]`，會因為中間的雜訊而導致此區塊的面積減少，而造成我們沒有辦法順利地用大小來判斷硬幣種類。

- `Sobel,Scharr,Laplacian,Canny`：說明觀察到四種邊緣檢測方法之間的差異 以及測試各項參數的影響並寫下心得。

`Canny` 與 `Scharr` 在測試中我認為效果最佳，另外兩個邊緣檢測方法在某些特定圖片中有可能檢測不出來圖片，變成一片灰色的情景



左邊是 `Canny`、右邊是 `Laplacian`，圖片是小組專題一的圖片而 `Scharr` 會有過多的雜訊產生，我個人在使用時會比較喜歡用 `Canny`，他給我的使用感覺最好。

- 霍夫圓、霍夫線：請自行找兩張圖片分別包含直線與圓，測試各項參數的影響並寫下心得。

我有得出一個心得，似乎霍夫線沒有辦法去判斷**斜線**，也有可能是我研究的不夠精深，我使用第一周作業老師讓我們學習如何用 `opencv` 畫圓、正方形、直線時，我有畫一條直線與斜線，但霍夫線指能夠順利偵測到直線的部分，斜線我上網並嘗試一陣子卻都沒有辦法成功讓霍夫線順利的辨識到斜線。

霍夫圓就順利很多，根據老師 `ppt` 打的語法介紹再加上我個人稍微修改參數設定後就順利畫出，下面是畫出的圖片

特別需要注意的是，霍夫線與霍夫圓都是以白色為基礎去判斷線與圓型，

如果二值化後你想要判斷的地方是黑色時，記得在二值化的判斷方法中改用 `cv2.THRESH_BINARY_INV`，此函數就能順利將黑變白、白變黑。

