

影像視訊處理實驗作業結報

班 級	電機	學 號	M11107S08	姓 名	王維澤
實驗題目	實習五-影像特徵的研究				

實驗內容

*請勿抄襲，否則視為未交

一、 實驗簡介(至少 200 字)：

利用影像特徵進行自動識別，輪廓搜尋、形狀特徵、輪廓特徵等等。
以物體特徵為基礎的追蹤方法，是萃取影像中形成目標物的各種成分，再將這些成分集成較高階的特徵資訊，藉由比對連續影像間的特徵資訊來追蹤目標物。根據不同的特徵，可粗分為：整體性特徵 (global feature-based)，如重心、色彩、面積等；局部性特徵 (local feature-based)，如線段、頂點等；相依圖形特徵 (dependence-graph-based)，如特徵間的結構變化等。此外，像是人，利用紋路、顏色、邊界等資訊建立追蹤物體的特徵模型。

二、 實驗動機及其解決方法(至少 500 字)：

1. 二值影像的特徵參數

面積：計算物體所包含的像素數量。

周長：可計算出物體輪廓線上的像素數，但沿著斜線方向。

真圓度

重心

2. 影像的標籤化，標籤化的處理步驟：

- (1). 先掃描影像，如果發現沒有附加標籤的像素 P，就附加新的標籤號碼。
- (2). 和像素 P 連接的像素，附加相同的標籤號碼。
- (3). 對已附加標籤號碼者，將與其連接之所有像素，附加同樣的標籤號碼。
- (4). 持續以上的操作，一直到再也沒有應當附加標籤的像素為止。這時，單一連接成分中的所有像素應當都已附加上相同的標籤號碼。
- (5). 返回到步驟 1 處，如果還可找到沒有附加標籤號碼的像素，就重複 2~4 處理步驟，貼上新標籤號碼。

3. 利用特徵參數分割影像



4. 利用特徵參數去除雜訊

三、 程式碼(須附註解說明，截圖即可)：

二值化(Image Binarization)

(1). 先對目標影像進行二值化，Thresholding 設為 55

標籤化(Labeling)並計算特徵參數，再由特徵參數分割影像:

- (1). 影像標籤化並計算特徵參數
- (2). 將面積小於 100 的標籤連通元件去除
- (3). 將剩餘的標籤連通元件透過輪廓偵測計算周長和真圓度
- (4). 將真圓度小於 0.5 的標籤連通元件去除，為了留下圓形的物體

```
1 def Image_Labeling(src, src_gray, src_image_binary):
2     kernel2 = cv.getStructuringElement(cv.MORPH_RECT, (3, 3))
3     bin_clo = cv.dilate(src_image_binary, kernel2, iterations=2)
4     analysis = cv.connectedComponentsWithStats(
5         bin_clo,
6         connectivity=8,
7         ltype=cv.CV_32S
8     )
9     num_labels, labels, stats, centers = analysis
10
11     src_image_labeling = np.copy(src)
12     src_h, src_w = src_image_binary.shape
13     src_image_contour = np.zeros((src_h, src_w, 1), dtype = np.uint8)
14     src_image_segmenation = np.zeros((src_h, src_w, 1), dtype = np.uint8)
15
16     for i in range(1, num_labels):
17         x, y, w, h, area = stats[i]
18         cx, cy = centers[i]
19         if (area > 100):
20             for row in range(src_h):
21                 for col in range(src_w):
22                     if (labels[row, col] == i):
23                         src_image_contour[row, col] = 255
24             contours, hierarchy = cv.findContours(
25                 src_image_contour,
26                 cv.RETR_EXTERNAL,
27                 cv.CHAIN_APPROX_SIMPLE
28             )
29             src_image_labeling = cv.drawContours(
30                 src_image_labeling,
31                 contours,
32                 -1,
33                 (255, 0, 0),
34                 1
35             )
36
37     cnt = contours[0]
38     perimeter = cv.arcLength(cnt, True)
39
40     e = 4 * math.pi * (1 / perimeter * perimeter)
41
42     cv.circle(src_image_labeling, (int(cx), int(cy)), 2, (0, 255, 0), 2, 8, 0)
43     cv.rectangle(src_image_labeling, (x, y), (x+w, y+h), (0, 0, 255), 1, 8, 0)
44     cv.putText(
45         src_image_labeling,
46         "No. " + str(i),
47         (x, y-10),
48         cv.FONT_HERSHEY_SIMPLEX,
49         .5,
50         (0, 0, 255),
51         1
52     )
53     print("No. " + str(i) + " 周長: %d, 面積: %d, 圓度: %f" % (perimeter, area, e))
54
55     if (e < 0.5):
56         continue
57     else:
58         for row in range(src_h):
59             for col in range(src_w):
60                 if (labels[row, col] == i):
61                     src_image_segmenation[row, col] = 255
62     return src_image_contour, src_image_segmenation, src_image_labeling
63
```

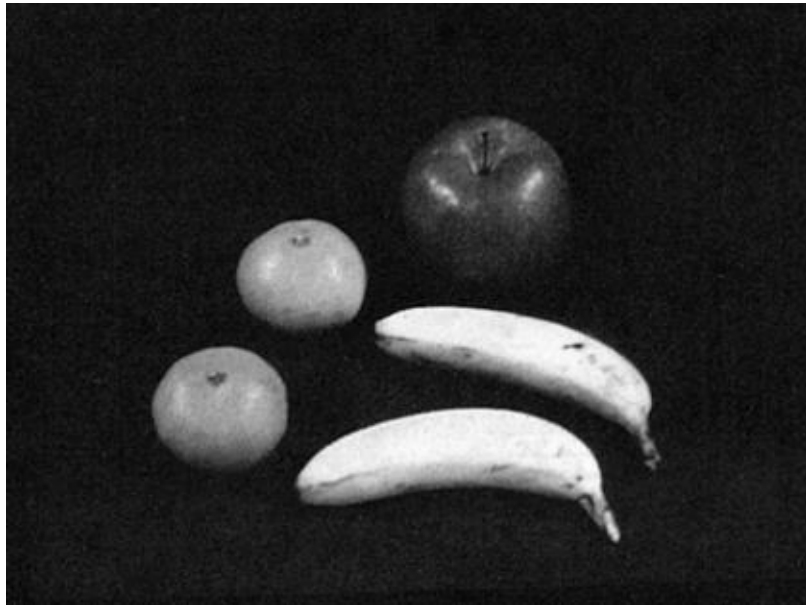
將圓度小於 0.5 的標籤連通元件去除，為了留下圓形的物體，最後返回

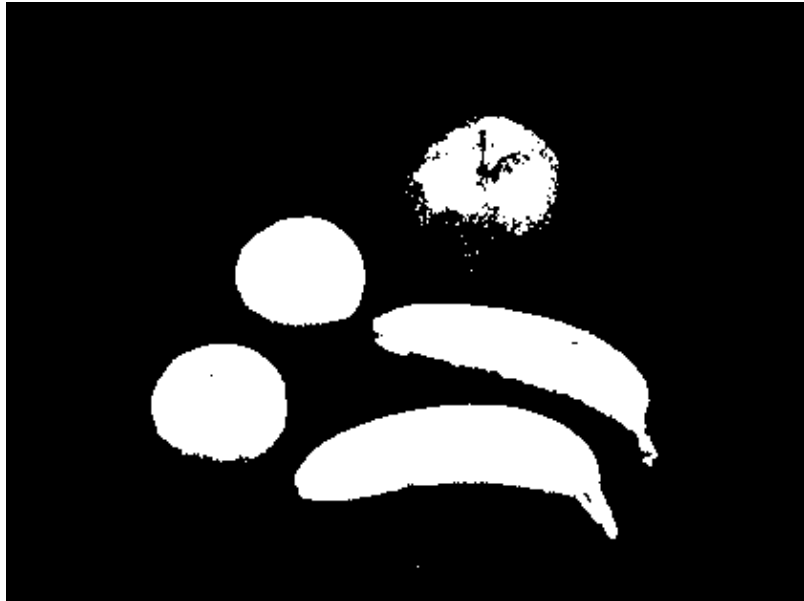
```
1 src_image_binary = Image_Binary(img_gray)
2 src_image_contour, src_image_segmenation, src_image_labeling = Image_Labeling(
3     img,
4     img_gray,
5     src_image_binary
6 )
7 cv_imshow(img)
8 cv_imshow(src_image_binary)
9 cv_imshow(src_image_contour)
10 cv_imshow(src_image_labeling)
```

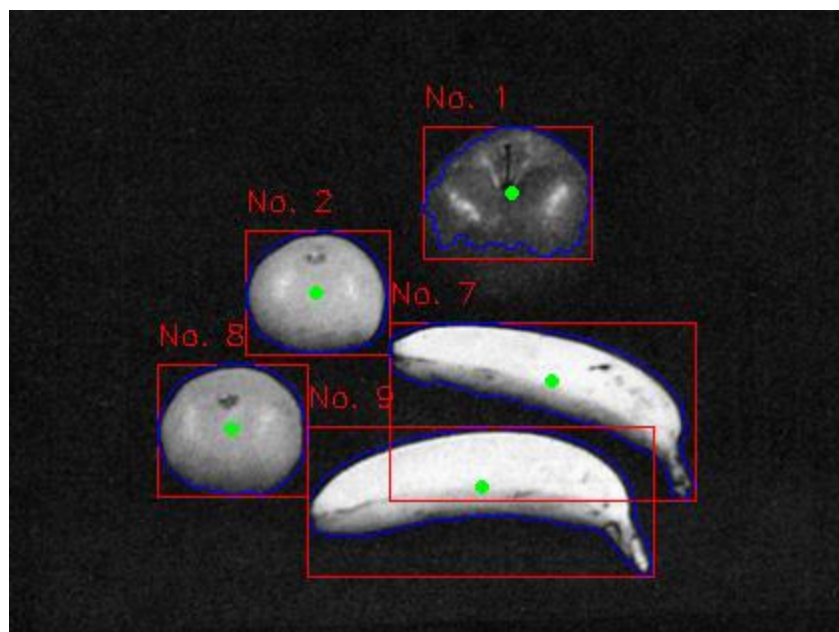
主函式

四、 實驗結果(貼圖與簡述說明)：

```
➤ No. 1 周長: 288, 面積: 4148, 圓度: 12.566371
  No. 2 周長: 224, 面積: 3608, 圓度: 12.566371
  No. 7 周長: 420, 面積: 5724, 圓度: 12.566371
  No. 8 周長: 242, 面積: 4038, 圓度: 12.566371
  No. 9 周長: 445, 面積: 7057, 圓度: 12.566371
```







(可自行增頁)

*內容字級為 12 字級、中文為標楷體、英文為 Times New Roman