

# Lab 11 Feature Matching

無人機自動飛航與電腦視覺概論

第4組／游騰德、鄭程哲、唐宇謙

---

## 1. 介紹SIFT、SURF、ORB

- **SIFT**

Scale-invariant feature transform, 尺度不變特徵轉換

SIFT由Lowe提出，可以聚焦在影像的區域性特徵，並且在旋轉、尺度縮放、亮度變化保持不變性，也可以對視角變換、仿射轉換、噪點保持一定的穩度性。

SIFT的流程主要可以分為以下幾點：

1. 檢測尺度空間的極值：根據不同尺度下的高斯模糊化影像差異（DoG），將每一個sample出來的點和它鄰接的每個點比較，找出局部極值。這些找到的極值對應的點稱為特徵點或關鍵點。
2. 精確定位關鍵點：因為可能找出過多的關鍵點，其中部分的關鍵點易受雜訊干擾（可能是對比度太低或被DoG產生的過量edge response干擾），因此這步會依靠關鍵點附近的像素資訊做定位，並去除干擾太嚴重的點。
3. 設定關鍵點的方向參數：利用關鍵點相鄰像素的梯度方向分布特性，計算方向分布直方圖，尋找直方圖中最大的方向做為關鍵點的主要方向。
4. 生成關鍵點的描述符：到這裡，關鍵點已經檢測完畢，每個關鍵點都帶有三項資訊：位置、所處尺度、方向。此外，還需要對關鍵點建立一個描述符向量，使其在不同光線及視角下都能維持穩定不變。這個描述符是關鍵點的鄰接點高斯影像梯度統計結果的一種表示向量，透過將鄰近影像區域分塊，計算各個分塊內的梯度直方圖來獲得。

- **SURF**

Speeded Up Robust Features, 加速穩健特徵

SURF是在SIFT發表的幾年後所提出，由Herbert Bay、Tinne Tuytelaars、Luc Van Gool於2006年的ECCV大會上共同發表，因為SIFT最主要的缺點就是計算

速度太慢，如果不藉由更高效運算的硬體(如:GPU)加速，是很難達到實時運算的。而SURF就如同它的名字所示，是一個比SIFT更快速的演算法，雖然改進後的速度沒有跳躍性的縮減，但仍是一個重要的進步。

在Feature Detection的步驟中，SURF使用了比SIFT的DoG計算上來得更快速更簡單的Hessian-Laplacian，稱為Fast-Hessian；SURF使用box filters取代原先SIFT的Gaussian filters，其中積分圖像的方法可以讓原先複雜的計算變成單純的加減運算。

在定位關鍵點的部分與SIFT大致相同；而在方向分配上，SURF透過計算關鍵點附近圓形區域內所有60度扇形的Haar wavelet responses取得向量，而在此之中最長的向量即為關鍵點的主要方向。

設定好關鍵點的方向後，在其周圍 $20 \times 20$  pixel範圍中劃分16個子區域(每個子區域為 $5 \times 5$  pixel)，分別計算各個子區域內的x, y方向的Haar wavelet responses與其向量長度總和共四個值，此處的x方向是和關鍵點平行的方向，而y方向是和關鍵點垂直的方向，透過計算後可以得到總共  $4 \times 16 = 64$  個descriptor，是SIFT數量的一半。

SURF雖然號稱Feature matching的結果不會比SIFT差太多，但實際上我們看一些網站與文章，SIFT偵測更多的特徵，表現的結果也比較好；不過SURF計算上的速度大概比SIFT快上3倍，是一個已經很不錯的效能了！

- **ORB**

Oriented FAST and Rotated BRIEF，尺度不變特徵變換

ORB由Ethan Rublee、Vincent Rabaud、Kurt Konolige和Gary R. Bradski於2011年在OpenCV實驗室開發，是一種替代SURF和SIFT的算法，ORB在特徵檢測上的表現與SIFT相同，並且優於SURF，同時速度提高了近兩個數量級。

ORB算法使用FAST算法尋找關鍵點，然後使用Harris角點檢測找到這些點當中的最好的N個點，並且計算了以當前角點為中心所在區域的強度加權質心來獲得旋轉不變性。

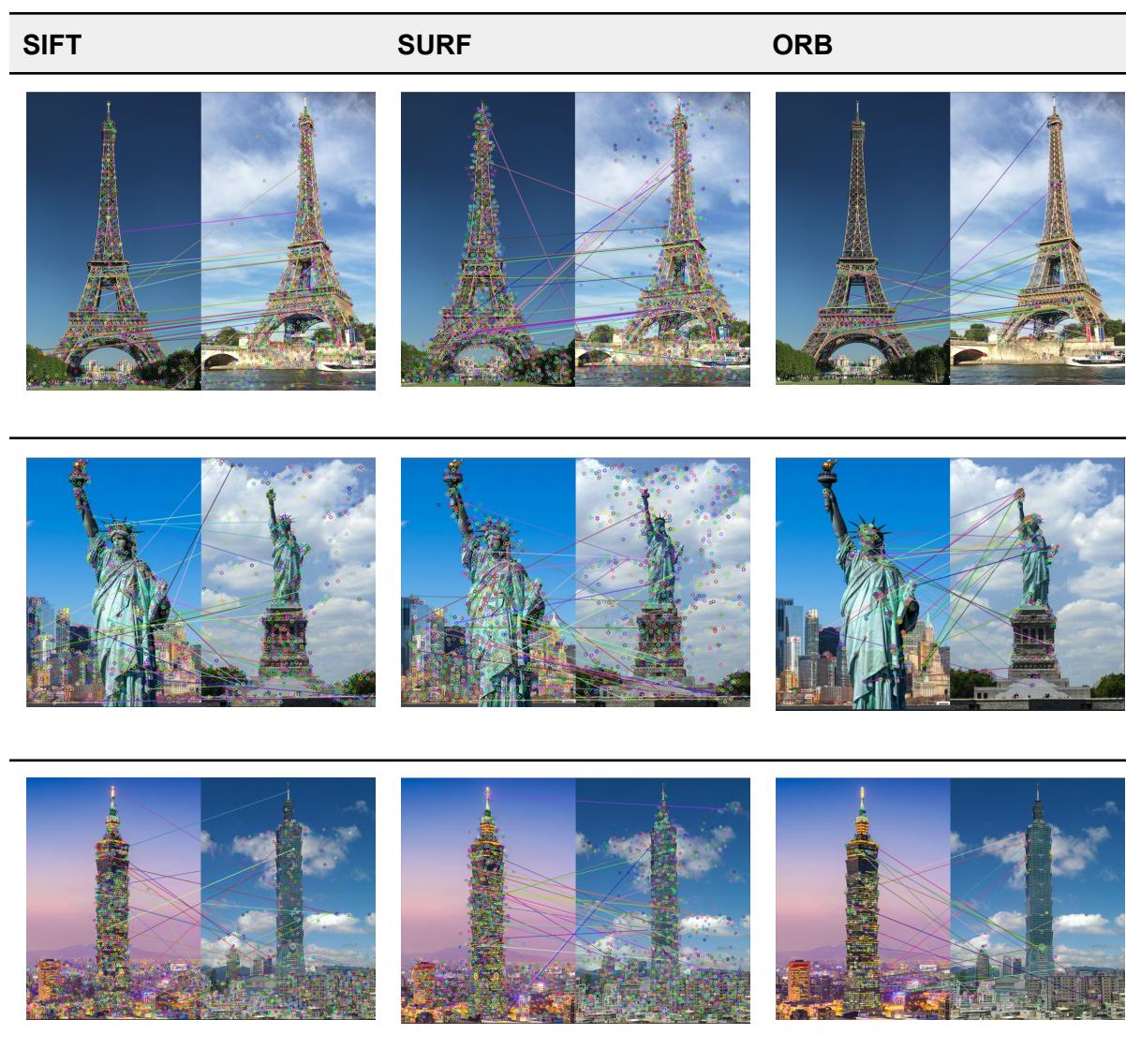
因此，在找到關鍵點後，ORB會每個關鍵點指定一個方向，指定的方向取決於強度變化的程度。為了檢測強度變化，ORB使用強度質心，假設角的強度偏離其中心，該向量可用於估算方向。

BRIEF採用FAST算法找到的所有關鍵點並將其轉換為二進位特徵向量，來描述一個對象。每個關鍵點由特徵向量描述，該特徵向量是128-512位字串。然而，BRIEF在旋轉的圖像下表現不理想，因此，ORB提出了一種根據關鍵點的方向引導BRIEF的方法，稱為rBRIEF。

rBRIEF針對所有訓練都做測試，按照它們與平均值0.5的距離對測試進行排序，形成向量，然後進行Greedy search，在所有可能的二進制測試當中去找到的一個平方差值高並且均值接近0.5的特徵點。

## 2. 實驗結果

這次lab我們找了三種圖片來做，分別是艾非爾鐵塔、自由女神像、台北101，以下是SIFT、SURF、ORB比對出來的結果：



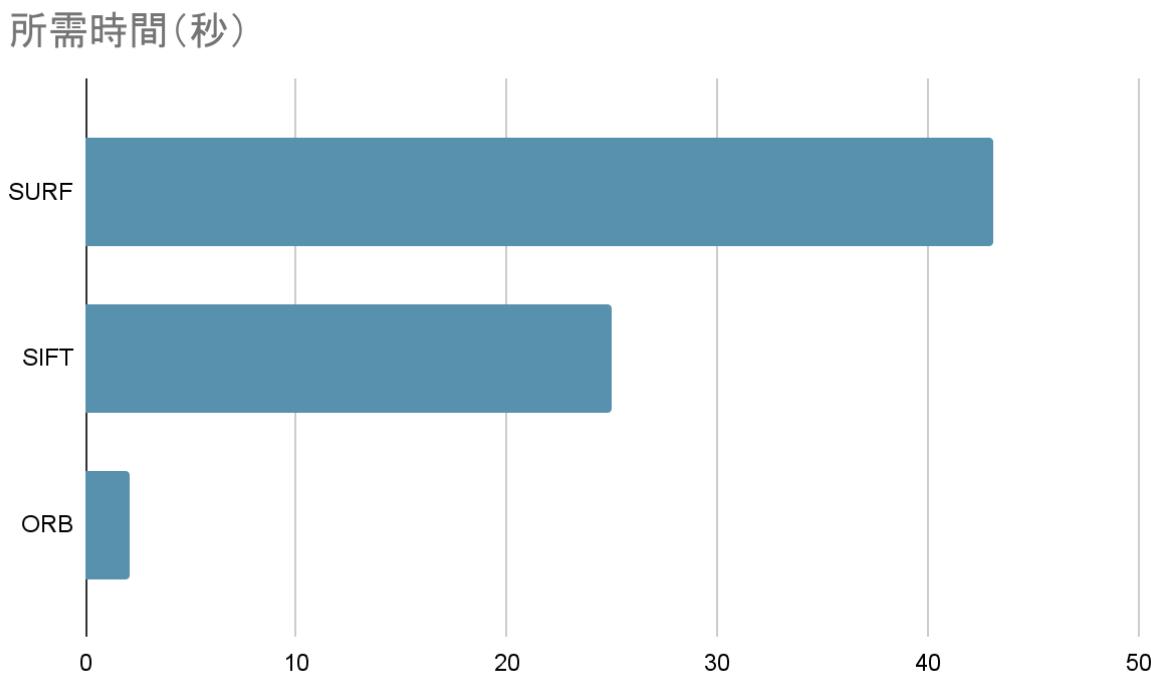
### 3. 分析與結論

- 準確率

在這次lab中根據我們做出的結果，特徵點比對準確率是SURF > ORB  $\approx$  SIFT。雖然SURF做自由女神那張的準確率很低，但SURF另外兩張的效果都較SIFT和ORB更佳，能夠找出更多正確對應的feature，因此把他放在準確率最高。

- 速度

我們設計了一個實驗，讓3種方法辨識上面巴黎鐵塔的兩張照片各100次，觀察各個方法所需秒數時間，結果為SURF > SIFT > ORB，詳細結果如下：



- 綜合比較與結論

從查到的資料上來說，速度應該會是ORB > SURF > SIFT，可能是因為opencv底層實做方式不同的關係導致我們測出來的速度不一樣。根據我們的結果，ORB擁有最快的速度以及相對來說不差的準確率，對以後想要做其他feature matching的實驗來說會是一個相對不錯的選擇。不過還要根據拿到的資料來選

擇一下，因為ORB沒有尺度不變性。如果需要高速的計算且場景變化不會太大的話，就可以考慮使用。