Embedded Vision Intelligent Laboratory

多媒體技術與應用
Spring 2021

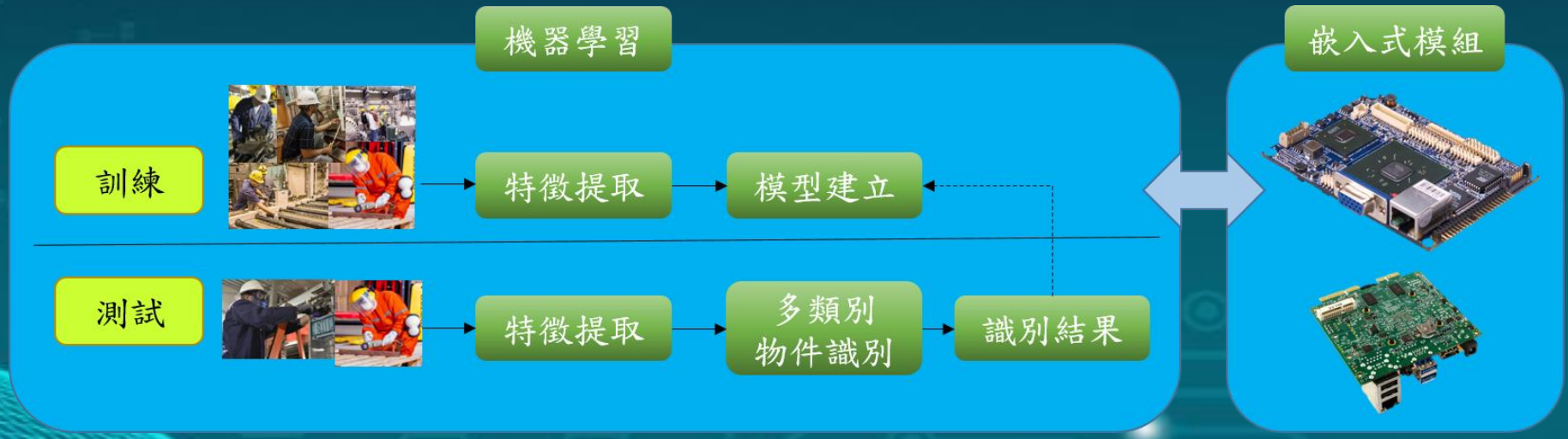
Instructor： Yen-Lin Chen(陳彥霖), Ph.D.
Professor
Dept. Computer Science and Information Engineering
National Taipei University of Technology

Lecture 5

機器學習與HOG特徵描述器

什麼是機器學習

- 傳統機器學習(特徵分類器)之優點與限制
  - 優點：對於少量的訓練集而言，比深度學習更容易訓練出好的模型，參數量少且易於分析，因此易於調整參數與線上更新訓練模型。
  - 限制：偵測效能非常依賴事先設計特徵之好壞，選擇最佳特徵並對資料進行充分降維，才可獲得有效之分類器。



影像特徵提取

- 影像特徵可以包括顏色特徵、紋理特徵、形狀特徵以及局部特徵點等。由於一幅影像中，對電腦來說只是許多的像素點所構成，所以我們必須根據這些數據提取出影像中的關鍵資訊，例如梯度變化或輪廓等資訊，接著可將這些資訊作為機器學習分類器(如SVM)對影像進行分類的依據。

HOG Feature

HOG(Histogram of gradients)-簡介

- 方向梯度直方圖 (HOG)為Dalad和Triggs在2005的CVPR上提出的一種在電腦視覺和影像處理中用來進行物體檢測的特徵描述子。
- HOG能取得影像中的梯度資訊做為特徵描述的方式，所以善於描述物件的輪廓。
- HOG針對局部向量進行正規化，可降低光影變化造成的影響。
- HOG是由一個影像中一連串局部區塊的方向梯度直方圖所組成。

HOG(Histogram of gradients)-簡介

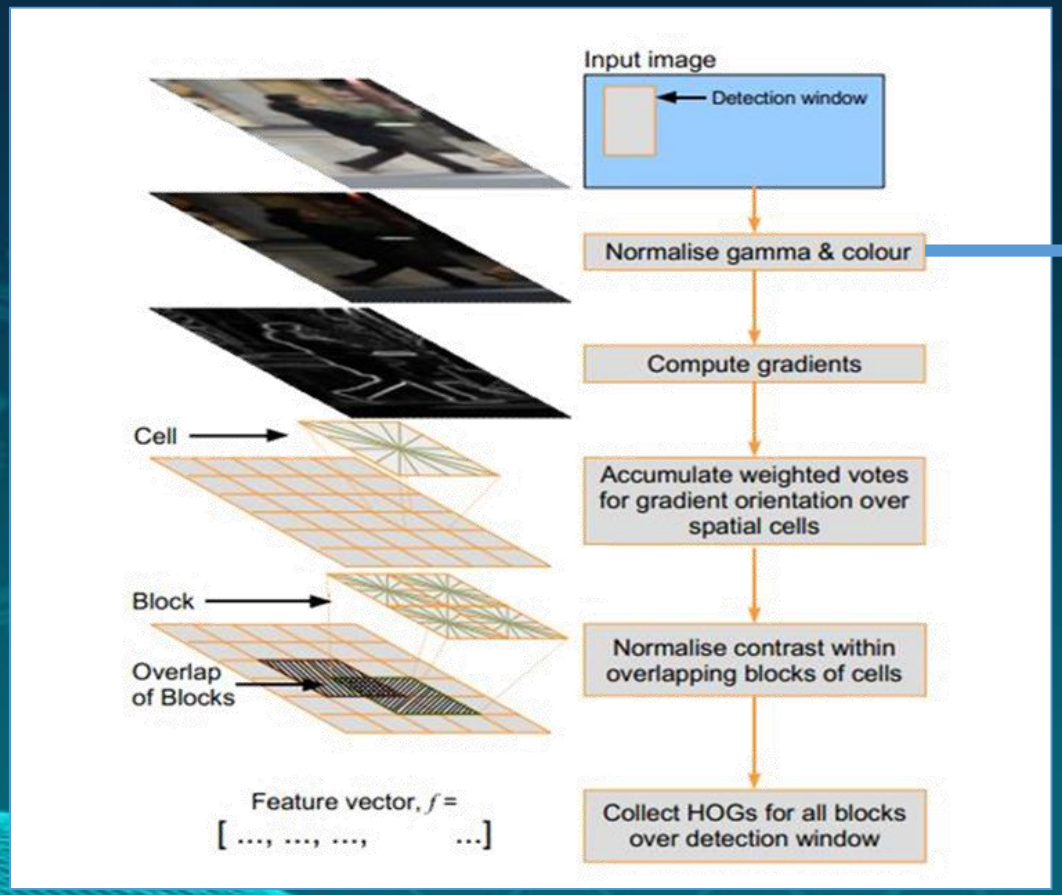
- 主要概念：
  - 在一幅影像中，局部目標的外型和形狀(appearance and shape)能夠被梯度或邊緣的方向密度分佈很好地描述。(本質：梯度的統計資訊，而梯度主要存在於邊緣的地方)。
- 實現方法：
  - 首先將影像轉為灰階影像，為了避免光線、陰影等干擾需進行gamma校正，然後將影像分割成眾多小方格區域，我們稱為Cell。然後採集Cell中各像素點的梯度或邊緣方向長條圖，成為一個特徵值。接著將多個Cell的特徵值合併為一個Block的特徵，再將所有Block特徵合併最終成為該幅影像的特徵值。

HOG(Histogram of gradients)-簡介

- 提高性能：
  - 把這些局部長條圖在影像中更大的範圍內（我們稱為Block）進行對比度正規化（contrast-normalized），所採用的方法是：先計算各長條圖在Block中的密度，然後根據這個密度對Block中的各個Cell做正規化。通過正規化後，能對光照變化和陰影獲得更好的效果。
- 優點：
  - 與其他的特徵描述方法相比，HOG有很多優點。首先，由於HOG是在影像的局部方格單元上操作，所以它對影像幾何和光學的形變都能保持很好的不變性，這兩種形變只會出現在更大的空間領域上。其次，在粗的空間領域抽樣、精細的方向抽樣以及較強的局部光學正規化等條件下，只要行人大體上能夠保持直立的姿勢，可以容許行人有一些細微的肢體動作，這些細微的動作可以被忽略而不影響檢測效果。因此HOG特徵是特別適合於做影像中的人體檢測的。

HOG(Histogram of gradients)-流程簡介

- HOG特徵計算流程圖



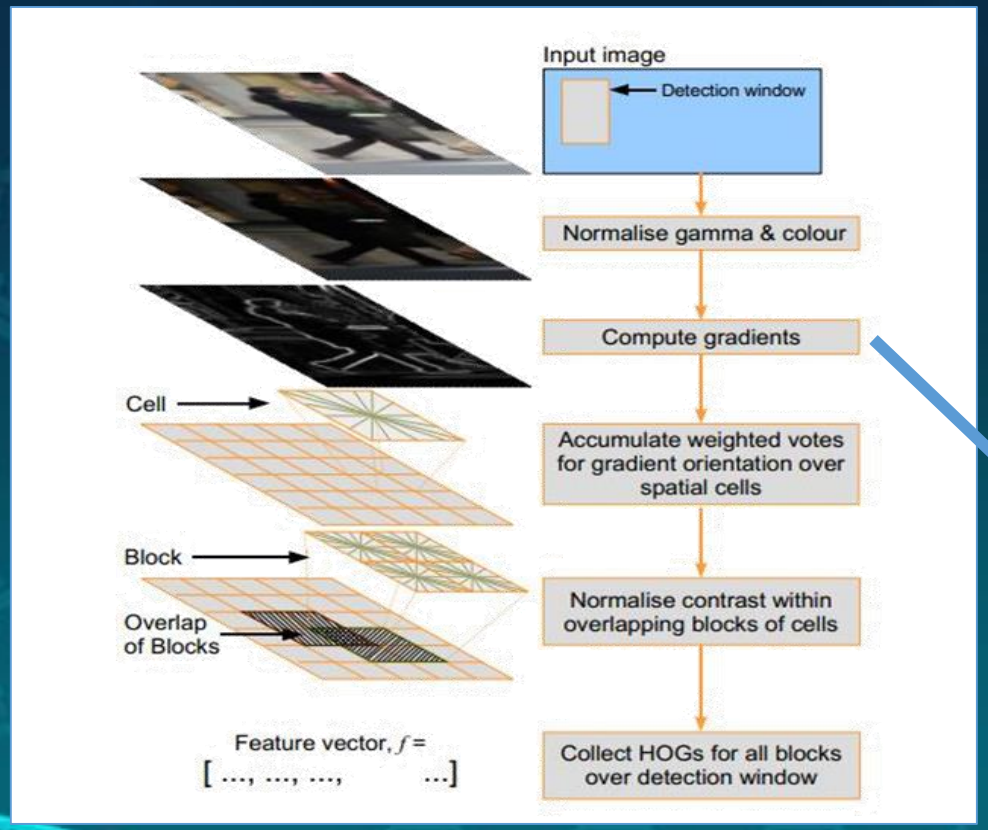
- 為了減少光照因素的影響，首先需要將灰階化後的影像進行Gamma校正，以減少因為影像中光線太強或是太弱所造成的影響。
- 下式為Gamma轉換公式，I為輸入影像，Y為輸出影像，x及y代表像素位置，r為Gamma值，r越小整體亮度值越大，低灰度處的對比度得到增加，能更加有效分辨低灰度值的影像細節。

$$Y(x,y) = I(x,y)^{r}$$

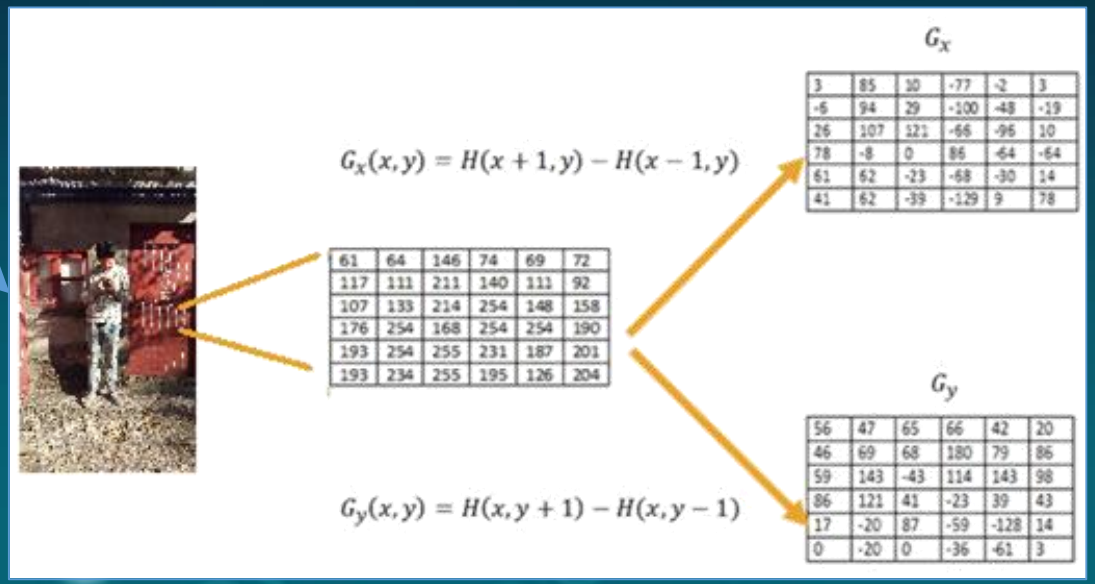
Ref : N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005

HOG(Histogram of gradients)-流程簡介

- HOG特徵計算流程圖



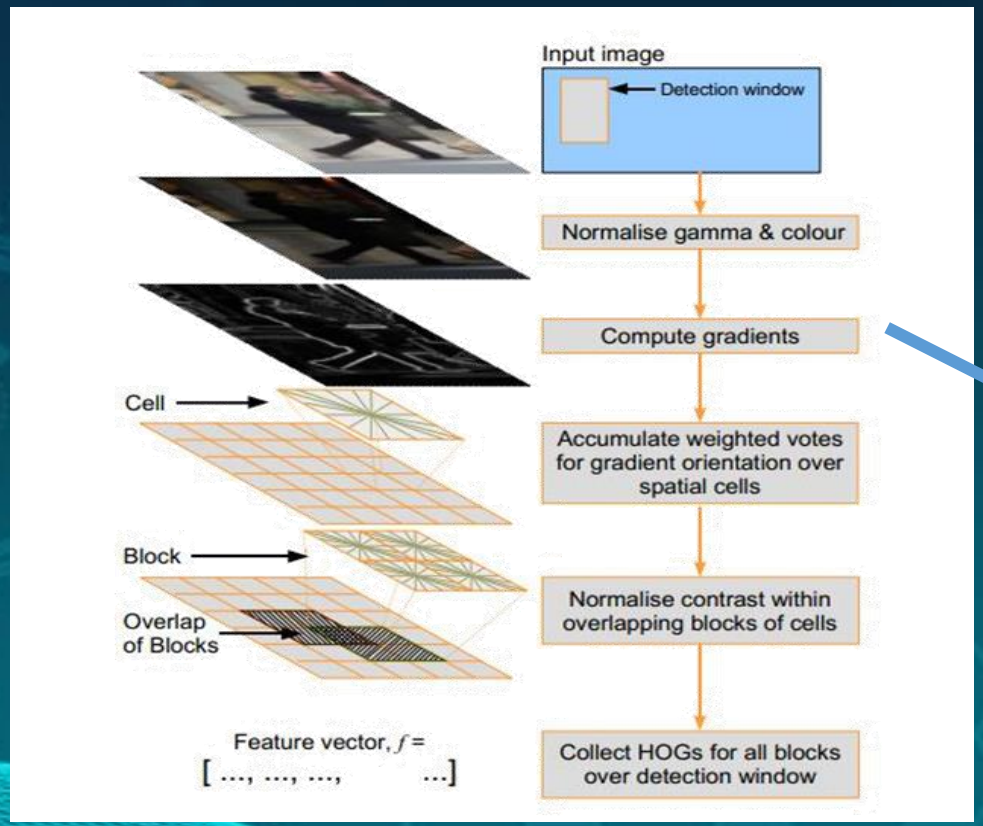
- 計算各個像素點的梯度強度，依據Gradient Filter [-1, 0, 1]計算每一個像素的灰階差值。
- 下式的G_x及G_y，計算出X方向與Y方向的方向強度，如下圖所示，其中H(x,y)代表灰階像素值。



Ref : N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005

HOG(Histogram of gradients)-流程簡介

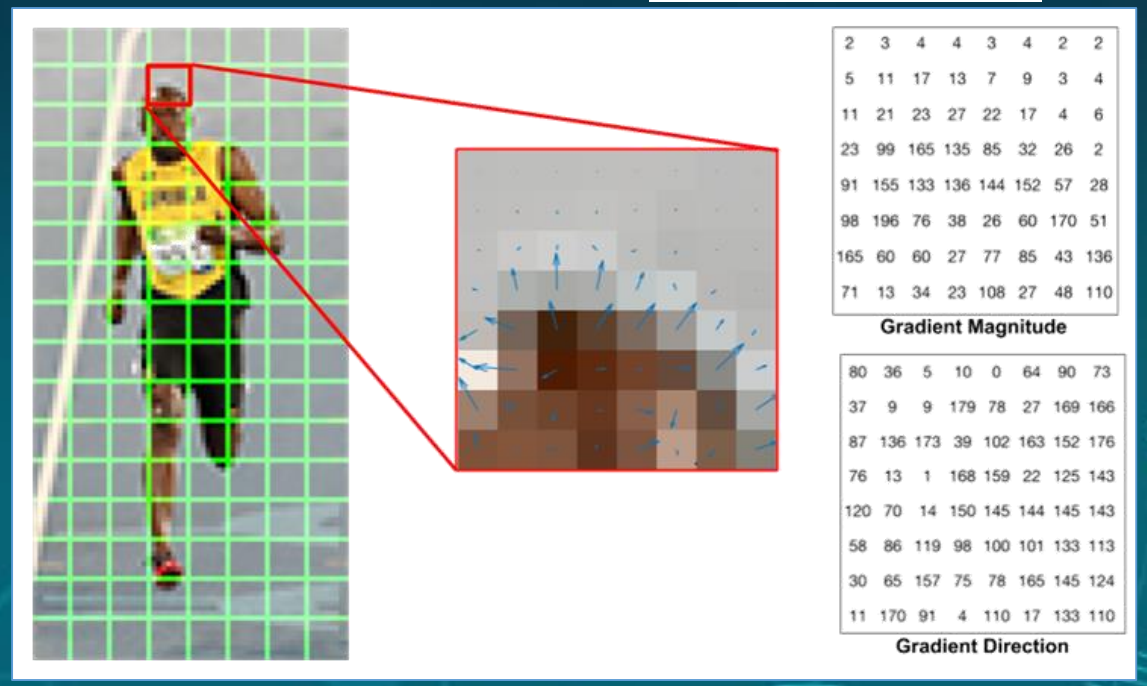
- HOG特徵計算流程圖



- 利用G_x及G_y計算梯度方向θ及梯度向量G。
- 下圖為梯度方向及梯度向量示意圖，其中紅色框代表一個8x8的Cell，Cell中藍色箭頭方向表示該像素的梯度方向，箭頭的長度則表示該像素的梯度向量大小。

$$G(x,y)=\sqrt{G_x(x,y)^2+G_y(x,y)^2}$$

$$\theta(x,y)=\tan^{-1}\frac{G_y(x,y)}{G_x(x,y)}$$

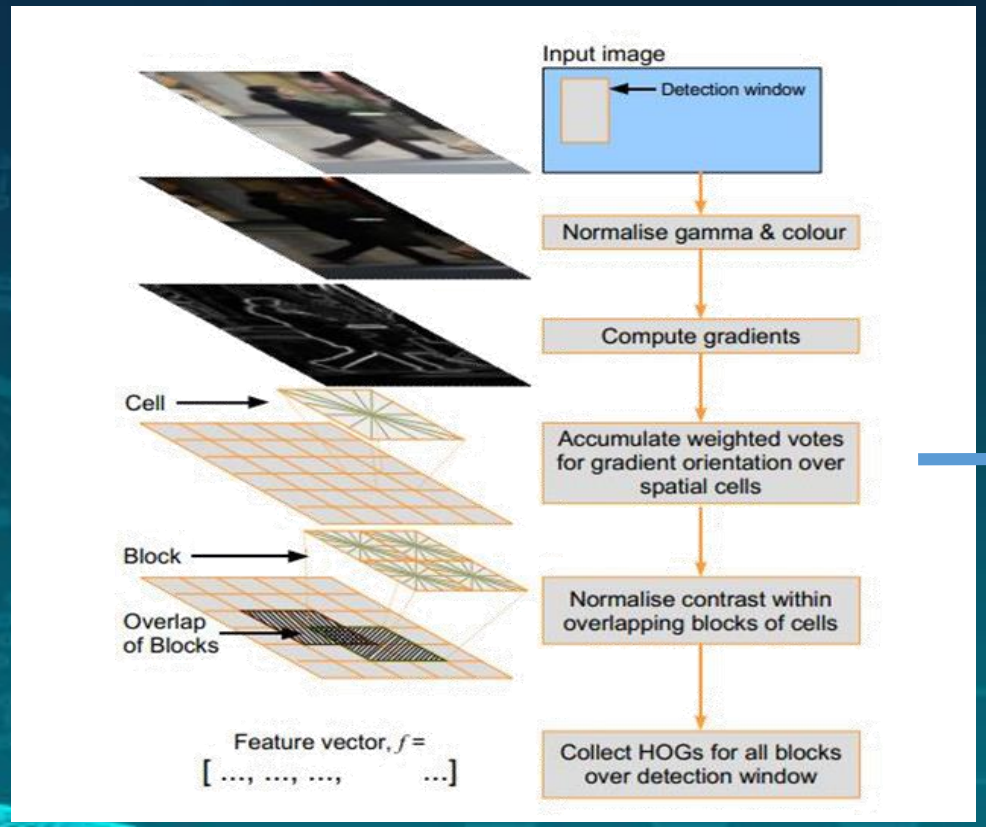


Ref : N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005

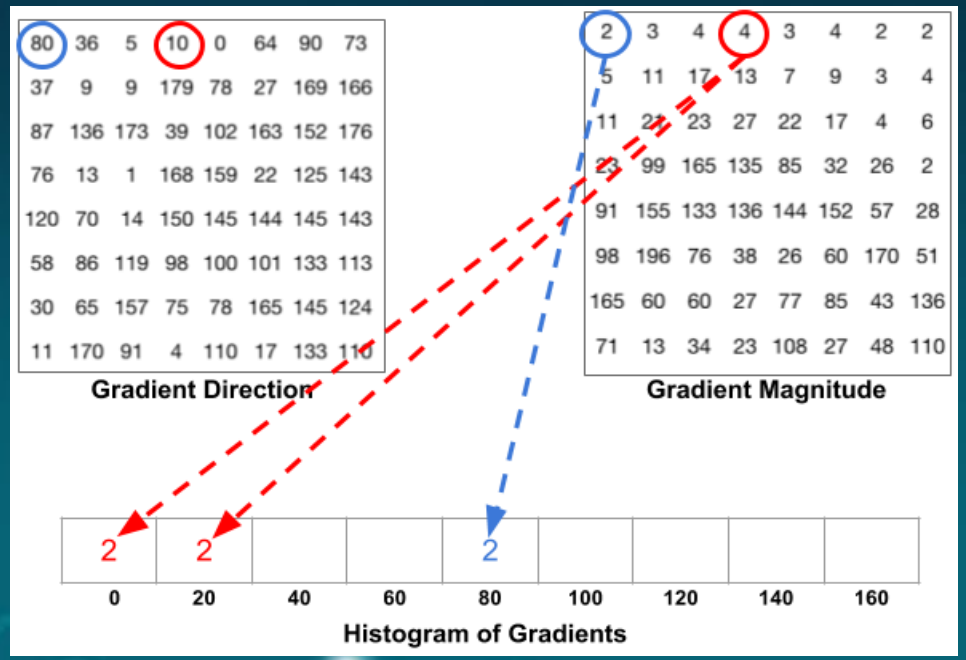
Ref : https://www.learnopencv.com/histogram-of-oriented-gradients/

HOG(Histogram of gradients)-流程簡介

- HOG特徵計算流程圖



- 接著為每個Cell建立梯度直方圖，梯度直方圖由九個方向所組成，每個方向代表20度的方向的向量，如下圖的8x8Cell中的梯度方向及向量表所示，將每個像素中的梯度向量依照梯度方向的權重分布，填入梯度直方圖中。

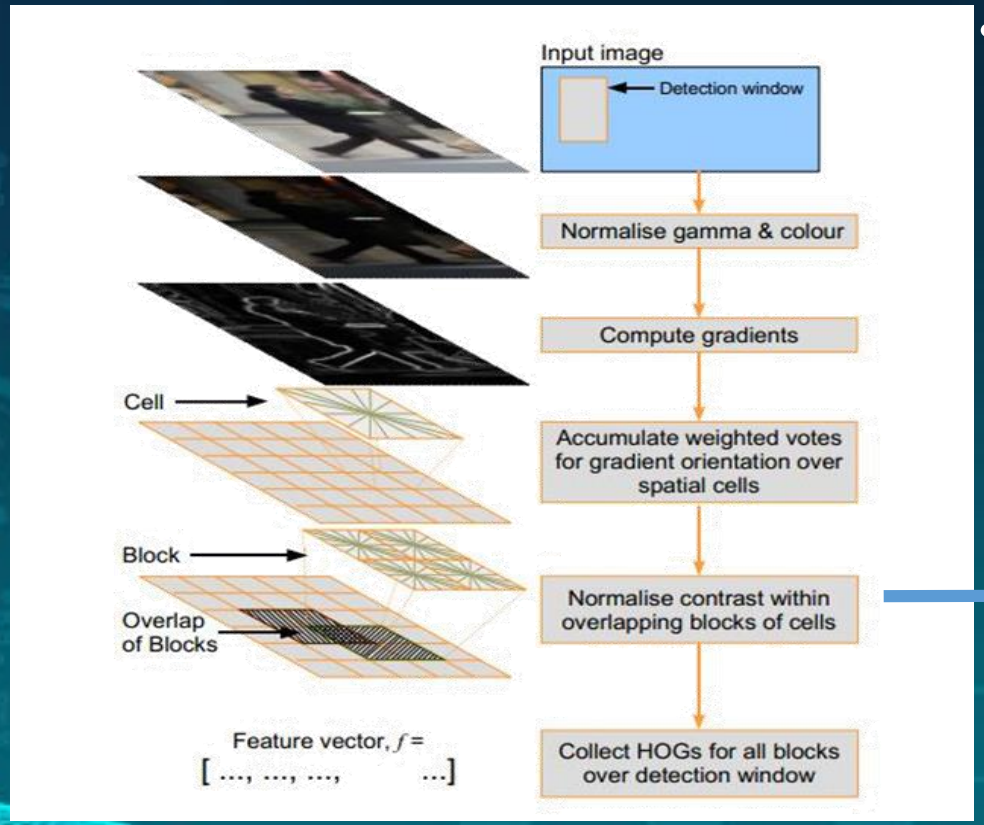


Ref : N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005

Ref : https://actuarialsociety.org.za/convention/convention2017/wp-content/uploads/2017/10/Unlocking_Non_Traditional_Data_final.pdf

HOG(Histogram of gradients)-流程簡介

- HOG特徵計算流程圖



(此範例是以2*2個cell作為一block來說明，其中block大小可以自行設定。在skimage中提供的HOG函式預設為3*3)

- 由於光照情況和背景的變化多樣，梯度值的變化範圍會比較大，每個Cell建立出梯度直方圖後，以四個Cell組成的Block做為正規化範圍。

Block

正規化方式

L2-norm:	$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}$
L2-hys:	進行L2-norm後將向量最大值設為0.2並進行第二次L2-norm
L1-norm:	$f = \sqrt{\frac{v}{(\|v\|_1 + e)}}$
L1-sqrt:	$f = \sqrt{\frac{v}{(\|v\|_1 + e)}}$

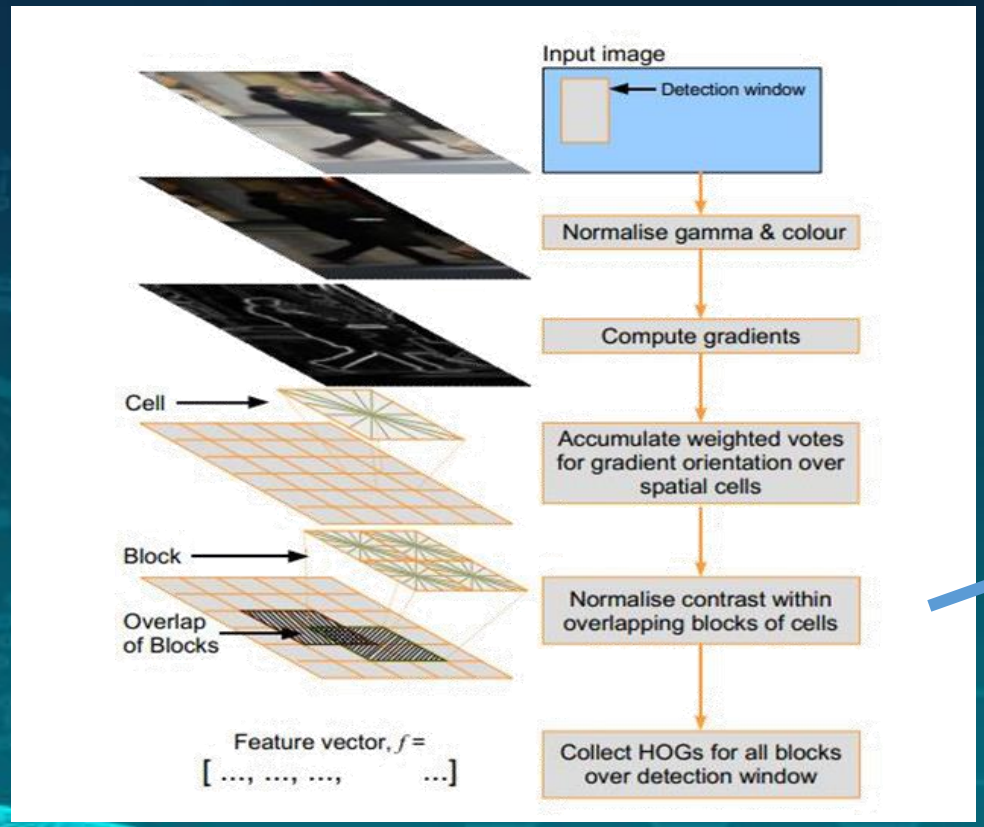
Ref : N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005

Ref : http://pascal.inrialpes.fr/data/human/

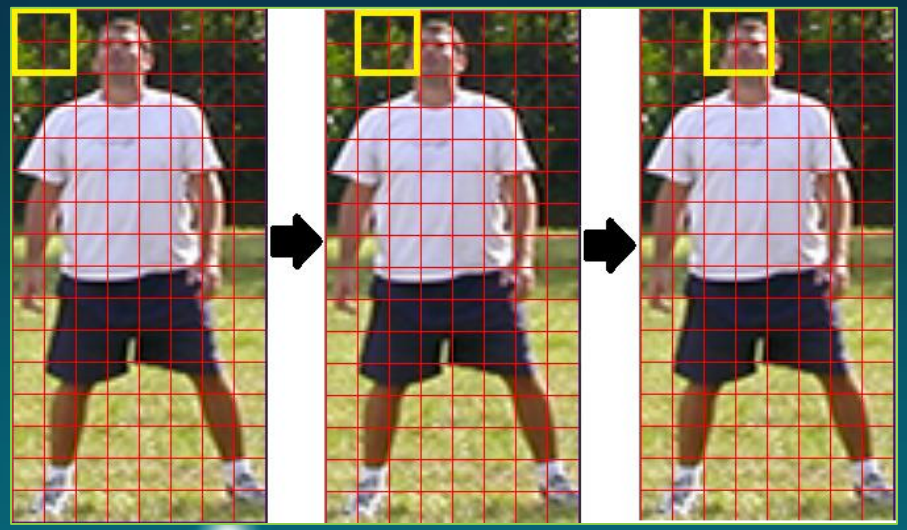
Ref : https://en.wikipedia.org/wiki/Support_vector_machine

HOG(Histogram of gradients)-流程簡介

- HOG特徵計算流程圖



依照黃色框從左上角到右下角進行Block正規化

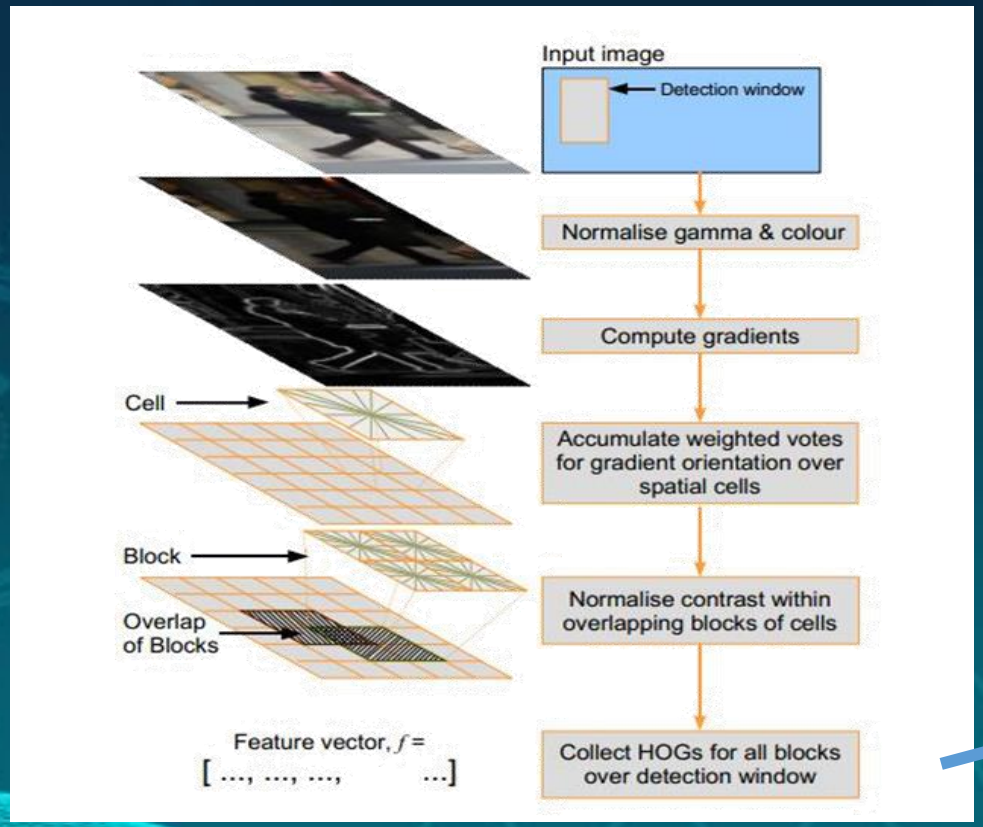


Ref : N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005

Ref : http://pascal.inrialpes.fr/data/human/

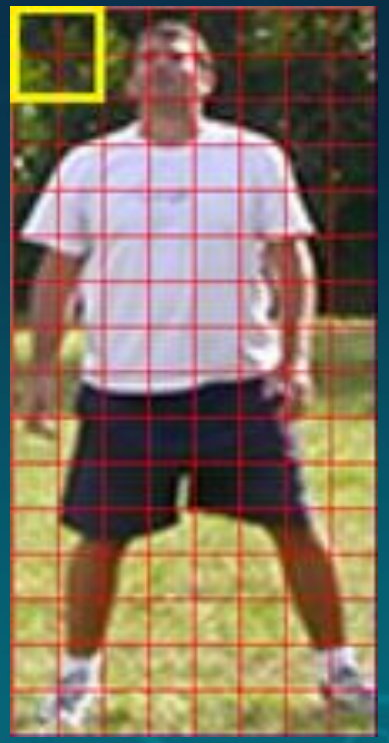
HOG(Histogram of gradients)-流程簡介

- HOG特徵計算流程圖



移動7次

移動15次



64x128影像

Ref : N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005

Ref : http://pascal.inrialpes.fr/data/human/

HOG(Histogram of gradients)-流程簡介

- 取得7x15個Block的資訊，每個Block中有4個Cell，每個Cell有一個9方向的梯度直方圖，最後可取得7x15x4x9=3780的特徵向量。

- 利用取得的HOG特徵向量放入SVM分類器進行訓練及分類。

Ref : http://pascal.inrialpes.fr/data/human/

HOG-Python程式碼實作

- 事前作業：
  - HOG為scikit-image函式庫中所提供的函式。
```
>pip install scikit-image
```
  - 資料可視化函式庫matplotlib。
```
pip install matplotlib
```

Ref : https://scikit-image.org/docs/dev/auto_examples/features_detection/plot_hog.html

HOG-Python程式碼實作

```python
import matplotlib.pyplot as plt
from skimage.feature import hog
from skimage import data, exposure

image = data.astronaut()

fd, hog_image = hog(image, orientations=8, pixels_per_cell=(16, 16),
                    cells_per_block=(1, 1), visualize=True, multichannel=True)

```

- 範例程式：
  - [1~3]引入函式庫
  - [5]讀取scikit-image函式庫中的資料(太空人影像)
  - [7]對影像進行HOG特徵提取並指定參數:
    - out, hog_image = hog(image, orientations, pixels_per_cell, cells_per_block, block_norm, visualize, transform_sqrt, feature_vector, multichannel)
      - out:輸入影像的HOG 描述值，若feature_vector為True，將return 一維陣列。
      - hog_image:可視化HOG影像，若visualize為True 才return。
      - image:指定輸入影像(ndarray)。
      - orientations:Number of orientation bins，預設為9。
      - pixels_per_cell:指定Cell的大小，預設為(8, 8)。
      - cells_per_block:指定Block的大小，預設為(3, 3)。
      - block_norm:要進行正規化的方式，有‘L1’, ‘L1-sqrt’, ‘L2’, ‘L2-Hys’等方式，預設為‘L2-Hys’。
      - visualize: 決定是否return HOG特徵影像，預設為False。
      - transform_sqrt:在處理之前，應用冪定律壓縮對影像進行正規化。如果影像包含負值，請勿使用此選項。預設為False。
      - feature_vector:若為True，將return 特徵的一維陣列。預設為True。
      - multichannel:若為True，則將影像視為一個3通道(彩色)，否則視為單通道。預設為None。

Ref : https://scikit-image.org/docs/dev/auto_examples/features_detection/plot_hog.html

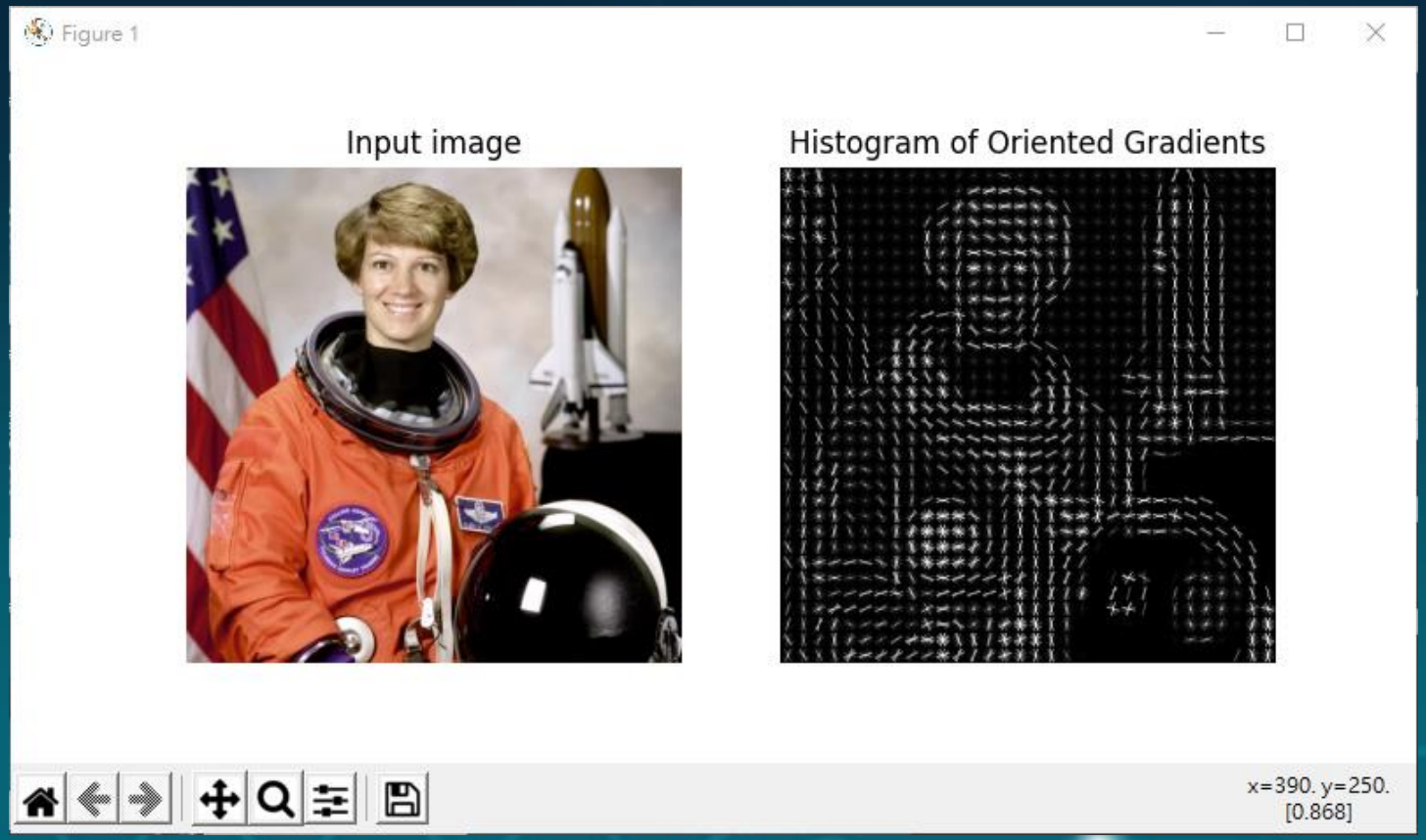
HOG-Python程式碼實作

```
10  fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(8, 4), sharex=True, sharey=True)
11
12  ax1.axis('off')
13  ax1.imshow(image, cmap=plt.cm.gray)
14  ax1.set_title('Input image')
15
16  # Rescale histogram for better display
17  hog_image_rescaled = exposure.rescale_intensity(hog_image, in_range=(0, 10))
18
19  ax2.axis('off')
20  ax2.imshow(hog_image_rescaled, cmap=plt.cm.gray)
21  ax2.set_title('Histogram of Oriented Gradients')
22  plt.show()
```

- 範例程式：
  - [10] 使用subplots函式建立可視化視窗，其中有1 row, 2 column，sharex和sharey分別為Ture代表能在可視化視窗中共用左右兩張影像的x, y值(同時zoom in, zoom out)
  - [12, 19]不顯示影像座標軸
  - [13, 20]秀圖
  - [14, 21]設定影像標題
  - [17]重新縮放直方圖以取得更好的展示圖
  - [22]秀出可視化視窗

HOG-Python程式碼實作

- 執行結果：



HOG-結合SVM分類器

- HOG特徵結合SVM分類器目前已經被廣泛應用於影像識別中，尤其在行人偵測中獲得了極大的成功。
- 以下會說明如何將影像中提取出的HOG特徵送入SVM分類器中以訓練模型，並使用該模型辨識輸入影像。

HOG-結合SVM分類器

- 流程：
  - 以辨識人的影像為例，我們可準備2類資料集：
    1. 人的影像(target:0)
    2. 非人的影像(target:1)
  - 分別將2類資料集所有的影像進行HOG特徵提取
  - 輸出的2個特徵陣列進行串接
  - 將特徵陣列送入SVM進行訓練
  - 使用訓練出的模型對輸入影像進行預測
    - (輸入人的影像若為0代表成功)

HOG-結合SVM分類器

- 取特徵：

```
7  fd, hog_image = hog(image, orientations=8, pixels_per_cell=(16, 16),
8                      cells_per_block=(1, 1), visualize=True, multichannel=True)
```

  - 此處對2類別來源影像進行HOG特徵提取，並同時給予target。
  - fd為特徵陣列，hog_image為可視化特徵影像。前者為我們想要的結果(即data)。
- 訓練：
  - train_test_split(data, target, test_size=0.2, random_state=0): 將data和其對應的target分割成8:2的比例
  - clf.fit(data, target): 給予data和其對應的target送入SVM進行訓練
- 預測：
  - clf.predict(img): img預測對象可為多個影像的陣列或單一影像
  - clf.score(data, target): 輸出以data進行predict後的結果與target進行比對計算準確率

```
72 X_train, X_test, y_train, y_test = \
73     train_test_split(X, y,test_size=0.2,random_state=0)
74
75 clf = svm.SVC(kernel='linear',C=1,gamma='auto')
76 clf.fit(X_train,y_train)
77
78 print("predict")
79 print(clf.predict(X_train))
80 print(clf.predict(X_test))
81 print("Accuracy:")
82 print(clf.score(X_train,y_train))
83 print(clf.score(X_test, y_test))
```

```
predict
[0. 0. 0. ... 0. 0. 1.]
[0. 0. 0. ... 1. 0. 1.]
Accuracy:
0.9966460417509975
0.9934081184225743
```