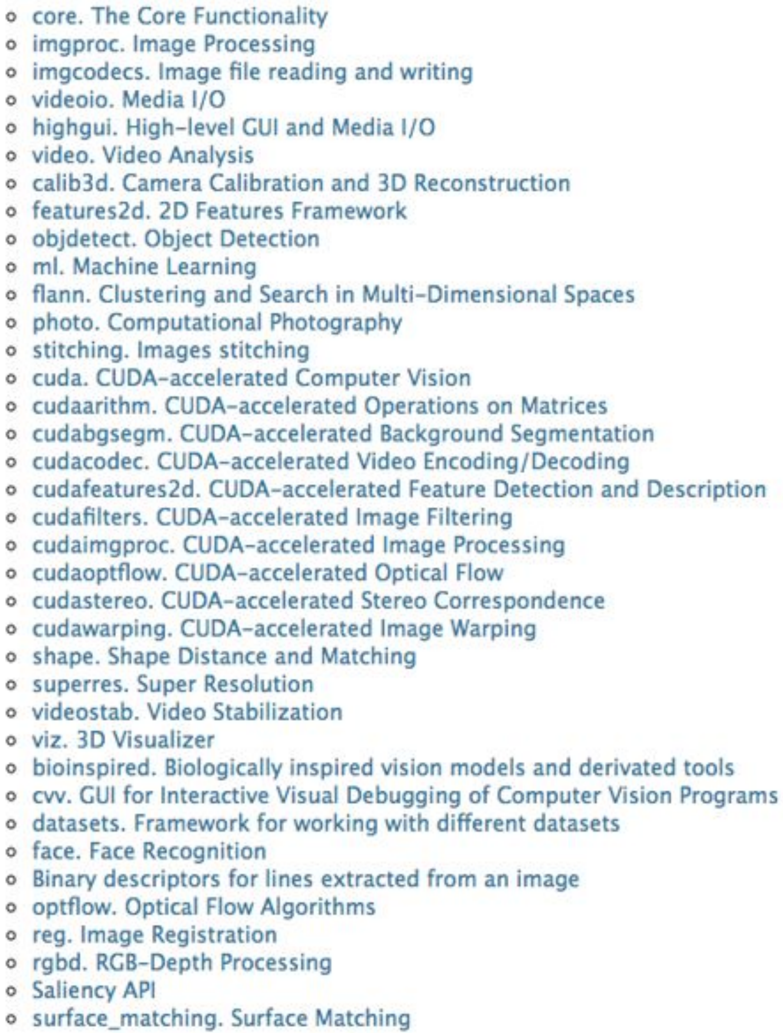
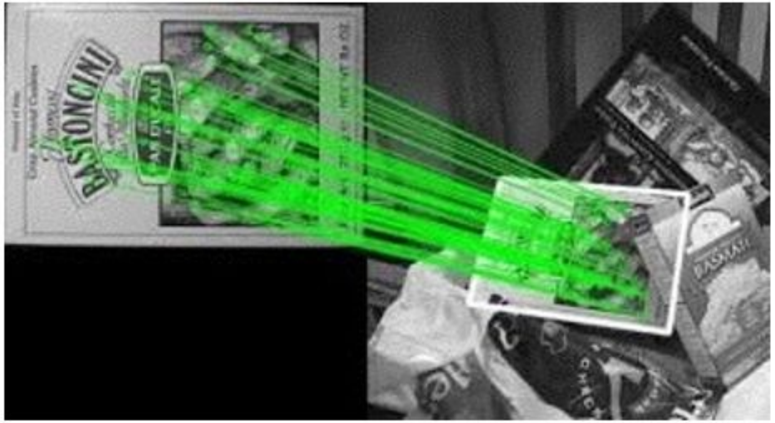
3/4 Lab

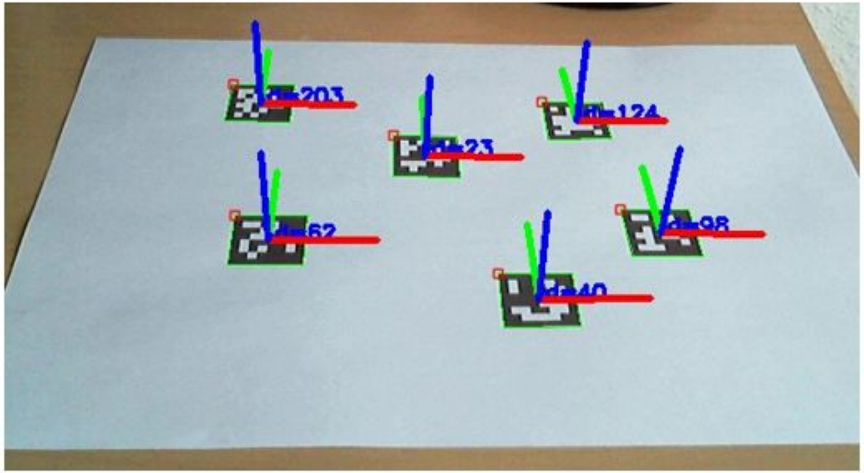
1. OpenCV introduction
2. Python 3 & opencv installation
3. Lab01



- core. The Core Functionality
- imgproc. Image Processing
- imgcodecs. Image file reading and writing
- videoio. Media I/O
- highgui. High-level GUI and Media I/O
- video. Video Analysis
- calib3d. Camera Calibration and 3D Reconstruction
- features2d. 2D Features Framework
- objdetect. Object Detection
- ml. Machine Learning
- flann. Clustering and Search in Multi-Dimensional Spaces
- photo. Computational Photography
- stitching. Images stitching
- cuda. CUDA-accelerated Computer Vision
- cudaarithm. CUDA-accelerated Operations on Matrices
- cudabgsegm. CUDA-accelerated Background Segmentation
- cudacodec. CUDA-accelerated Video Encoding/Decoding
- cudafeatures2d. CUDA-accelerated Feature Detection and Description
- cudafilters. CUDA-accelerated Image Filtering
- cudaimgproc. CUDA-accelerated Image Processing
- cudaoptflow. CUDA-accelerated Optical Flow
- cudastereo. CUDA-accelerated Stereo Correspondence
- cudawarping. CUDA-accelerated Image Warping
- shape. Shape Distance and Matching
- superres. Super Resolution
- videostab. Video Stabilization
- viz. 3D Visualizer
- bioinspired. Biologically inspired vision models and derivated tools
- cvv. GUI for Interactive Visual Debugging of Computer Vision Programs
- datasets. Framework for working with different datasets
- face. Face Recognition
- Binary descriptors for lines extracted from an image
- optflow. Optical Flow Algorithms
- reg. Image Registration
- rgbd. RGB-Depth Processing
- Saliency API
- surface_matching. Surface Matching

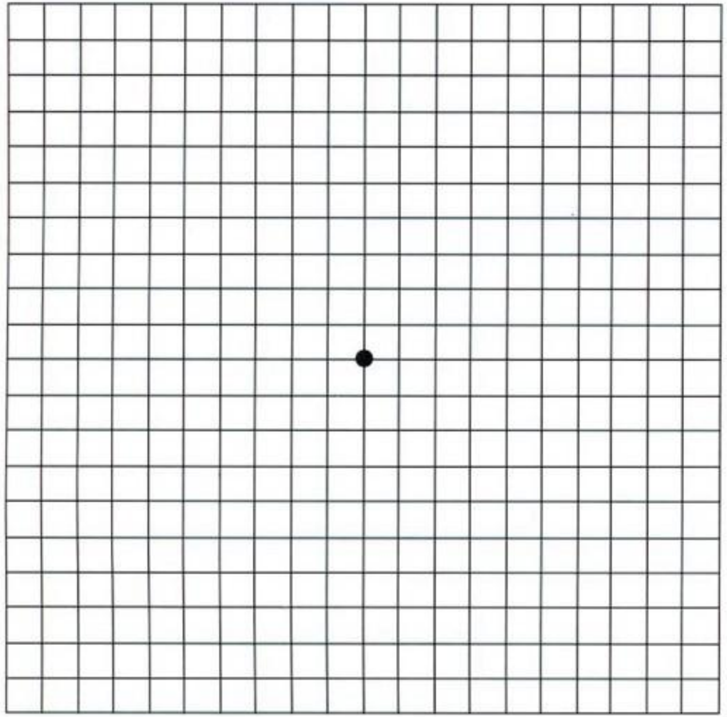
feature detection





pattern recognition

Mat



rows: 長

cols: 寬

type: 像素型態

channels: 通道數

normal:

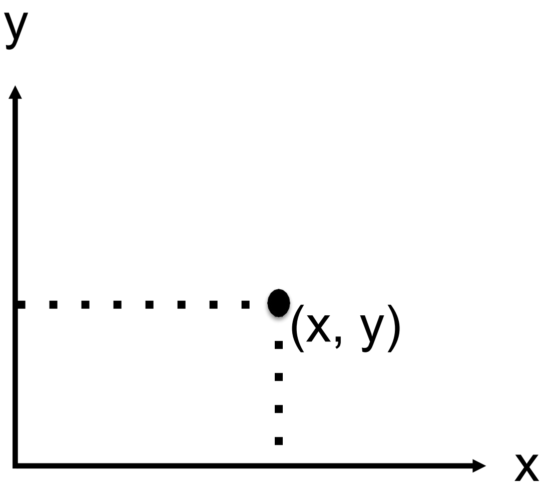
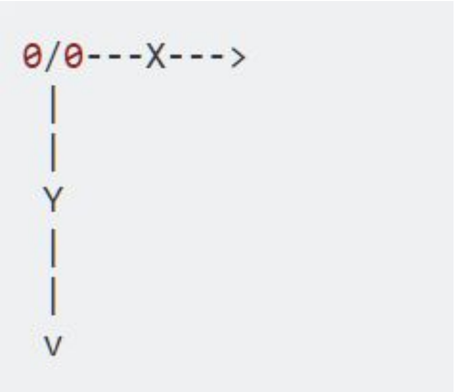
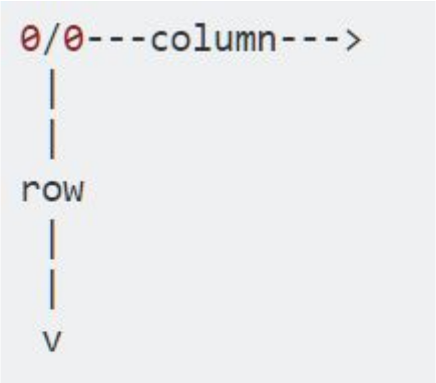


image:

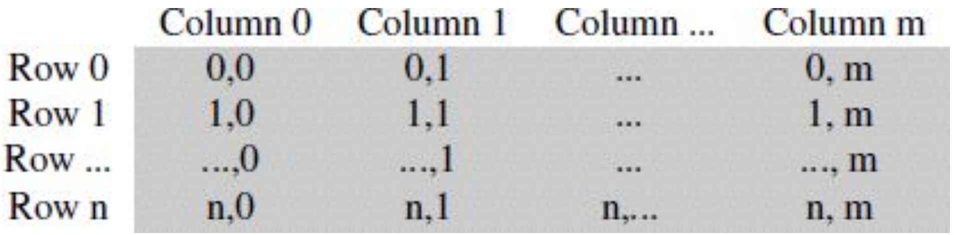


```
0/0---X--->
 |
 |
 Y
 |
 |
 v
```



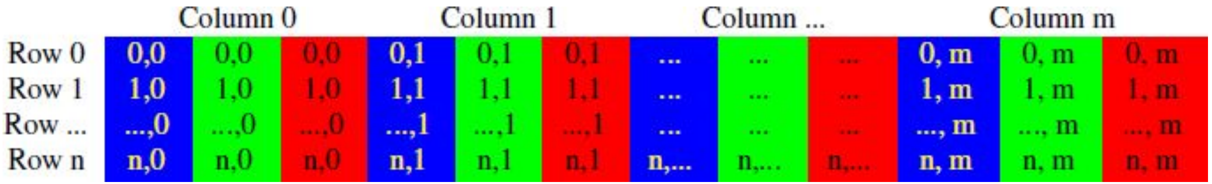
```
0/0---column--->
 |
 |
row
 |
 |
 v
```

Mat value access

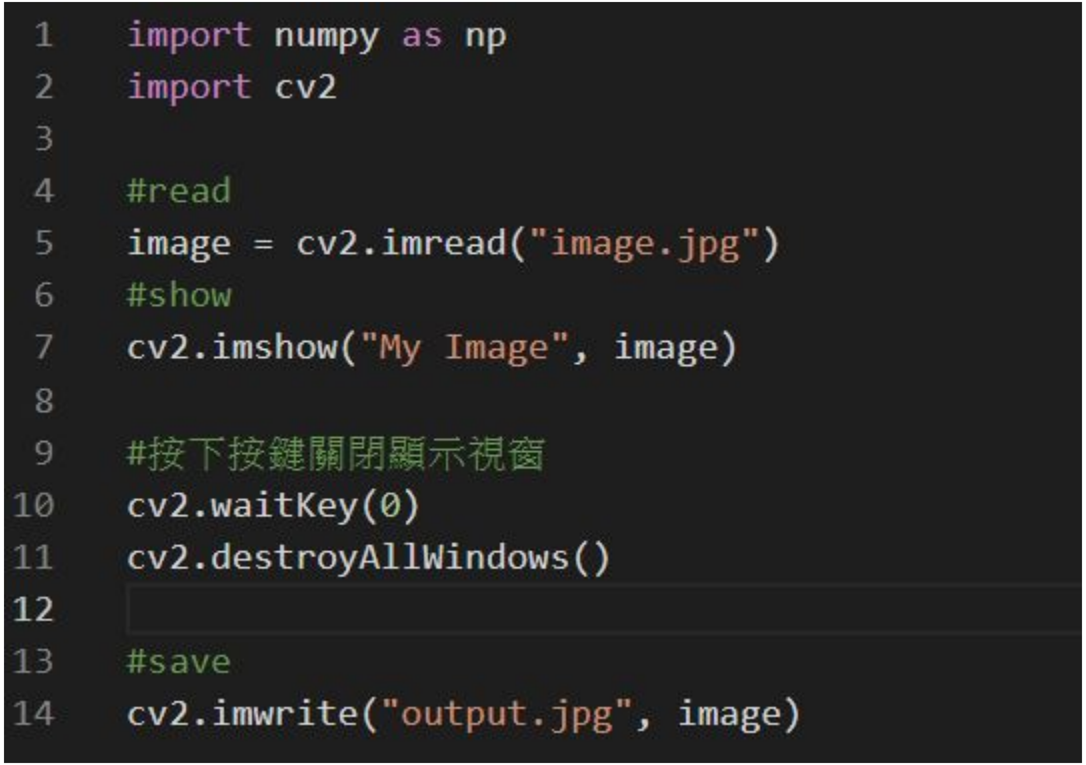


	Column 0	Column 1	Column ...	Column m
Row 0	0,0	0,1	...	0, m
Row 1	1,0	1,1	...	1, m
Row ...	...,0	...,1	...	..., m
Row n	n,0	n,1	n,...	n, m

3-channel : B, G, R

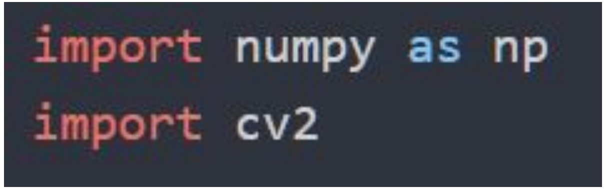


	Column 0			Column 1			Column ...			Column m		
Row 0	0,0	0,0	0,0	0,1	0,1	0,1	...	...	...	0, m	0, m	0, m
Row 1	1,0	1,0	1,0	1,1	1,1	1,1	...	...	...	1, m	1, m	1, m
Row ...	...,0	...,0	...,0	...,1	...,1	...,1	...	...	...	..., m	..., m	..., m
Row n	n,0	n,0	n,0	n,1	n,1	n,1	n,...	n,...	n,...	n, m	n, m	n, m



```python
import numpy as np
import cv2

#read
image = cv2.imread("image.jpg")
#show
cv2.imshow("My Image", image)

#按下按鍵關閉顯示視窗
cv2.waitKey(0)
cv2.destroyAllWindows()

#save
cv2.imwrite("output.jpg", image)
```

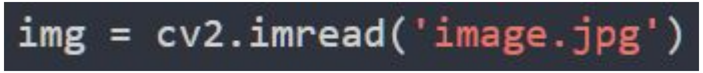
標頭引入



```
import numpy as np
import cv2
```

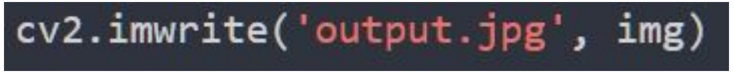
讀寫圖片

讀取:



```
img = cv2.imread('image.jpg')
```

儲存:

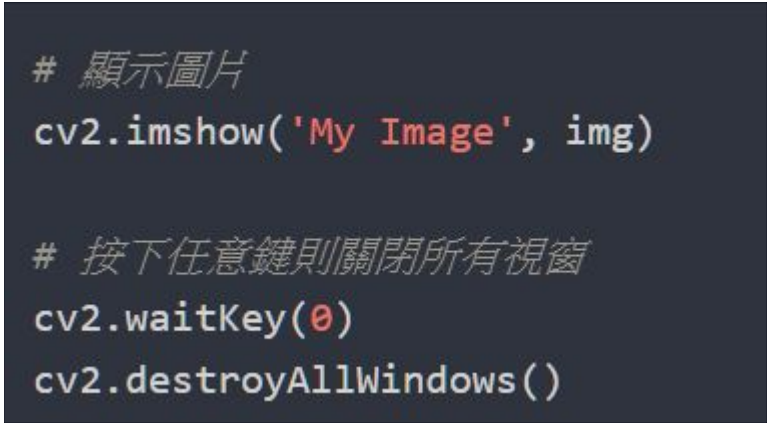


```
cv2.imwrite('output.jpg', img)
```

顯示圖片

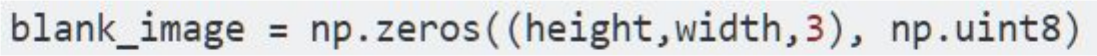
秀出影像:

等待按鍵輸入:

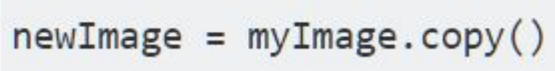


```
# 顯示圖片
cv2.imshow('My Image', img)

# 按下任意鍵則關閉所有視窗
cv2.waitKey(0)
cv2.destroyAllWindows()
```

開一個指定大小的黑圖
複製圖片



```
blank_image = np.zeros((height,width,3), np.uint8)
```



```
newImage = myImage.copy()
```

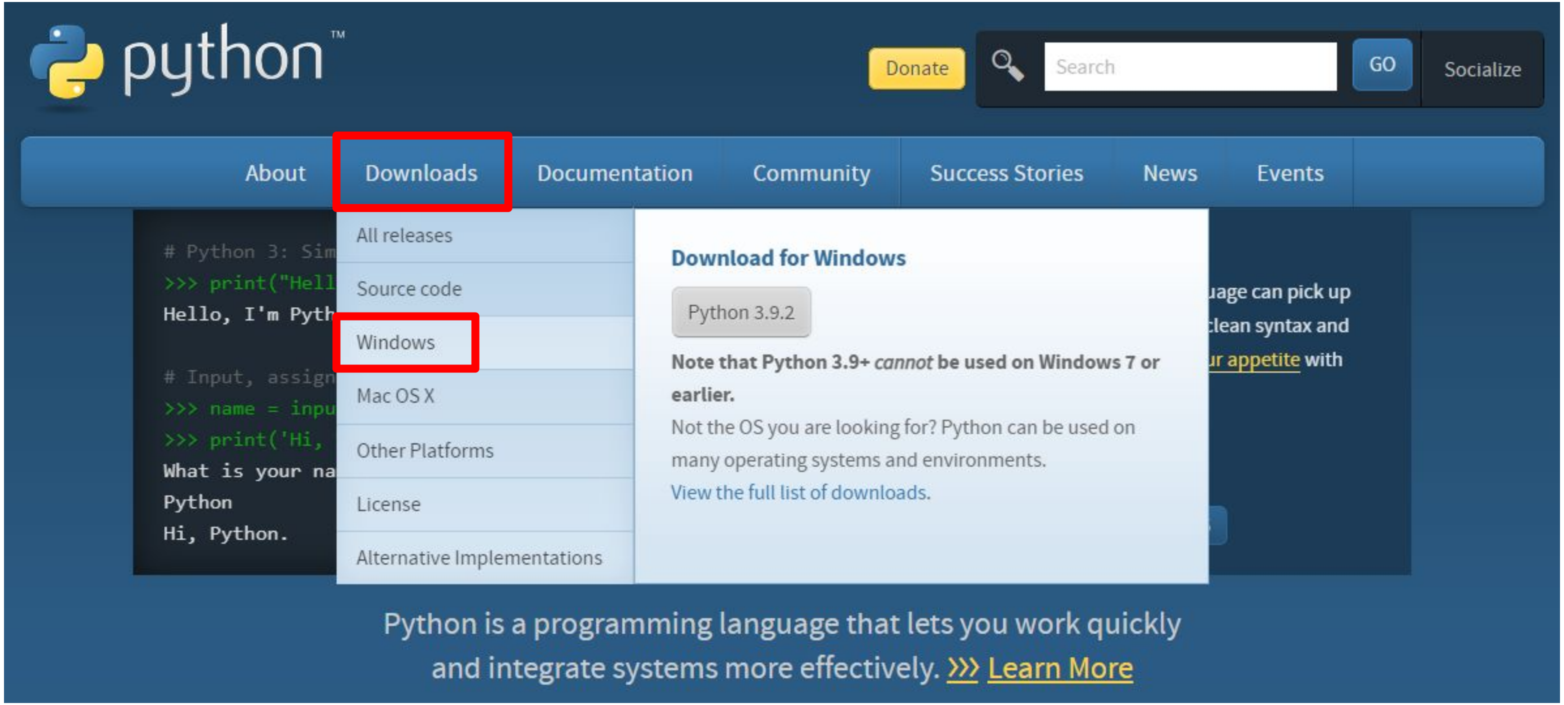
操作像素

image[row, col, channel]

Windows10
python 3 & opencv

1. 下載python

到官網點選Downloads



```
# Python 3: Sim
>>> print("Hell
Hello, I'm Pyth

# Input, assign
>>> name = inpu
>>> print('Hi,
What is your na
Python
Hi, Python.
```

Download for Windows

Note that Python 3.9+ *cannot* be used on Windows 7 or earlier.

Not the OS you are looking for? Python can be used on many operating systems and environments.

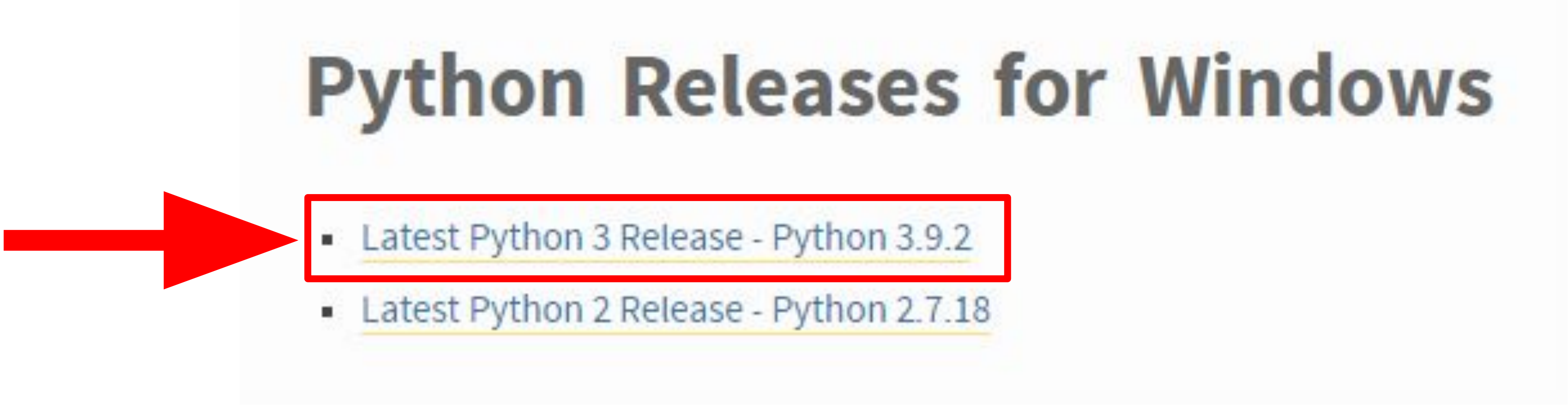
View the full list of downloads.

uage can pick up
clean syntax and
ur appetite with

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> Learn More

1. 下載python

選擇python 3

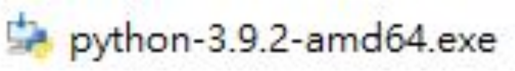


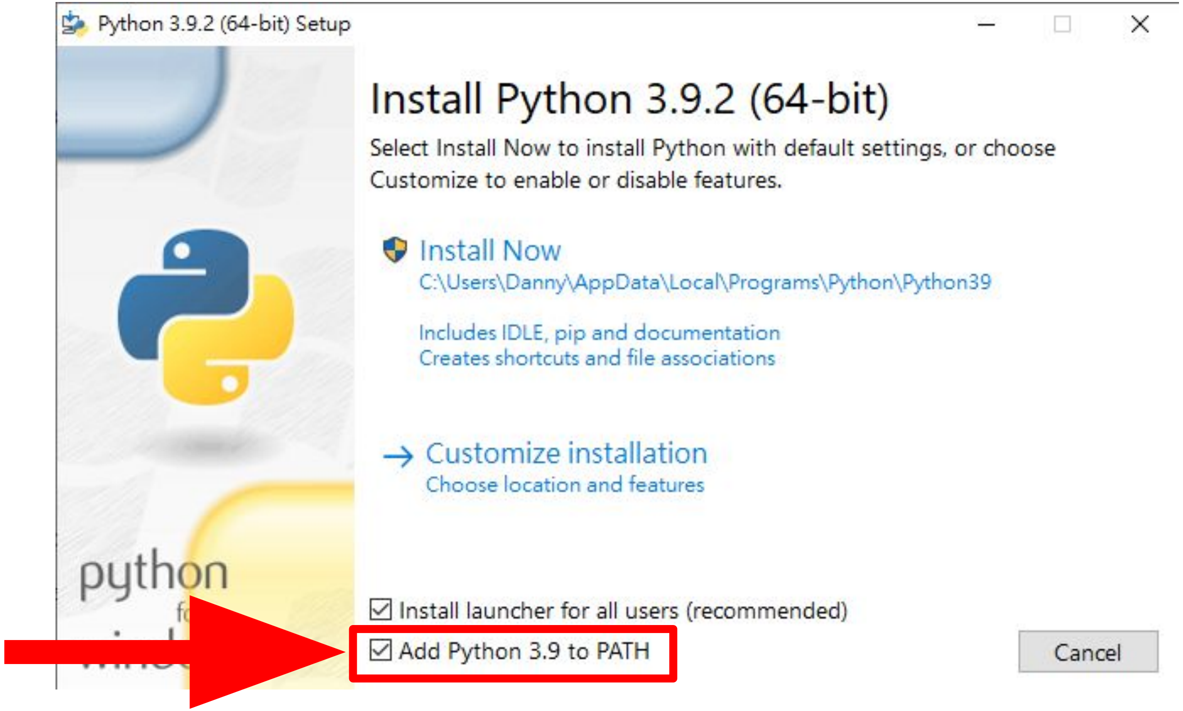
Python Releases for Windows

- Latest Python 3 Release - Python 3.9.2
- Latest Python 2 Release - Python 2.7.18

2. 安裝軟體

- 點選並安裝
- 加入環境變數


python-3.9.2-amd64.exe


Python 3.9.2 (64-bit) Setup

Install Python 3.9.2 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\Danny\AppData\Local\Programs\Python\Python39

Includes IDLE, pip and documentation
Creates shortcuts and file associations

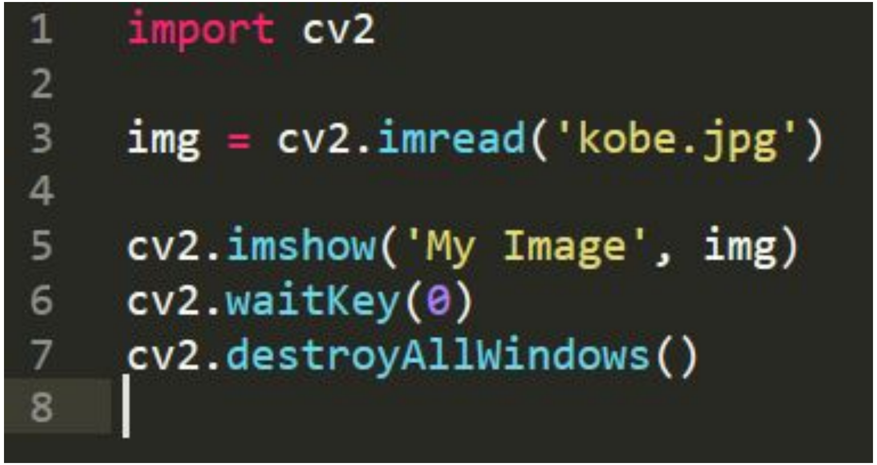
→ Customize installation
Choose location and features

☑ Install launcher for all users (recommended)

☑ Add Python 3.9 to PATH

3. 安裝opencv

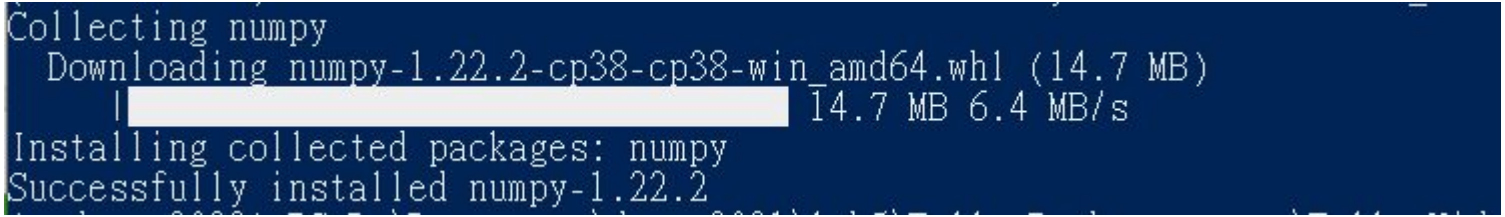
- pip install opencv-python
- Test :



```python
import cv2

img = cv2.imread('kobe.jpg')

cv2.imshow('My Image', img)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

4. 安裝numpy

- pip install numpy



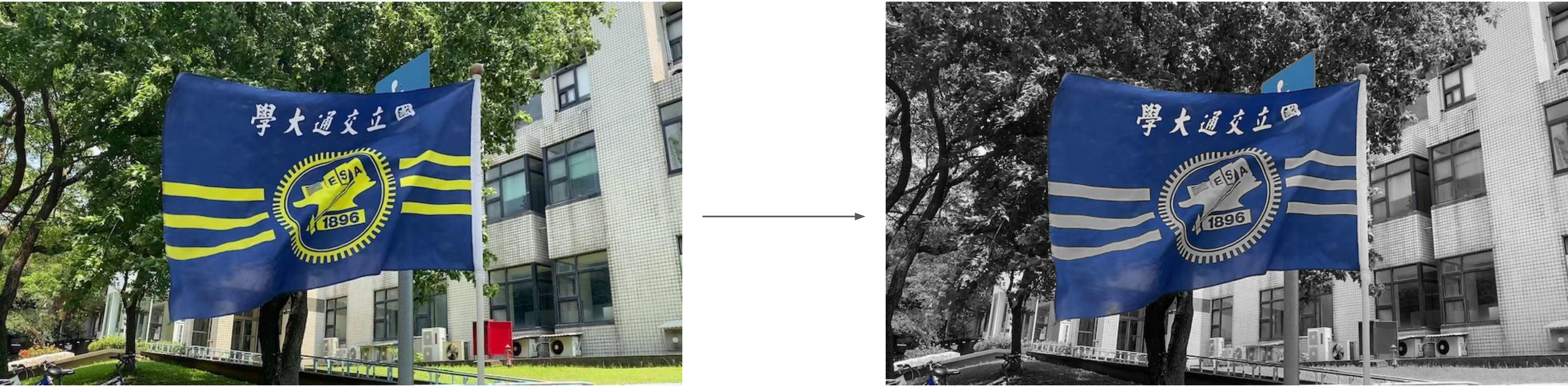
```
Collecting numpy
  Downloading numpy-1.22.2-cp38-cp38-win_amd64.whl (14.7 MB)
     |████████████████████████████████| 14.7 MB 6.4 MB/s
Installing collected packages: numpy
Successfully installed numpy-1.22.2
```

Lab 01

1. 圖片灰階與顏色濾鏡
2. Interpolation (bilinear, nearest neighbor)

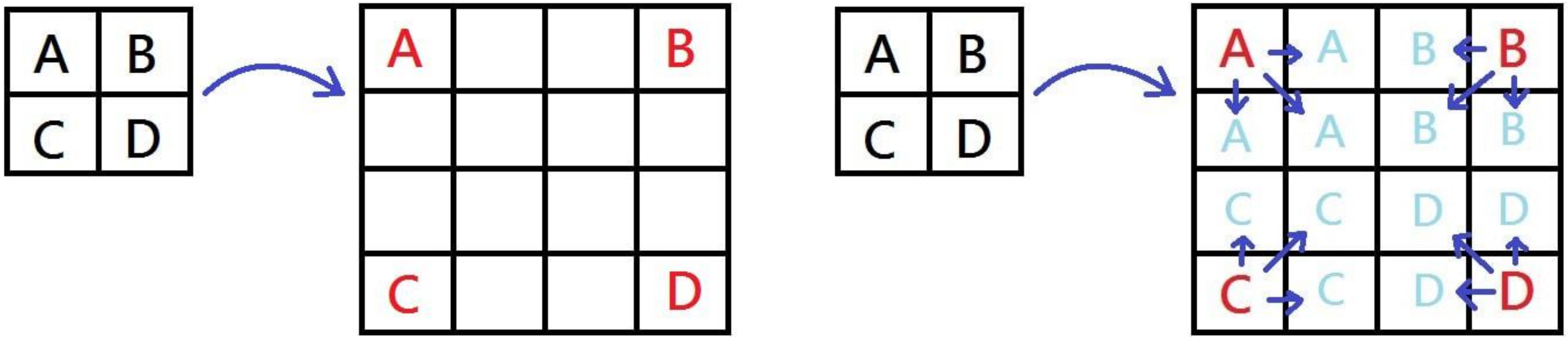
1. 灰階與藍色濾鏡(30%)

- 將原始圖片中的藍點予以保留，並把其餘的點改為灰階。
- Hint: B > 100 and B * 0.6 > G and B * 0.6 > R



2. Interpolation - 最近相鄰內插法 (30%)

- 根據輸出影像的像素位置 ,找到輸入影像中最鄰近的點 ,即當作輸出影像的像素強度。
- 以下圖為例



A	B
C	D

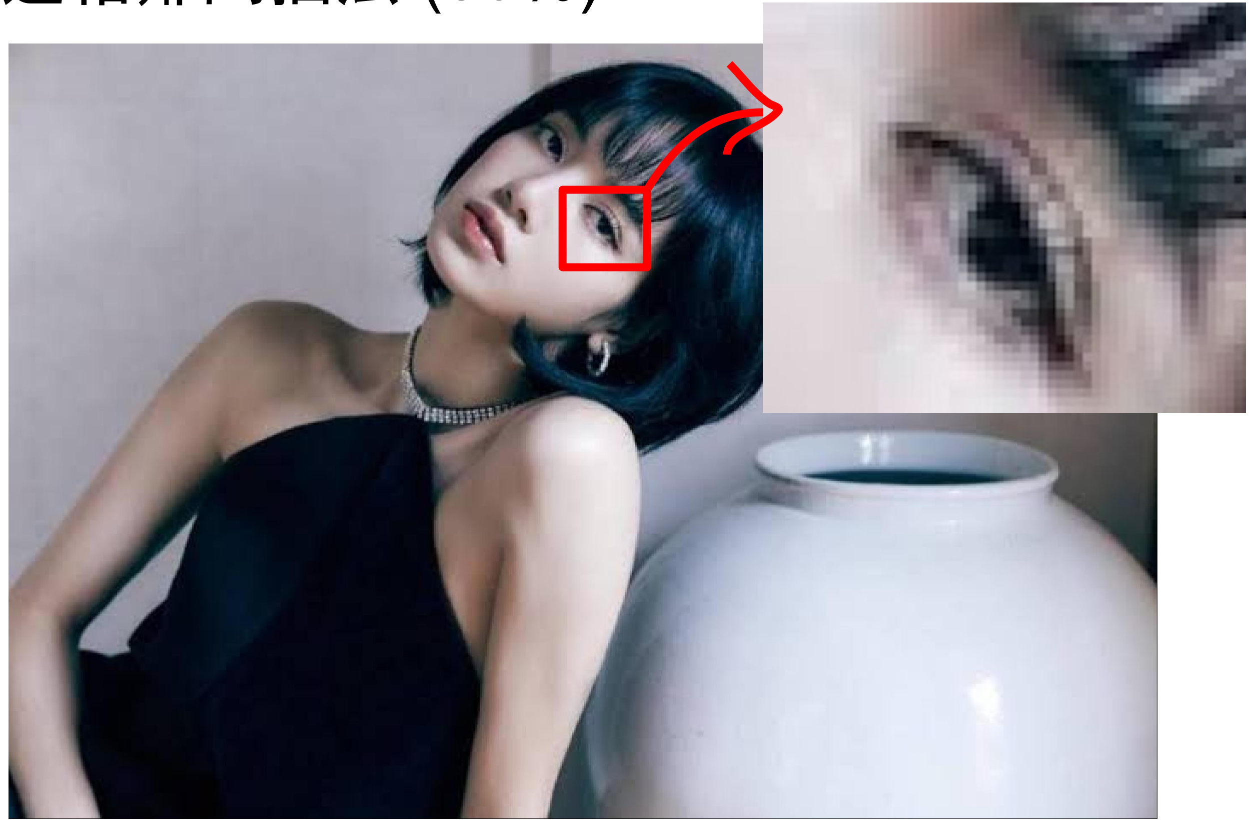
A			B
C			D

A	B
C	D

A	A	B	B
A	A	B	B
C	C	D	D
C	C	D	D

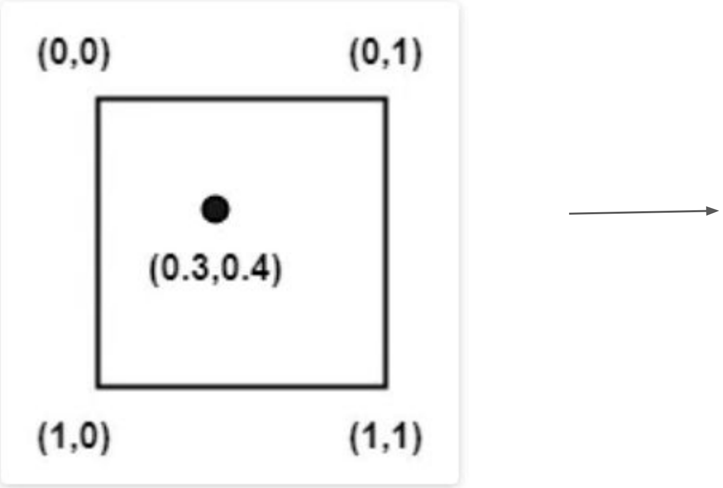
2. Interpolation - 最近相鄰內插法 (30%)

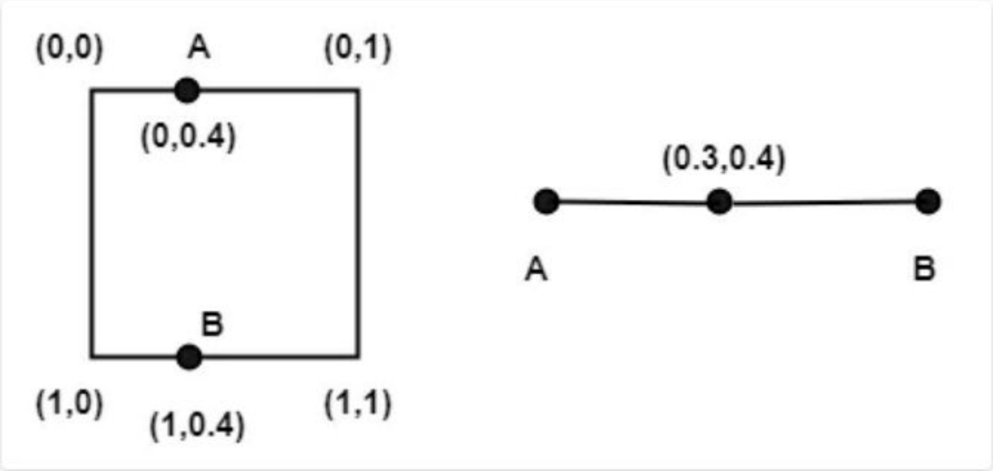
- 將照片放大3倍



3. Interpolation - 雙線性內插法 (40%)

- 根據輸出影像的像素位置，找到輸入影像中最鄰近的四個點 ,再利用雙線性 內插法求出輸出影像的像素強度。





3. Interpolation - 雙線性內插法 (40%)

- 以參數方式輸入影像以及倍率
- 學會使用 OpenCV API (10%)

  自行實作雙線性 內插法 (40%)
- 下圖為輸入影像
  右圖為
  倍率=3之結果