

The background of the slide is a light gray gradient. It is decorated with numerous realistic water droplets of various sizes. Some droplets are large and prominent, while others are small and scattered. They are primarily located in the top-left and bottom-right corners, with a few smaller ones in the center and along the edges. The droplets have a glossy, reflective surface with highlights and shadows, giving them a three-dimensional appearance.

從零開始的影像處理

CHAP0: PYTHON進階程式設計

BASIC INFORMATION

- INSTRUCTOR

- 蔡明翰
- MINGHTSAI@FCU.EDU.TW
- 資電館229
- 分機#3737

- TA

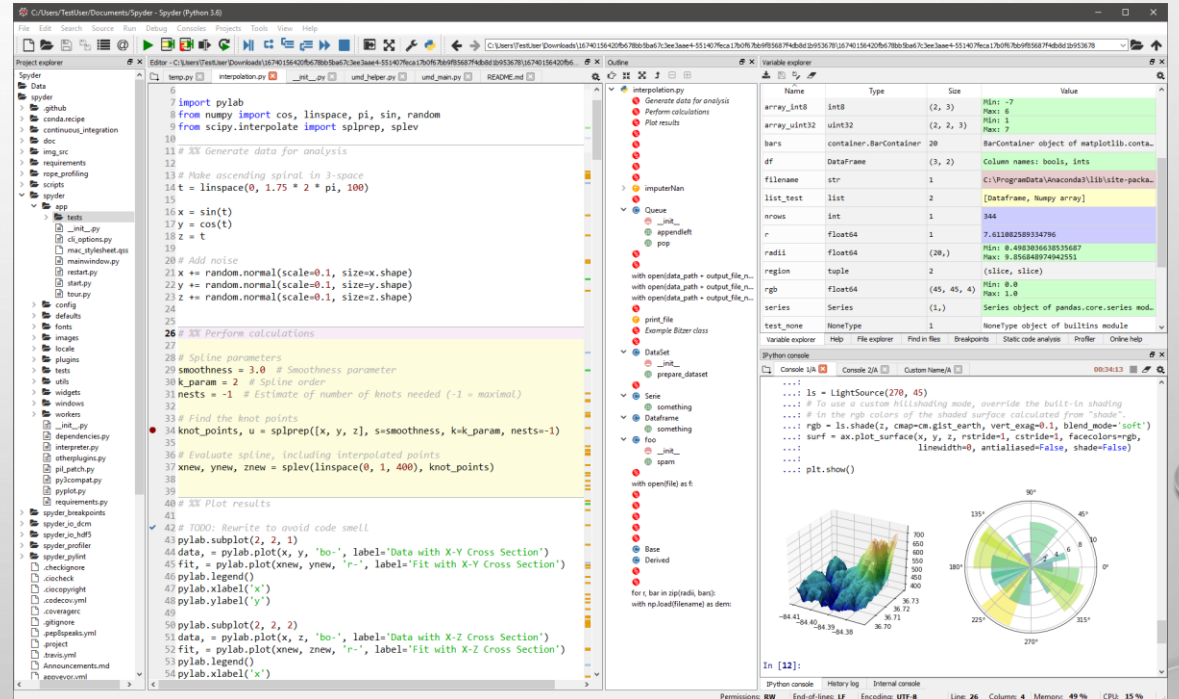
- 侯景勛: JIMHOU28@GMAIL.COM
- 蔡宏育: XBOXCAR9008@GMAIL.COM

REFERENCE BOOKS

- DIGITAL IMAGE PROCESSING 3/E
 - RAFAEL C. GONZALEZ, AND RICHARD E. WOODS
 - PRENTICE HALL
- 數位影像處理: PYTHON程式實作
 - 張元翔
 - 全華

REQUIREMENTS

- BASIC PROGRAMMING SKILLS
 - PYTHON
 - 直譯式、物件導向的高階程式語言
- CROSS-PLATFORM
- ENVIRONMENTS
 - SPYDER WITH ANACONDA
 - 視覺化
- 套件
 - NUMPY
 - OPENCV

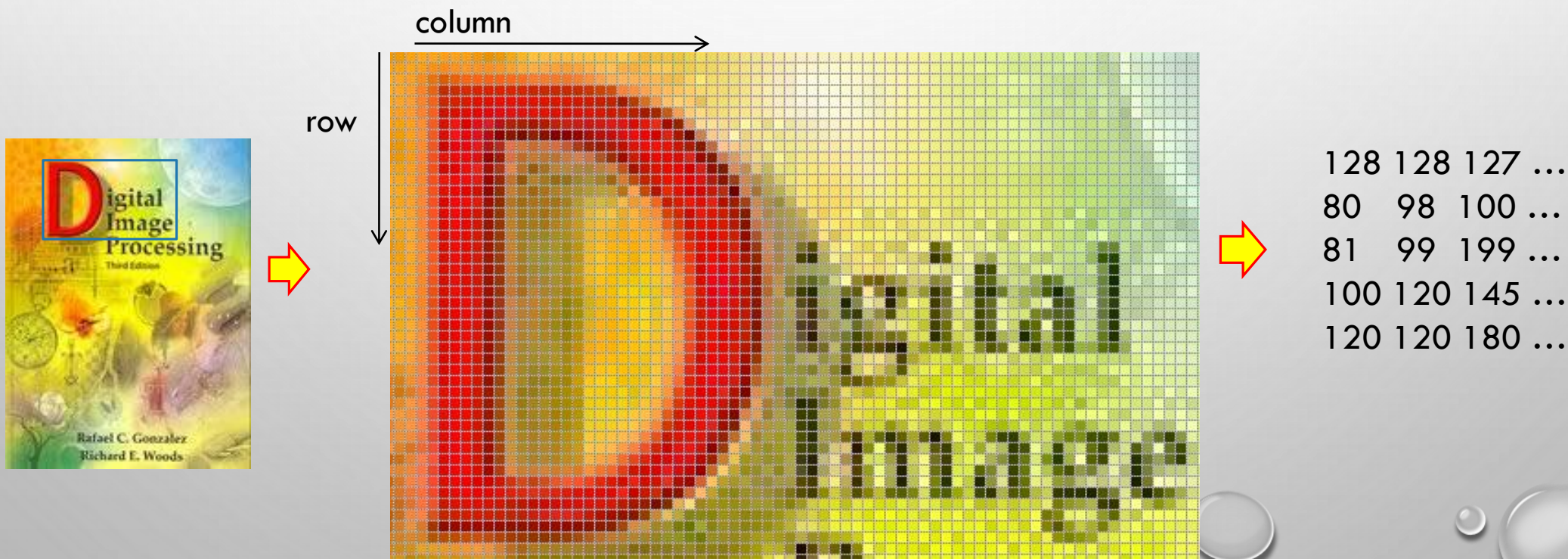


REQUIREMENTS

- 沒有作業!
 - 沒有作業!!
 - 沒有作業!!!
-
- 但會有兩次小考...

做影像處理前的基本認知

- 影像是什麼?
- 影像是由數字構成的陣列(通常是二維或三維)
- 數位影像處理就是數學的運算



基礎PYTHON程式設計之後

- 跨出數位影像處理的第一步：處理數字
- 先練習一下簡單的運算吧
- EX1: 矩陣乘法
- 請寫一個小程序計算兩個 2×2 矩陣的乘法
- EX: $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 4 \\ 1 & 1 \end{bmatrix}$ $A*B = ?$ **$\begin{bmatrix} 2 & 6 \\ 4 & 16 \end{bmatrix}$**
 - 怎麼寫個小乘法就這麼難...
 - 不要氣餒，這就是NUMPY出馬的時候了!

什麼是NUMPY

- PYTHON的套件之一，專門處理數學運算、矩陣運算並支援平行處理
- 提供許多內建的科學計算
- 不只用於影像處理，對於大數據或深度學習也是必要的套件

首先載入NUMPY

```
In [1]: import numpy
```

```
In [2]: import numpy as np
```

```
In [3]: from numpy import *
```

載入numpy

載入numpy並縮寫成np

將numpy中所有函數直接載入
(之後使用時不用再打命名空間名稱)

- 選一行就好!!
- 例如: 使用NUMPY內建的圓周率常數PI

```
numpy.pi  
3.141592653589793
```

```
np.pi  
3.141592653589793
```

```
pi  
3.141592653589793
```

NUMPY提供的部分數學運算

函數名稱	說明	函數名稱	說明
Sign(x)	取正負號	Sin(x)	
Abs(x)	取絕對值	Cos(x)	
Sqrt(x)	取根號	Tan(x)	
Log(x)	取自然對數	Arcsin(x)	
Log10(x)	取對數(以10為底)	Arccos(x)	
Log2(x)	取對數(以2為底)	Arctan(x)	
Exp(x)	取指數 e^x	Min(x)	
Ceil(x)	取頂值 4.5->5	Max(x)	
Floor(x)	取底值 4.5->4	Degrees(x)	
Round(x,d)	四捨五入(到d位)	Radians(x)	

NUMPY運算

- EX2: 參考課本的2-10 2-11，計算以下算式的值
 - $\text{FLOOR}(4.5) + \text{FLOOR}(-4.5)$
 - $\text{CEIL}(4.5) + \text{CEIL}(-4.5)$
 - $20 * \text{ARCTAN} \frac{1}{7} + 8 * \text{ARCTAN} \frac{3}{79}$
 - $\sqrt{6 * (\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots)}$

NPARRAY

- NUMPY的最主要運算資料結構：NPARRAY陣列
- 可以為一維、二維或多維

```
In [1]: import numpy as np
```

```
In [2]: y = np.array([1,2,3,4])
```

```
In [3]: print(y.ndim,y.shape,y.dtype)  
1 (4,) int32
```

```
In [4]: z = np.array([[1,2],[3,4],[5,6]])
```

```
In [5]: print(z.ndim,z.shape,z.dtype)  
2 (3, 2) int32
```

- LIST <-> NPARRAY
 - NP.ARRAY(A)
 - A.TOLIST()
- 實作時小心不同函數的輸入不同

NPARRAY

```
In [1]: import numpy as np
```

```
In [2]: x = np.array([1,2,3,4])
```

```
In [3]: y = np.array([5,6,7,8])
```

```
In [4]: x+y
```

```
Out[4]: array([ 6,  8, 10, 12])
```

```
In [5]: x-1
```

```
Out[5]: array([0, 1, 2, 3])
```

```
In [6]: x-y
```

```
Out[6]: array([-4, -4, -4, -4])
```

```
In [7]: x*y
```

```
Out[7]: array([ 5, 12, 21, 32])
```

載入NUMPY，重新命名為NP

宣告NPARRAY X

宣告NPARRAY Y

執行X+Y

執行X-1

執行X-Y

執行X*Y !!

NARRAY運算

- 這好像還是不是我們常用的矩陣乘法...

- USE NP.MATMUL

- EX3: 利用MATMUL計算EX1的問題

- $A = [1 \ 2, \ 3 \ 4] \ B = [0 \ 4, \ 1 \ 1] \ A*B = ?$

統計分析

- 平均值MEAN,標準差 STD
- EX4: 執行右邊程式並觀察說明結果

```
In [7]: runfile('C:/U
5.5
2.8722813232690143
10
[1 2 3]
5.5
2.8722813232690143
[ 7  8  9 10]
[15 40]
[ 7  9 11 13 15]
```

```
import numpy as np

x = np.array([1,2,3,4,5,6,7,8,9,10])

print(np.mean(x))
print(np.std(x))
print(np.max(x))
print(x[x < 4])

y = np.array([[1,2,3,4,5],[6,7,8,9,10]])
print(np.mean(y))
print(np.std(y))
print(y[y>6])
print(y.sum(axis=0))
print(y.sum(axis=1))
```

到影像處理的最後一哩路

- `NP1 = NP.ZEROS([1024,1024], DTYPE="UINT8")`
- `NP2 = NP.ONES([1024,1024], DTYPE="UINT8")`
- 宣告全部為0 或全部為1的陣列，長寬為1024,1024，資料型態為UINT8
- 影像處理時常使用UINT8(強度範圍為0-255)或FLOAT(強度範圍為0-1)

強度範圍

- EX5: 執行下面程式看看

```
import numpy as np
import cv2

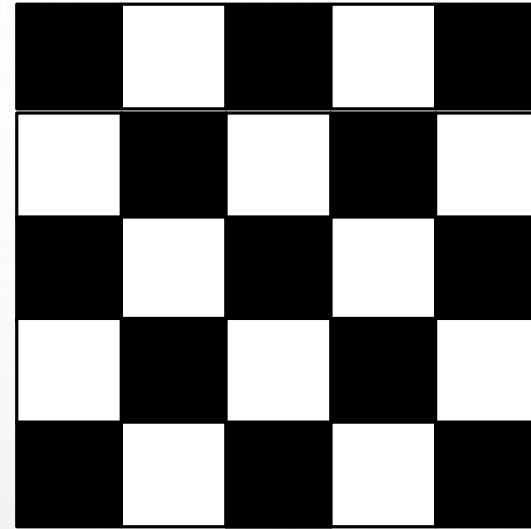
img1 = np.zeros([512,512],dtype="uint8")
img2 = np.ones([512,512],dtype="float32")

cv2.imshow("test1",img1)
cv2.imshow("test2",img2)
cv2.waitKey()
```

- 將FLOAT32改成UINT8再執行看看

PRACTICE

1. 生出500*500的黑白影像如右圖：



2. 生出500*500的黑白影像並繪製出 $Y = x^{2/3}$ 的曲線圖

(提示：計算出Y之後在陣列中(x,Y)的位置填入1 OR 255)

(提示：超過範圍可以不畫，或者是調整Y軸的比例)

(提示：想要讓線變粗可以把附近的幾格也填入顏色)