

Python影像處理：可見光訊號解碼

Recall

Practice8-2我們如何將數字轉換輸出？

Ex. 111→0110 1111 我們存入bits string 但是方向剛好相反 1111 0110

之後1111 轉換RLL(0,2) →01111

0110轉換RLL(0,2) →10110

存成bitsRLL→0111110110

灌入Header 10101000

輸出→101010000111110110



0110111100010101
0001101111000010101



00011011111000010101 由右往左看

Header:10101000

第一個訊息:01111→查表轉換→1111→倒過來→1111→15

第二個訊息:10110→查表轉換→0110→倒過來→0110→6

得到訊息 $6*16+15=111$

RLL(0,2)

原始位元組	編碼後位元組	原始位元組	編碼後位元組
0000	11001	1000	11010
0001	11011	1001	01001
0010	10010	1010	01010
0011	10011	1011	01011
0100	11101	1100	11110
0101	10101	1101	01101
0110	10110	1110	01110
0111	10111	1111	01111

影像處理流程



影像處理流程



轉成灰階影像

```
1  from PIL import Image
2  import numpy as np
3  import cv2
4  import scipy.misc
5  import csv
6  from numpy import savetxt
7
8
9  with Image.open('111.jpg') as img:
10     img = np.array(img)
11     picture = img[:, :, 0]
```


影像處理流程



計算條紋寬度

可參考 `practice6-2_finding_width.py`

```
13 pixel=1
14 maxi=0
15
16 #finding how many pixels in each line
17 test=5
18 ▼ while test<50:
19     one = np.ones((test,10),np.float32)
20     minusone=one*-1
21     lap=np.vstack((one,minusone))/test/10
22     lap2=np.vstack((lap,lap))/2
23     lap3=np.vstack((lap,lap2))/2
24     result = cv2.filter2D(picture,-1,lap3)
25     maxx=np.max(result)
26 ▼     if maxx>maxi:
27         pixel=test
28         maxi=maxx
29         test=test+1
30
31 print ('Pixel width is:',pixel)
```

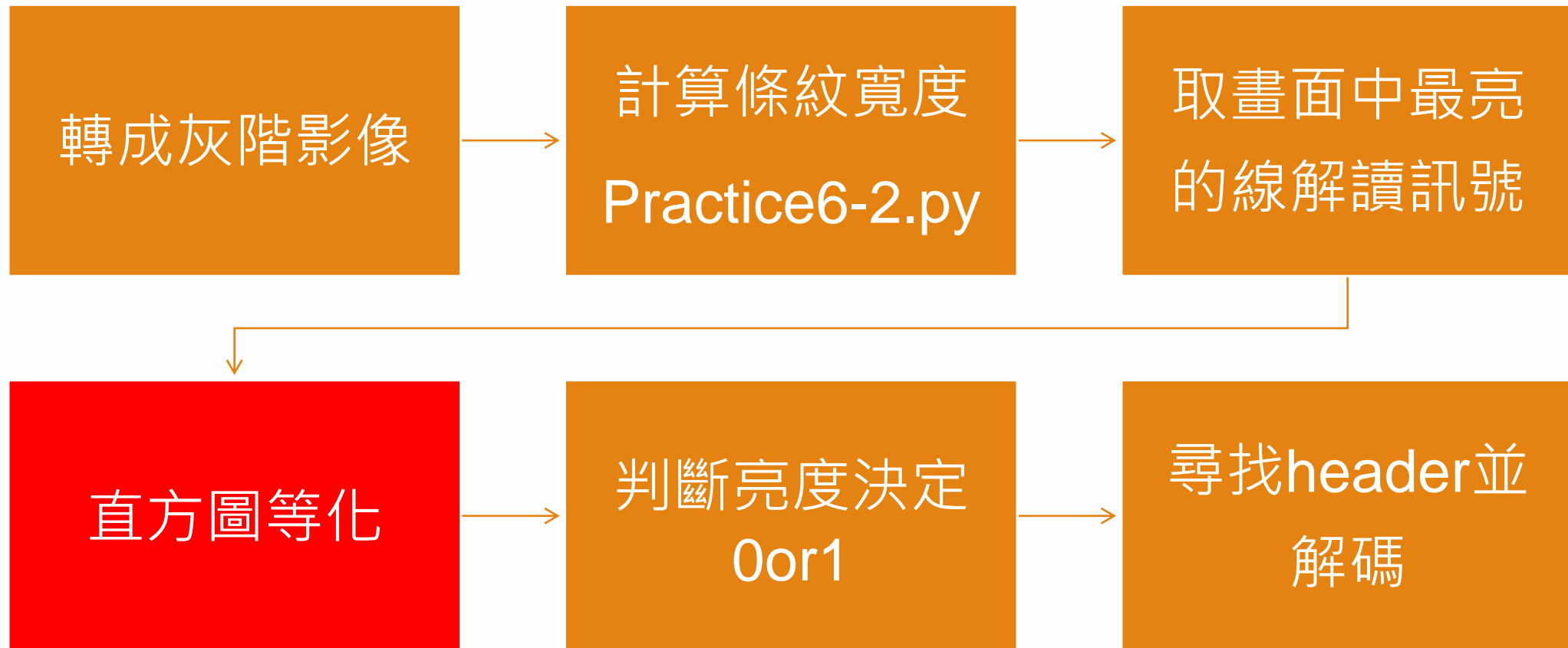
影像處理流程



取畫面中最亮的線解讀訊號

```
35  #Finding the brightest line to decode
36  bright=0
37  total=0
38  keep=0
39  ▼ for i in range(0,3264,1):
40      for j in range(0,2448):
41          total+=picture[j,i]
42  ▼  if(total>keep):
43      bright=i
44      keep=total
45      total=0
46  print ('The brightes row is',bright)
```

影像處理流程



直方圖等化

```
49  pic = cv2.equalizeHist(image)
```

影像處理流程



判斷亮度決定0or1

```
51  #Decode 0 or 1
52  a=""
53  ▼ for i in range(0,2448,pixel):
54      if pic[i,bright]<150:
55          a+=str(0)
56      else :
57          a+=str(1)
58  print ('Decode number:',a)
```


影像處理流程



尋找header並解碼

```
60 #Decode header and find the answer
61 k=0
62 j=0
63 tenth=0
64 oneth=0
65 ans=0
66 ▼ RLL = {
67     '11001' : 0, '11010' : 1, '11101' : 2, '11110' : 3,
68     '10010' : 4, '01010' : 5, '10110' : 6, '01110' : 7,
69     '11011' : 8, '01001' : 9, '10101' : 10, '01101' : 11,
70     '10011' : 12, '01011' : 13, '10111' : 14, '01111' : 15
71 };
72 ▼ while k>=0:
73
74     k=a.find('10101000',j)
75     if k==-1:
76         break;
77     if k+18>len(a):
78         break;
79 ▼     try:
80         ans=RLL[a[k+8:k+13]]+16*RLL[a[k+13:k+18]]
81         print (ans)
82     except:
83         print ('RLL error')
84     j=k+1
```