

# 影像處理 Assignment1 - 合併影像

姓名：吳嘉偉 學號：5105056013 日期：2017/12/10

## 1 合併步驟

### 1.1 三點座標

找出兩張圖對應的三點座標，此三點圍成的面積越大越好

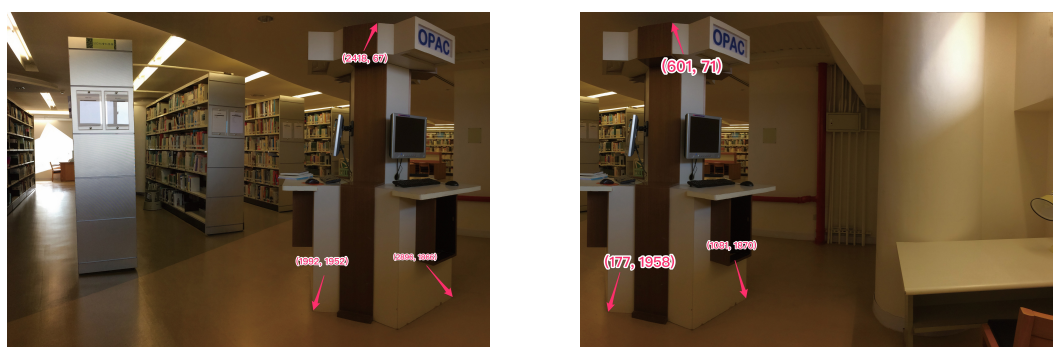


Figure 1: combine pictures

## 1.2 六參數

用上面兩張圖的 3 個座標，透過  $Ax=B$  找出六參數

矩陣

$$\begin{pmatrix} 2418 & 67 & 1 \\ 1992 & 1952 & 1 \\ 1896 & 1866 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} 601 \\ 177 \\ 1081 \end{pmatrix}$$

$$\begin{pmatrix} 2418 & 67 & 1 \\ 1992 & 1952 & 1 \\ 1896 & 1866 & 1 \end{pmatrix} \begin{pmatrix} d \\ e \\ f \end{pmatrix} = \begin{pmatrix} 71 \\ 1958 \\ 1870 \end{pmatrix}$$

程式碼

# 取得六參數

```
def getFunctionParaOfX():  
    a = np.array([[2418, 67, 1],  
                  [1992, 1952, 1],  
                  [2896, 1866, 1]])  
    b = np.array([601, 177, 1081])  
    x = np.linalg.solve(a, b)  
    return x
```

def getFunctionParaOfY():

```
    a = np.array([[2418, 67, 1],  
                  [1992, 1952, 1],  
                  [2896, 1866, 1]])  
    b = np.array([71, 1958, 1870])  
    y = np.linalg.solve(a, b)  
    return y
```

# 轉換座標

```
def getTransferCoordinate(X, Y, x, y):  
    newX = X[0] * x + X[1] * y + X[2]  
    newY = Y[0] * x + Y[1] * y + Y[2]  
    return (newX, newY)
```

### 1.3 Bilinear Interpolation

因為輸出像素對應的地方，不一定是輸入圖像某個整數像素位置，這時要用整數座標的灰度值進行推斷

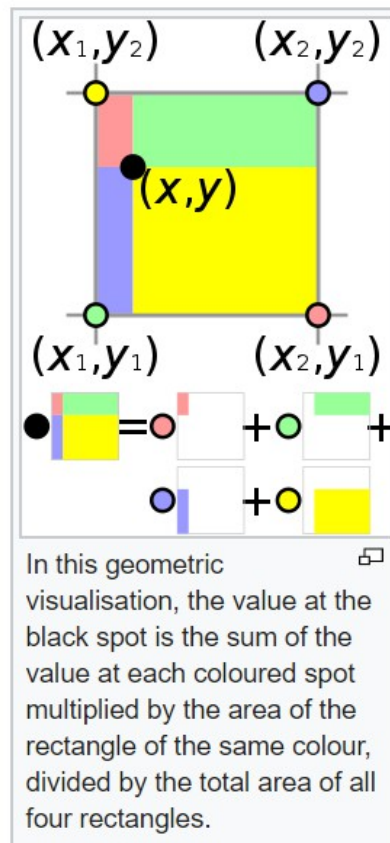


Figure 2: Bilinear

程式碼

```
# 做 Bilinear Interpolate
def getBilinearInterpolate(im, x, y):
    x = np.asarray(x)
    y = np.asarray(y)

    x0 = np.floor(x).astype(int)
    x1 = x0 + 1
    y0 = np.floor(y).astype(int)
    y1 = y0 + 1
```

```

x0 = np.clip(x0, 0, im.shape[1] - 1)
x1 = np.clip(x1, 0, im.shape[1] - 1)
y0 = np.clip(y0, 0, im.shape[0] - 1)
y1 = np.clip(y1, 0, im.shape[0] - 1)

imA = im[y0, x0]
imB = im[y1, x0]
imC = im[y0, x1]
imD = im[y1, x1]

wa = (x1 - x) * (y1 - y)
wb = (x1 - x) * (y - y0)
wc = (x - x0) * (y1 - y)
wd = (x - x0) * (y - y0)

result = wa * imA + wb * imB + wc * imC + wd * imD
return result

```

## 1.4 合併圖片

先產生一張可以包含合併後圖片大小的圖，再用兩個 for-loop 跑長寬，如果座標是在照片一內，則直接輸出照片一的像素。

如果不再照片一的範圍，則透過轉換座標判斷是否在照片二的座標內，如果是，則透過 Bilinear Interpolate 算出該像素值，如果不是則給黑色。

程式碼

```

# 處理新照片
print('start_process_photo')
for x in range(width):
    for y in range(height):
        if y < 2227 and x < 3147:
            newImage[y][x] = getBilinearInterpolate(im1, x, y)
        else:
            newX, newY = getTransferCoordinate(X, Y, x, y)

```

```
if 0 <= newY < 2227 and 0 <= newX < 3147:  
    newImage[y][x] = getBilinearInterpolate(im2,  
  
    else:  
        newImage[y][x] = 0  
print('process_photo_done')
```

## 2 結論

一開始拍攝兩張照片，但角度問題會造成要合併的物件大小在兩張照片是不相同，導致兩張照片合併時會有大小的落差。

所以後來改成先拍一張大圖，裁切成兩張後進行合併。

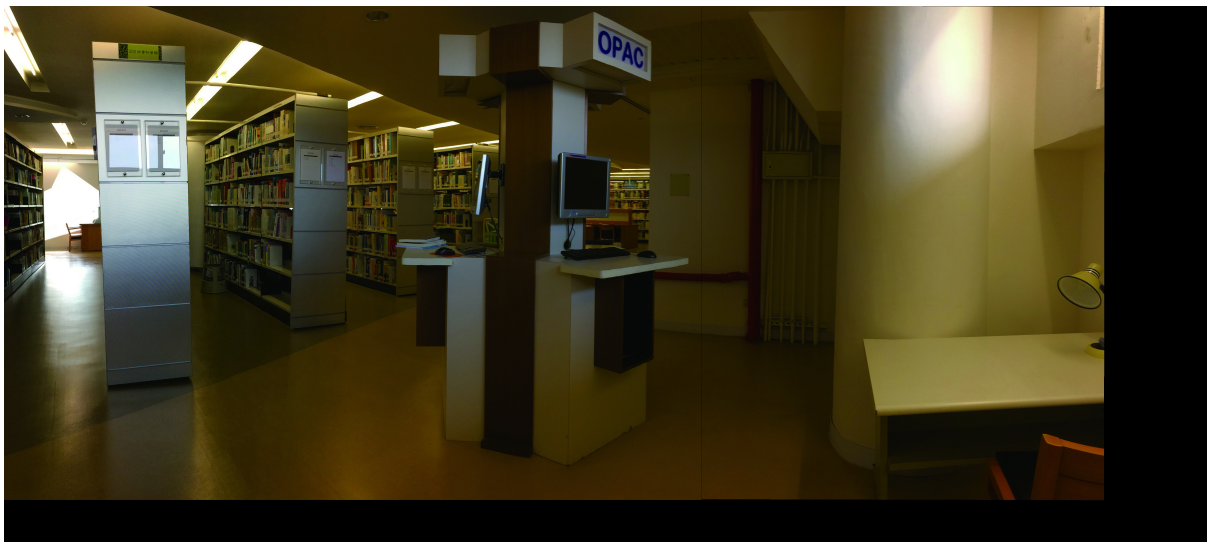


Figure 3: 合併影像