



COMPUTER VISION

Homework 02

姓名：蘇宛琳

系所：電信所碩一

學號：R05942060

指導教授：傅楸善老師

Computer Vision Report – Homework 02

R05942060 蘇宛琳

➤ Write a program to generate Image Matching (Detecting Motion Vectors)

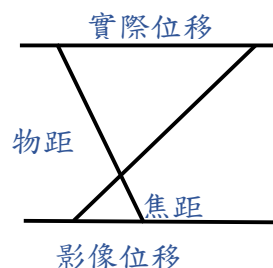
- ✚ Camera calibration i.e. compute #pixels/mm object displacement
- ✚ Use lens of focal length: 16mm, 25mm, 55mm
- ✚ Object displacement of: 1mm, 5mm, 10mm, 20mm
- ✚ Object distance of: 0.5m, 1m, 2m
- ✚ Camera parameters: 23.7mm * 15.6mm ==> 4672*3104 pixels
- ✚ Are pixels square or rectangular?
- ✚ Calculate theoretical values and compare with measured values.
- ✚ Calculate field of view in degrees of angle.

➤ 設計原理

✚ 影像位移理論值計算

(1) 透過下圖的原理，可以經由相似三角形得到以下的公式:

$$\text{影像位移} = (\text{焦距} * \text{實際位移}) / \text{物距}$$

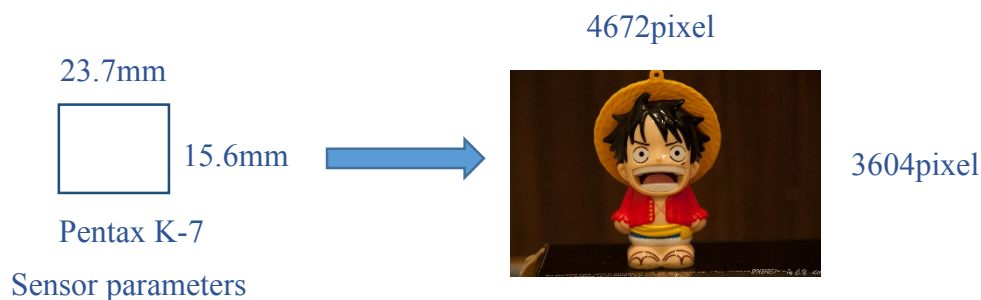


舉例說明: 焦距=18mm，實際位移=1mm，物距=600mm

$$>> \text{影像位移} = (18 * 1) / 600 = 0.03\text{mm}$$

(2) 在換算影像位移(mm)和影像位移(pixel)時，我們必須先得知相機的感應器大小和影像大小，可得比例尺。

$$\text{比例尺} = 4672\text{pixel} / 23.7\text{mm}$$

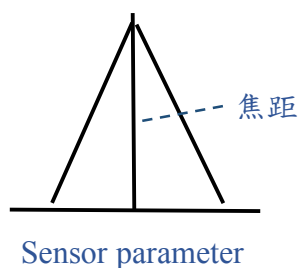


舉例說明：影像位移為 0.03 mm $\gg (4672/23.7)*0.03 = 6$ pixel

✚ 視野夾角計算

✓ 感應器大小 = w ，焦距 = f ，視野夾角 θ 的計算，如下式

$$\theta = \tan^{-1}\left(\frac{f}{w}\right)$$



舉例說明：

$$d = 23.7 \text{ mm}, f = 18 \text{ mm}, \theta = \tan^{-1}\left(\frac{f}{w}\right) = \tan^{-1}\left(\frac{18}{23.7/2}\right) = 56.64 \text{ 度}$$

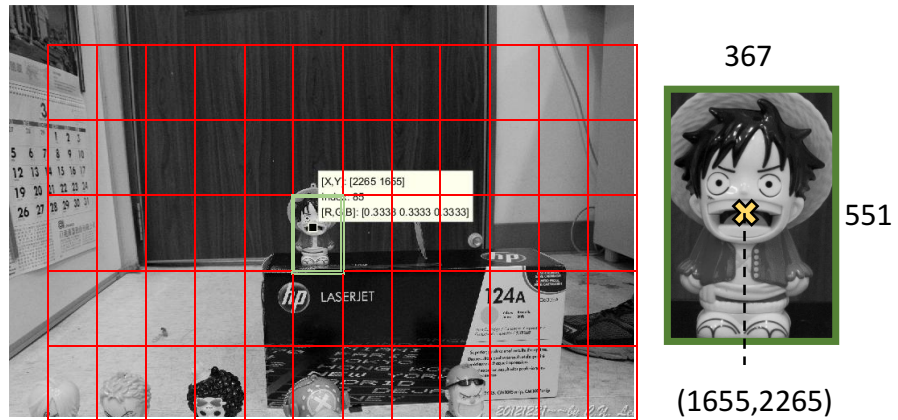
✚ 影像位移計算值計算

Step1. 先利用 MATLAB 中的 Data Curaor 取出魯夫的 **center** 大約位置

還有依據影像中魯夫的大小，來取設定 **block** 大小。

舉例說明：焦距(18mm) 物距(600) 的影像

假設中心 **center**=(2265,1655)，**block**=551x371



Step2. 接著透過第一堂課所教的 correlation 方式，來比較兩張圖魯夫位置的
 水平位移的 pixel 數，且因為透過 Data Curaor 所取出的
 中心位置並不準確，故將其周圍約 10 個 pixel 左右皆設為中心
 點，分別與圖 b 進行 correlation，找出 correlation 的最小值，即
 可求得最準確之 a 圖中心點，並同時設定 search range > 位移理
 論值*2，此時計算與圖 b 最佳的中心點的位移，如此一來不用
 跑整張影像的向量圖，就可以輕鬆針對魯夫的移動來求得計算
 值。

舉例說明：

圖 a 為焦距(18mm) 物距(600) 位移(0)

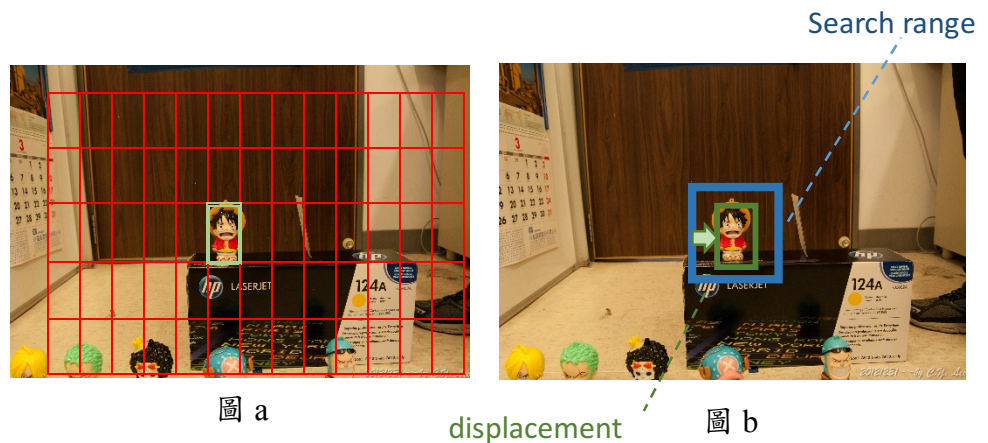
圖 b 為焦距(18mm) 物距(600) 位移(20)

取周圍 10 個點為中心點，故圖 a 的 center=(1650~1659,2260~2269)

Step1.

Center(1)=(1650,2260) Block=551x371

Search range=490 > 位移理論值*2=120*2=240



計算 correlation --> dmin(1)-->dx(1) , dy(1)

接著依此類推計算到 center(10) , 計算 correlation --> dmin(10)

-->dx(10),dy(10)

Step2.

取得最小的 dmin(k) , 則 center(k)最為準確 , 且所得到的 dx(k) , dy(k) , 即為影像位移計算值。

➤ 程式碼

```
%%% use correlation to do image matching a
a=imread('600mm_0mm.jpg');a=rgb2gray(a);a=double(a);
b=imread('600mm_5mm.jpg');b=rgb2gray(b);b=double(b);
%%% assume the block and search size and center
% block size
Bx=551; hBx=fix(Bx/2);
By=371; hBy=fix(By/2);
% search size ( hS=10 is equal that search size is 20)
hS=10;
% padarray zeros in image a and b
x=1650:1654;
y=2272:2276;
for i=1:length(x)
    d=[];
    for r=x(i)-hS:x(i)+hS
        for c=y(i)-hS:y(i)+hS
            d(r-x(i)+hS+1,c-y(i)+hS+1)=norm(a(x(i)-hBx:x(i)+hBx,y(i)-hBy:y(i)+hBy)-b(r-hBx:r+hBx,c-hBy:c+hBy));
        end
    end
    [~,row1{i}]=min(d);
    [dmin(i),col(i)]=min(min(d));
    row(i)=row1{i}(col(i));
end
[~,k]=min(dmin);
```

```

xc=x(k);
yc=y(k);
dx=row(k)-hS-1;
dy=col(k)-hS-1;

```

➤ 實驗結果

焦距 (mm)	物距 (mm)	實際 位移 (mm)	影像 位移 理論值 (mm)	影像 位移 理論值 (pixel)	影像 位移 計算值 (pixel)	視野 夾角 (度)	採用 Center Block
18	600	1	0.030	6	7	56.64	Center
		5	0.150	30	26		(1650,2272)
		10	0.300	60	54		Block
		20	0.600	120	101		551x371
	1200	1	0.015	3	2		Center
		5	0.075	15	16		(1612,2339)
		10	0.150	30	30		Block
		20	0.300	60	55		291x201
	1800	1	0.010	2	3		center
		5	0.050	10	9		(1590,2324)
		10	0.100	20	19		Block
		20	0.200	40	45		191x141
53	600	1	0.088	17	21	77.40	center
		5	0.441	87	61		(1670,2277)

		10	0.883	174	128		Block 1401x1001
		20	1.766	384	252		
	1200	1	0.044	9	10		center (1630,2400) Block 771x501
		5	0.220	44	45		
		10	0.441	87	78		
		20	0.883	174	162		
	1800	1	0.029	6	12		center (1322,2400) Block 521x342
		5	0.147	29	37		
		10	0.294	58	58		
		20	0.588	116	101		
135	600	1	0.225	44	50	85.98	center (1597,2221) Block 2531x1801
		5	1.125	222	159		
		10	2.250	444	430		
		20	4.500	887	788		
	1200	1	0.113	22	23		center (1550,2500) Block 1650x1101
		5	0.563	111	109		
		10	1.125	222	200		
		20	2.250	444	414		
	1800	1	0.075	15	16		center (1140,2325) Block 1201x801
		5	0.375	74	78		
		10	0.750	148	149		
		20	1.500	296	310		