

Advanced Computer Vision Homework2

生醫電資所 碩一 R07945029 王思敏

Objective:

Calculate the equivalent distance of the real world in 1 pixel of the digital camera and FOV value.

Parameters:

1. Focal: 18mm, 53mm, 135mm
2. Objective length: 600mm, 1200mm, 1800mm
3. Motion: 0mm (basis), 1mm, 5mm, 10mm, 20mm

Method:

Find the circle shape in the picture and capture the motion of the center of circle.

1. Set parameters

```
focalpath = {'18mm'; '53mm'; '135mm'};  
obj = {'600mm'; '1200mm'; '1800mm'};  
shi = {'0mm'; '1mm'; '5mm'; '10mm'; '20mm'};  
objmm = [600, 1200, 1800];  
shimm = [0, 1, 5, 10, 20];  
focalmm = [18, 53, 135];  
Allrange = {[5 45]; [3 45]; [5 45]};
```

2. Set the circle shape search range within pictures

```
AllinterstArea(:, :, 1) = [1250 1530 2000 2650; 1400 1550 2200 2500; 1400 1535 2250 2450];  
AllinterstArea(:, :, 2) = [1820 2030 2180 2550; 1700 1900 2350 2550; 1390 1460 2350 2500];  
AllinterstArea(:, :, 3) = [1850 2300 2050 2680; 1750 2050 2380 2800; 1280 1450 2200 2500];
```

3. Read image and intercept the interested range

```
for f = 1:3  
    interstArea = AllinterstArea(:, :, f);  
    focal = focalmm(f);  
    Focal = sprintf('%s_%d', 'Focal', focal);  
    range = Allrange{f, 1};  
  
    img = [];  
    imgbound = [];  
    for i = 1:3  
        for j = 1:5  
            temp = rgb2gray(imread(['Photo/' focalpath{f} '/' obj{i} '_' shi{j} '.jpg']));  
            img{i, j} = temp;  
            imgbound{i, j} = temp(interstArea(i, 1):interstArea(i, 2), interstArea(i, 3):interstArea(i, 4));  
        end  
    end  
end
```

4. Find the center of the circles

```
for i = 1:3
    for j = 1:5
        [centers, radii, metric] = imfindcircles(imgbound{i,j},range);
        centersStrong5 = centers(1,:);
        radiiStrong5 = radii(1,:);
        metricStrong5 = metric(1,:);
        allcenter{i,j} = centersStrong5;
        allradius{i,j} = radiiStrong5;
        allmetric{i,j} = metricStrong5;
        figure(count)
        imshow(imgbound{i,j})
        viscircles(centersStrong5, radiiStrong5, 'EdgeColor', 'b')
        count = count + 1;
    end
end
```

5. Find the horizontal displacement (pixel)

```
move = [];
moveavg_X = [];
moveavg_Y = [];
for i = 1:3
    for j = 1:4
        move{i,j} = abs(allcenter{i,j+1} - allcenter{i,1});
        if f == 1
            avg = abs(allcenter{i,j+1} - allcenter{i,1});
        else
            avg = mean(abs(allcenter{i,j+1} - allcenter{i,1}));
        end
        moveavg_X(i,j) = avg(1,1);
        moveavg_Y(i,j) = avg(1,2);
    end
end
```

6. Calculate mm/pixel

```
mm_pixel = [];
for i = 1:3
    mm_pixel(i,:) = shimm(1,2:5) ./ moveavg_X(i,:);
end
```

7. Calculate FOV estimated value and theoretical value

```
FOV_measure = [];
for i = 1:3
    FOV_measure(i,:) = 2*atan(size(img{1,1},2).*mm_pixel(i,:)/(2*objmm(i)))*180/pi;
end

FOV_theory = 2*atan(23.4/2/focal)*180/pi;
```

8. Results

Focal = 18mm

焦距(mm)	18														
物距(mm)	600					1200					1800				
位移(mm)	0	1	5	10	20	0	1	5	10	20	0	1	5	10	20
位移(pixel)	0	6.78	25.4	53.97	98.65	0	2.53	16.27	30.46	54.62	0	3.26	9.04	18.34	44.36
mm/pixel	0	0.15	0.2	0.19	0.2	0	0.4	0.31	0.33	0.37	0	0.31	0.55	0.55	0.45
FOV 估計值	0	59.72	74.93	71.61	76.57	0	75.18	61.78	65.16	70.97	0	43.36	71.34	70.58	60.66
FOV 理論值	66.05														

Focal=53mm

焦距(mm)	53														
物距(mm)	600					1200					1800				
位移(mm)	0	1	5	10	20	0	1	5	10	20	0	1	5	10	20
位移(pixel)	0	21.95	63.22	130.41	264.66	0	9.7	44.32	77.4	161.78	0	12.11	37.11	58.61	101.9
mm/pixel	0	0.05	0.08	0.08	0.08	0	0.1	0.11	0.13	0.12	0	0.08	0.13	0.17	0.2
FOV 估計值	0	20.11	34.23	33.24	32.79	0	22.69	24.77	28.24	27.06	0	12.24	19.84	24.97	28.59
FOV 理論值	24.9														

Focal=135mm

焦距(mm)	135														
物距(mm)	600					1200					1800				
位移(mm)	0	1	5	10	20	0	1	5	10	20	0	1	5	10	20
位移(pixel)	0	60.08	161.58	265.28	503.95	0	36.71	62.48	144	313.09	0	3.96	33.58	88.25	212.5
mm/pixel	0	0.02	0.03	0.04	0.04	0	0.03	0.08	0.07	0.06	0	0.25	0.15	0.11	0.09
FOV 估計值	0	7.42	13.74	16.7	17.57	0	6.07	17.7	15.4	14.18	0	36.3	21.87	16.73	13.93
FOV 理論值	9.91														