Visió Tridimensional
Helena Cots, 183951
Paula Castillo, 184646

Pràctica 3 : Camera calibration using a planar pattern

Els resultats obtinguts en les respostes següents són trobats a partir de les nostres imatges:

Imatges del món real

Question 1. Complete the code to compute the IAC from the homographies.

Per a calcular la *Image of the Absolute Conic w* utilitzem les línies de codi següents:

```
[U, S, V] = svd(A);
omega = V(:,length(V));
w = [omega(1),omega(2),omega(3);
             omega(2),omega(4),omega(5);
             omega(3),omega(5),omega(6)];
```

On A és el sistema d'equacions creat a partir de les homografies $\mathbf{h}_i$ basades en les correspondències entre les imatges (extretes amb SIFT i RANSAC - DLT algoritme). Volem obtenir la IAC w, és a dir, saber les seves 6 incògnites:

$$\mathbf{w} = \begin{pmatrix} w11 & w12 & w13 \\ w12 & w22 & w23 \\ w13 & w23 & w33 \end{pmatrix}$$

i denominem Ω com la última columna de V (vector nul si V és ideal) tal que:

Ω = (w11 w12 w13 w22 w23 w33).

Un cop hem trobat Ω, col·loquem les variables del vector corresponents per a formar la matriu w.

Question 2. How many images are required to compute the IAC? What would you do if you had fewer images?

Per a computar l'IAC necessitem tres punts de vista diferents, ja que cada una de les imatges resulta en dues equacions per a poder resoldre els sis paràmetres de K. K representa els paràmetres interns de la càmera que s'utilitzen per a fer el calibratge de la càmera.

$$K = \begin{pmatrix} ax & s & px \\ 0 & ay & py \\ 0 & 0 & 1 \end{pmatrix}$$

Tot i així, si es tinguessin menys de tres imatges (tres vistes), es podrien realitzar assumpcions sobre aquests paràmetres el quals la majoria de càmeres que es troben al mercat comparteixen.

$$\alpha x = \alpha y,\ s = 0.$$

Amb les dues equacions afegides, eliminaríem dues incògnites pel que només necessitaríem dues imatges. Si, a més a més, el punt (px,py), que sol ser sempre el mateix, és conegut, només caldria tenir una imatge per a calibrar la càmera.

Compute the camera calibration from the IAC

The IAC relates to the camera calibration matrix, K as $w=K^{-T}K^{-1}$. Knowing w we can get K using the Cholesky factorization.

Question 3. Write the code to compute the camera calibration from the IAC.

Per a computar els paràmetres interns de la càmera a partir de la IAC, fem servir l'algoritme *Cholesky* que descomposa una matriu simètrica positiva-definida en un producte d'una matriu triangular superior i la seva transposada.

```
K = inv(chol(w));
Knorm = K/K(3,3);
```

Question 4. What result do you get? Can you interpret the values of K in geometric terms?

```
1.0e+03 *

  3.0552   -0.0199    1.7020
       0    3.0513    0.9987
       0         0    0.0010
```

- -19.9 → s (factor d'esbiaixament)
- 3055.2 → αx (dist. focal en la dimensió x)
- 3051.3 → αy (dist. focal en la dimensió y)
- (1702, 998.7) = (px,py) (coordenades del punt principal - en píxels)

Com podem veure, l'assumpció prèviament mencionada αx = αy no s'allunya gaire de la realitat quan hem fotografiat amb la càmera del nostre telèfon mòbil.

Visió Tridimensional
Helena Cots, 183951
Paula Castillo, 184646

Question 5. Compute the field of view angle of the camera.

fovx = 2* arctg ($\frac{imageWide}{2*\alpha x}$) → 2*arctg ($\frac{2448}{2*3055.2}$) = 43.66º obertura horitzontal.

fovy = 2* arctg ($\frac{imageHeight}{2*\alpha y}$) → 2*arctg ($\frac{3264}{2*3051.3}$) = 56.28º obertura vertical.

Question 7. Complete the code to compute the rotation and translation of each camera. Use the provided plot camera function to display the results.

Un cop extreta la K, la rotació de la càmera i la translació poden ser recuperades de tal forma:

$$(K^{-1}h_1 \quad K^{-1}h_2 \quad K^{-1}h_3) \sim (r_1 \quad r_2 \quad t)$$

,

sent (**h_1,h_2,h_3**) les projeccions de les imatges que relacionen el patró de calibratge amb les diferents càmeres.

```
h = H{i};
r1 = inv(K)*h(:,1);
r2 = inv(K)*h(:,2);
t{i} = inv(K)*h(:,3);
```

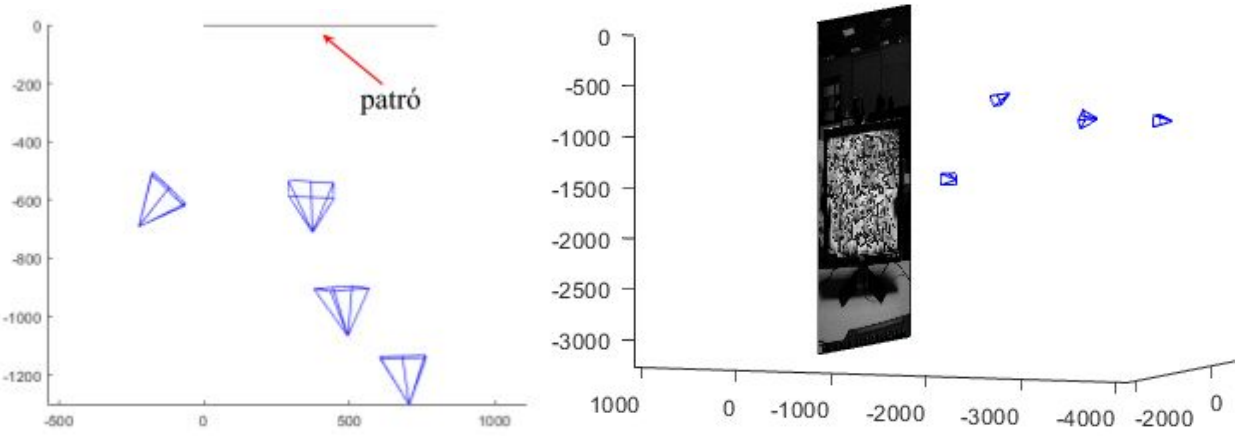
La matriu de rotació de la càmera serà, finalment:

```
R{i} = [r1, r2, cross(r1,r2)];
```

Un cop hem trobat els paràmetres interns (K) i calculades la translació (t) i la rotació de la càmera ®, ja podem computar P = K[R|t] → la matriu de projecció de la càmera.

```
P{i} = K * [R{i} t{i}];
plot_camera(P{i}, 3008, 2000, 200);
```

A partir d'aquest procés podem veure on estava col·locada la càmera i quina inclinació tenia quan es van fer les diferents fotografies al patró.

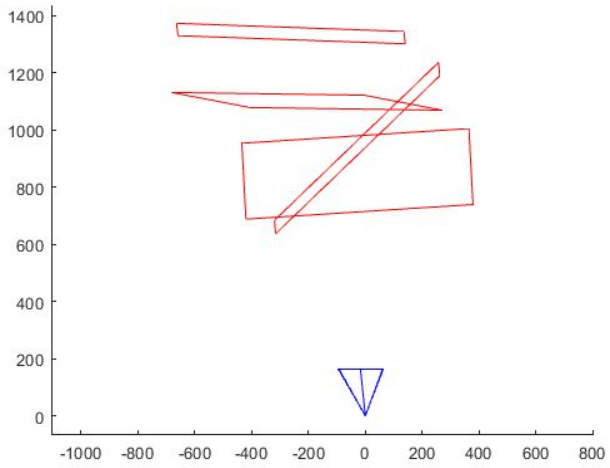


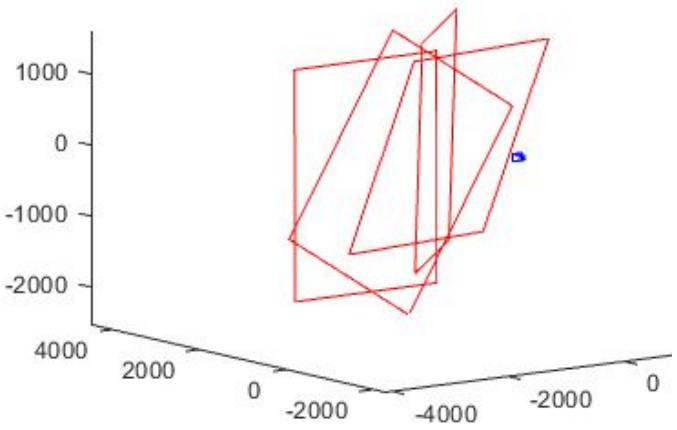
Question 8. In which units are expressed the translation vectors t{i}?

Els vectors de translació es troben expressats en píxels.

Question 9. Write the code to plot the moving planar target in front of a static camera. Draw only the target frame; no need to draw the image texture using the surface command.

```
vgg_scatter_plot([R{i}*pl+t{i}, R{i}*p2+t{i}, R{i}*p3+t{i}, R{i}*p4+t{i}, R{i}*pl+t{i}], 'r');
```





Podem observar que els plans que s'han creat en vermell corresponen als punts de vista de les imatges originals com si, enlloc de moure el punt de la càmera, s'haguéssin transformat les imatges (gràcies a la matriu de transformació projectiva P).

Question 10. When taking the images, we did move the camera and the planar pattern was static. The last plot shows a moving pattern in front of a static camera. What is the relation between the two situations?

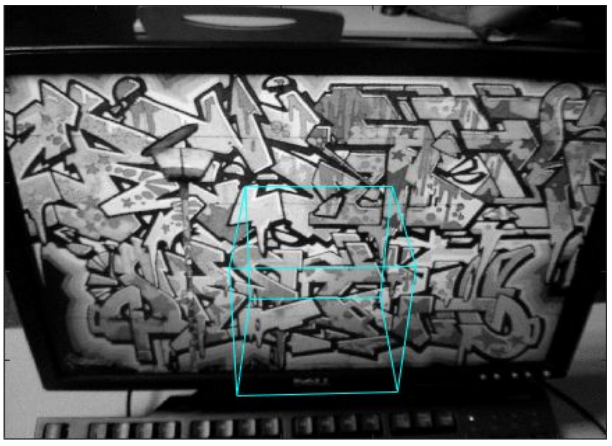
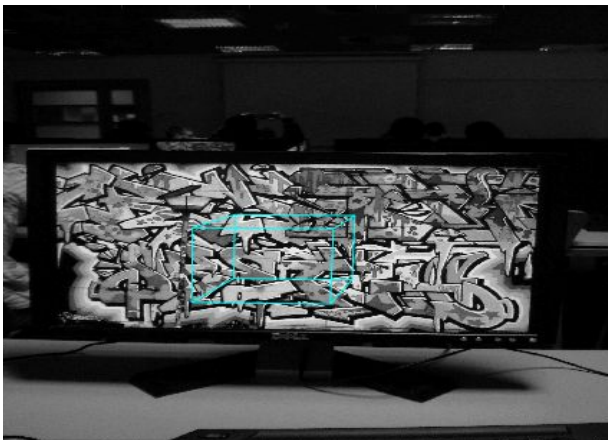
Gràcies a les matrius de projecció de la càmera, podem recrear les vistes que teníem movent la càmera però mantenint-la en un punt de referència fix. Denominant el 'moving pattern' a cada posició *i*, podríem definir com: $\mathbf{x_i} = \mathbf{P}*\mathbf{X}$, on X seria el patró original del món real (3D).

Draw 3D objects over the images

We now have the projection matrices, P, of each image with respect to the planar target's reference frame. We can use these projection matrices for projecting 3D points onto the images. Given a 3D point, we can project it using the projection matrix and plot the projection over the image.

Question 11. Complete the code to plot a cube in front of the calibration pattern on each image. The result should look as if a cube was attached to the pattern as the pattern moves from image to image.

Col·loquem el cub virtual sobre les imatges originals gràcies a les matrius de projecció de la càmera P.

Question 12. Repeat all the process using the five provided images for calibration.

Utilitzant 5 punts de vista (5 imatges) de les que ens donaven, el resultat ha sigut molt bo ja que pot estimar molt millor els paràmetres interns K de la càmera. Tot i així, hem decidit posar els resultats i les imatges amb 3 punts de vista força diferents presos per nosaltres mateixos per a veure com d'eficient és el procediment.

Question 13. Repeat all the process using your own images. Comment the results and try to find an explanation in case it doesn't work.

Fotos sobre el mural

Amb les nostres imatges preses amb el mòbil del mural a la pantalla funciona correctament, tal i com es pot apreciar en aquest informe. Creiem que funciona bé gràcies al template perfecte que tenim ja que és una imatge del ordinador i a que no hem realitzat gaires moviments i angles bruscos amb la càmera i l'hem mantinguda més o menys a una mateixa altura.

Tot i així, ens ha sorprès gratament el resultat obtingut amb una qualitat tan pobre i unes vistes tan distintes entre les imatges.

Fotos sobre les figuretes

Quan hem provat l'algoritme amb les fotografies fetes de les figuretes sobre la taula, al no tenir un *template* "bo" , creiem que SIFT és poc capaç de computar bé les correspondències entre les imatges. Degut a que no tenim prous keypoints per a computar les homografies, l'algoritme no funciona.