COMPUTER VISION LABORATORY

SESSION 6: Fundamental matrix estimation

June 23, 2019

Arlotta Andrea 4089306
Cantoni Francesca 4698289

Contents

Chapter 1

Theoretical information

Before talking about the **8-points algorithm** and analyze our results [1] it is important to make a brief introduction on epipolar geometry, essential matrix **E** and fundamental matrix **F**.

1.1 Epipolar geometry

Epipolar geometry is the projective geometry of a stereo vision [2].

It is possible to demonstrate that, when two cameras view the same 3D scene from two different viewpoints, there exists some geometric relationships between the 3D points and their projections on the 2D images[3].

Thanks to these dependencies the problem of finding correspondences between two image pairs can be reduce from a 2D problem to a 1D one.

[1] These arguments will be discussed in **Chapter 2**.

[2] System that allows multiple observation of the same scene from different viewpoints in order to extract 3D information.

[3] It is important to remark that these relations are derived based on the assumption that the two considered cameras can be approximated by the pinhole camera model.

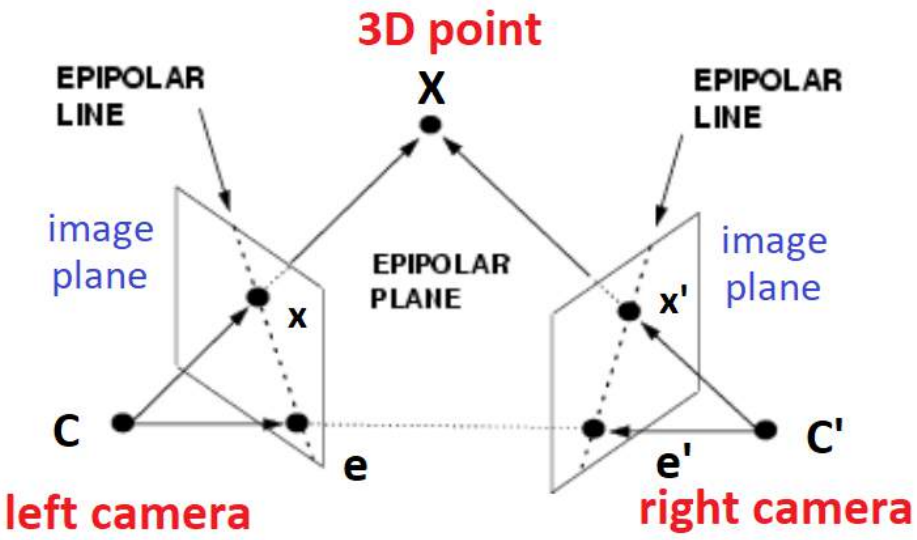

Figure 1.1: Epipolar geometry

TERMINOLOGY:

- **baseline:** line that passes through the two centers of the camera **C** and **C'**
- **epipole:** intersection of the baseline with each one of the image plane (**e** for the left camera and **e'** for the right one)
- **epipolar plane:** plane that contains the baseline and passes contemporary through **C**, **C'** and **X**.
- **epipolar line:** intersection of the epipolar plane with each one of the image plane (**l** considering the left camera and **l'** considering the other one). Is it important to highlights that all epipolar lines contain the *epipole*.

It is important to remark that each one of the 2D projections of the 3D point (called **x** for the left image plane and **x'** for the right one) belong always on the epipolar line (**l** and **l'** respectively). This constraint is called **epipolar constraint** and it is the base idea for most of the relations that will be presented during the next sections.

1.2 Essential matrix

The essential matrix **E** is a 3X3 matrix that allow to find each pair of corresponding points (in mm coordinates) by this relation:

$$\mathbf{x'_m}^{\top}\ \mathbf{E}\ \mathbf{x_m}\ =\ \mathbf{0} \tag{1.1}$$

A 3X3 matrix is an *essential matrix* if and only if two of its singular values are equal and the third is 0.

Properties:

- rank(**E**) = 2
- d.o.f = 5

1.3 Fundamental matrix

The fundamental matrix **F** is obtained by replacing inside equation **2.1** the pixel coordinates in the following way:

$$\mathbf{x_m}\ = \mathbf{K}^{-1}\ \mathbf{x} \tag{1.2}$$

$$\mathbf{x'_m}\ = (\mathbf{K'})^{-1}\ \mathbf{x'} \tag{1.3}$$

(where **K** and **K'** are 3X3 matrices each one represents the intrinsic parameters of the corresponding left or right camera)

Then:

$$\mathbf{x'_m}^{\top}\ \mathbf{E}\ \mathbf{x_m}\ =\ \mathbf{0}\quad \Rightarrow\quad [\mathbf{x'}\ (\mathbf{K'})^{-1}]^{\top}\ \mathbf{E}\ [\mathbf{K}^{-1}\ \mathbf{x}] = \mathbf{0} \tag{1.4}$$

$$\mathbf{x'}^{\top}\ \mathbf{F}\ \mathbf{x}\ = \mathbf{0} \tag{1.5}$$

So the fundamental matrix **F** is the unique 3X3 matrix, with rank(**F**)=2, that satisfy this equation for each corresponding pair (**x**, **x'**):

$$\mathbf{F}\ =\ (\mathbf{K'})^{-\top}\ \mathbf{E}\ \mathbf{K}^{-1} \tag{1.6}$$

We can now conclude that matrix **F** is a map between points and epipolar lines because, recalling that a point belong to a line if and only if $\mathbf{point}^\intercal * \mathbf{line} = \mathbf{0}$ (where **point** must to be express into homogeneous coordinates), we have that:

$$\mathbf{x'}^\intercal \ \mathbf{F} \ \mathbf{x} \ = \mathbf{0} \ \leftrightarrows \ \mathbf{l'} \ = \ \mathbf{F} \ \mathbf{x} \tag{1.7}$$

Properties:

- rank(**F**) = 2
- d.o.f = 7
- the transpose of **F** models the camera pairs in the opposite way because:

$$\mathbf{x}^\intercal \ \mathbf{F}^\intercal \ \mathbf{x'} \ = \mathbf{0} \ \leftrightarrows \ \mathbf{l} \ = \ \mathbf{F}^\intercal \ \mathbf{x'} \tag{1.8}$$

- the epipole **e'** is the left annihilator of **F**, and similarly **e** is the right annihilator of **F**, because each one must satisfy:

$$\mathbf{e'}^\intercal \ \mathbf{F} \ = \mathbf{0} \tag{1.9}$$

$$\mathbf{F} \ \mathbf{e} \ = \mathbf{0} \tag{1.10}$$

Chapter 2

Matlab Code

2.1 main.m script

This Matlab script:

1. loads the *Mire1.points* and *Mire2.points*, or *Rubik1.points* and *Rubik2.points* depending on the value of *input* variable as described in the code,

$$P1_{NX2} = \begin{bmatrix} \mathbf{x_i} & \mathbf{y_i} \end{bmatrix} \quad \text{with i=1,..,N}$$

$$P2_{NX2} = \begin{bmatrix} \mathbf{x_j} & \mathbf{y_j} \end{bmatrix} \quad \text{with j=1,..,N}$$

   and arranges each one into homogeneous coordinates as follow:

$$P1_{3XN} = \begin{bmatrix} \mathbf{x_i} & \mathbf{y_i} & 1 \end{bmatrix}^\intercal \quad \text{with i=1,..,N}$$

$$P2_{3XN} = \begin{bmatrix} \mathbf{x_j} & \mathbf{y_j} & 1 \end{bmatrix}^\intercal \quad \text{with j=1,..,N}$$

2. calls the 8-points algorithm[1] and the 8-points algorithm normalized[2] for estimating respectively the fundamental matrix **F1** and **F2** from the homogeneous point **P1** and **P2**

3. displays the resulting epipolar lines obtained by the estimated **F1** and **F2** [3]

[1] See section **2.2**.
[2] See section **2.3**.
[3] See section **2.4**.

2.2 8-points Algorithm

The 8-point algorithm is an algorithm used for the estimation of the fundamental matrix **F**, it is related to a stereo camera pair and it uses a set of 8 (or more) pairs of corresponding points.

Now we will proceed to analyze step-by-step the considered algorithm:

1. write the matrix **A** (with N number of correspondence points)

$$\mathbf{A} = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ x'_2x_2 & x'_2y_2 & x'_2 & y'_2x_2 & y'_2y_2 & y'_2 & x_2 & y_2 & 1 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ x'_Nx_N & x'_Ny_N & x'_N & y'_Nx_N & y'_Ny_N & y'_N & x_N & y_N & 1 \end{bmatrix} \tag{2.1}$$

   in order to be able to solve this homogeneous equation:

$$\mathbf{A}\ \mathbf{f}\ =\ \mathbf{0} \tag{2.2}$$

$$\text{with} \qquad \mathbf{f}\ =\ (f_{11}\ \ f_{12}\ \ f_{13}\ \ f_{21}\ \ f_{22}\ \ f_{23}\ \ f_{31}\ \ f_{32}\ \ f_{33})^{\mathsf{T}} \tag{2.3}$$

   Solve this equation is important because the **f** vector corresponds to the desired **F1** matrix only written as a column vector instead of a matrix one, in fact:

$$\mathbf{F1} = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{23} & f_{31} & f_{33} \end{bmatrix} \tag{2.4}$$

2. due to the fact that we are no considering the minimal problem and that we cold have noisy points and/or noisy correspondences, to solve equation **2.2** we need to use the *Direct Linear Transformation algorithm* that is based on the constraint of minimization of the cost function as follow:

$$\text{minimize} \quad \|\mathbf{Af}\| \tag{2.5}$$

$$\|\mathbf{f}\|\ =\ 1 \qquad \text{(important for avoid the trivial solution } \mathbf{f} = \mathbf{0}) \tag{2.6}$$

The *DLT* uses the SVD for computing the solution, in fact letting:

$$\mathbf{A} = \mathbf{UDV}^{\mathsf{T}} \tag{2.7}$$

the desired **f** corresponds to the last column of **V** (due to the fact that **D** is diagonal with elements in descending order).
At this point we can reshape **f** column vector into a 3X3 matrix in order to obtain **F1** as shown in relation **2.4**

3. we need to force

$$\text{rank}(\mathbf{F1}) = 2 \tag{2.8}$$

due to the fact that we are not in the ideal case so the rank of **F1** could becomes equal to 3. For ensure that the 'new' **F1**, the one with the constraint of det(**F1**)=0, is as much as possible closes to the one estimated, we need to do following steps:

- decompose **F1** via SVD
- set D(3,3) = 0
- recompute **F1** using this relation:

$$\mathbf{F1} = \mathbf{UDV}^{\mathsf{T}} \tag{2.9}$$

The estimated fundamental matrix **F1** obtained with this type of algorithm is not invariant to point transformations, this is the reason for which there's also the normalized version.

2.3 8-points Algorithm normalized

This algorithm is based on the same idea of the *8-points algorithm* but uses as inputs points that are previously normalized. As for the previous algorithm, we will proceed to analyze it step-by-step:

1. normalize the inputs points **P1** and **P2** in such a way that:

   - the image coordinate center corresponds to the points centroid ⇒ **TRANSLATION of points by factor** (c_x, c_y)
   - the average distance from points to the origin is $\sqrt{2}$ ⇒ **SCALING of points by factor s**

These two transformations can be summarized as follow:

$$\mathbf{T_1} = \begin{bmatrix} s_1 & 0 & -s_1 c_{1x} \\ 0 & s_1 & -s_1 c_{1y} \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{T_2} = \begin{bmatrix} s_2 & 0 & -s_2 c_{2x} \\ 0 & s_2 & -s_2 c_{2y} \\ 0 & 0 & 1 \end{bmatrix} \tag{2.10}$$

2. as could be expected, steps **2**, **3**, **4** are equal to the ones that have already been explained in section **2.2** (respectively steps **1**, **2**, **3**) with the only difference that the input **P1** and **P2** are normalized values and so they can be represented as:

$$P1 = T_1 * P1 \quad , \quad P2 = T_2 * P2 \tag{2.11}$$

(For distinguish the fundamental matrix estimated via the normalized algorithm we changed the name into **F2**)

5. De-normalize **F2** via this relation:

$$F2 = T_2^{\mathsf{T}} * F2 * T_1 \tag{2.12}$$

As just mentioned, this fundamental matrix **F2** is invariant to point transformation so will lead to better results (as will be explained in the following section).

2.4 Evaluation of the results

1. check that the epipolar constraint (relation **1.5**) holds for all points with **F1** and **F2**

   The two tables on the side contain the results of the epipolar constraints checked for *Mire* image (table 1) and *Rubik* image (table 2).
   In each table the first column represents the product obtained using **F1** as fundamental matrix instead the second the outcomes get considering the normalized one so **F2**.

	1	2
1	0.0072	0.0094
2	-0.0018	-4.0070e-04
3	0.0073	-5.8579e-06
4	0.0100	0.0050
5	0.0227	0.0031
6	0.0253	0.0097
7	0.0252	-0.0048
8	0.0547	0.0072
9	0.0246	0.0030
10	0.0401	0.0010
11	0.0427	0.0092
12	7.8664e-04	0.0043
13	0.0192	0.0021
14	0.0115	0.0099
15	0.0010	0.0069

	1	2
1	0.0905	0.0030
2	-0.7632	0.0012
3	-0.7653	0.0023
4	-0.1886	-0.0020
5	-0.0934	-0.0014
6	-0.2757	0.0011
7	-0.2190	-0.0012
8	-0.6220	0.0013
9	-0.4213	0.0035
10	-0.1686	-9.8534e-04
11	-0.8453	0.0012
12	-0.0050	0.0015
13	-0.1218	0.0014
14	0.0115	0.0099
15	0.0010	0.0069

   From that tables we can observe that all values are almost near to zero and, for the majority of the values, the second column are lower than those in the first.

2. visualization of the stereo pairs with the epipolar lines of the corresponding points

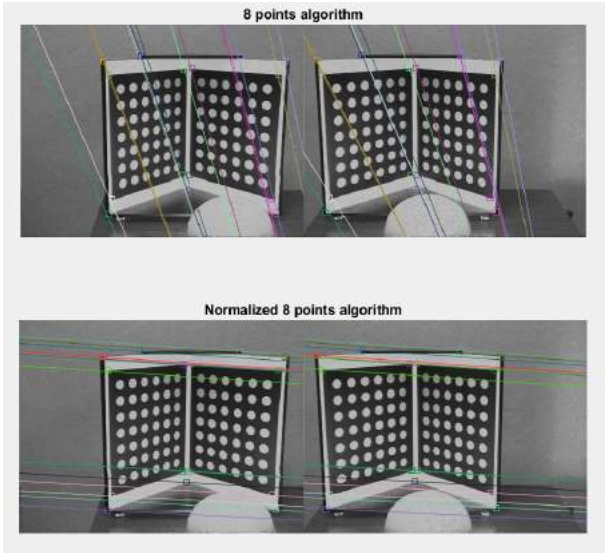

Figure 2.1: Stereo pairs *Mire1.pgm* and *Mire2.pgm* with epipolar lines

From **Figure 2.1** it is possible to notice that there are big differences between the results obtained with the two algorithms. We can conclude that for this image stereo pairs it is necessary to uses the normalized version because the epipolar lines obtained with the first method are unacceptable[4].

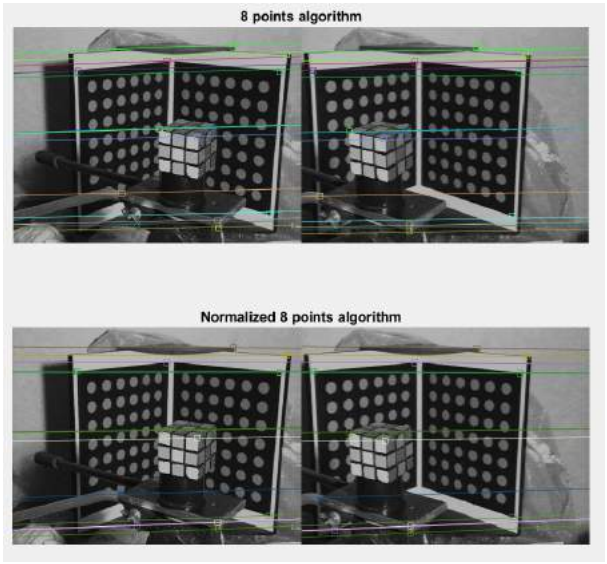

Figure 2.2: Stereo pairs *Rubik1.pgm* and *Rubik2.pgm* with epipolar lines

From **Figure 2.2**, instead, we can see that the results obtained by the two methods are almost the same, so apply the second method is not strictly necessary in order to gets good results.

3. check the properties of the right and the left epipole (respectively **e'** and **e**)

   Recalling from section **1.3**, we have that the left epipole **e** is the right annihilator for **F** (relation **1.10**) and similarly the right epipole **e'** is the left annihilator for **F** (relation **1.9**).
   To compute **e** and **e'** we need to do these steps:

   - perform the SVD on **F1** and **F2**
   - extract the values of the last column of **V** (for obtaining the coordinates of **e**) and **U** (for obtaining the coordinates of **e'**)

[4]Other considerations on this aspect will be made during the analysis of the epipole results.

Considering the stereo pairs **Mire1.points** and **Mire2.points**:

- 8-points algorithm
  - check that **e** is the right annihilator of **F1**

$$\mathbf{F1}\ \mathbf{e} = \mathbf{0} \tag{2.13}$$

    our results: (6.9456e-20, -3.1763e-20,-1.0842e-19)

  - check that **e'** is the left annihilator of **F1**

$$\mathbf{e'}^{\mathsf{T}}\ \mathbf{F1} = \mathbf{0} \tag{2.14}$$

    our results: (-1.9385e-20,4.3047e-20,-5.3216e-17)

  - positions of **e** and **e'**

    e = (-0.4090,-0.9125,-0.0003)
    e' = (0.4486,0.8936,0.0005)

- 8-points algorithm normalized
  - check that **e** is the right annihilator of **F1**

$$\mathbf{F1}\ \mathbf{e} = \mathbf{0} \tag{2.15}$$

    our results: (-1.2485e-20,1.8282e-19,6.3554e-18)

  - check that **e'** is the left annihilator of **F1**

$$\mathbf{e'}^{\mathsf{T}}\ \mathbf{F} = \mathbf{0} \tag{2.16}$$

    our results: (-1.7506e-21,2.5597e-20,-1.6875e-18)

  - positions of **e** and **e'**

    e = (-0.9977,-0.0672,-2.5569e-05)
    e' = (0.9974,0.0711,4.2770e-05)

Considering the stereo pairs **Rubik1.points** and **Rubik2.points**:

- 8-points algorithm
  - check that **e** is the right annihilator of **F2**

$$\mathbf{F2}\ \mathbf{e} = \mathbf{0} \tag{2.17}$$

    our results: (-2.8402e-19,-8.2016e-18,1.95075e-17)

  - check that **e'** is the left annihilator of **F2**

$$\mathbf{e'}^{\mathsf{T}}\ \mathbf{F2} = \mathbf{0} \tag{2.18}$$

    our results: (-5.6080e-22,-2.4534e-20,1.1037e-17)

  - positions of **e** and **e'**

    e = (0.9993,-0.0346,-2.2448e-05)
    e' = (0.9999,-0.0112,1.0950e-05)

- 8-points algorithm normalized
  - check that **e** is the right annihilator of **F1**

$$\mathbf{F1}\ \mathbf{e} = \mathbf{0} \tag{2.19}$$

    our results: (-1.0458e-20,1.1140e-18,-8.4978e-19)

  - check that **e'** is the left annihilator of **F1**

$$\mathbf{e'}^{\mathsf{T}}\ \mathbf{F} = \mathbf{0} \tag{2.20}$$

    our results: (1.2641e-23,-6.1198e-22,-2.2059e-20)

  - positions of **e** and **e'**

    e = (-0.9999,-0.0094,-8.7418e-05)
    e' = (-0.9998,-0.0168,-0.0001)

Summarizing the coordinates of all epipoles:

input	**8-points algorithm**	**8-points algorithm normalized**
Mire	**e** = (-0.4090,-0.9125,-0.0003) **e'** = (0.4486,0.8936,0.0005)	**e** = (-0.9977,-0.0672,-2.5569e-05) **e'** = (0.9974,0.0711,4.2770e-05)
Rubik	**e** = (0.9993,-0.0346,-2.2448e-05) **e'** = (0.9999,-0.0112,1.0950e-05)	**e** = (-0.9999,-0.0094,-8.7418e-05) **e'** = (-0.9998,-0.0168,-0.0001)

From the above table it is possible to notice that:

- the z-coordinate (third component) of each epipole is closes to zero and this means that, for each stereo pairs, the image planes are almost parallel each other
- considering the 8-points algorithm, the epipoles coordinates found in *Mire* image pairs have x and y coordinates more different each other than in the ones obtained in the *Rubik* image pairs. This leads to the fact that epipolar lines are not horizontal at all.
- applying the 8-points algorithm normalized to the *Mire* image pairs allow to have more similar values of epipoles coordinates and so more horizontal epipolar lines
- the coordinates of the epipoles obtained using the first method are slightly different from the second one. This confirm the the fact that for these images we can get good results even using the non normalized algorithm