Zhuobiao Qiao
UFID:15695876

Homework 4

CAP 5416
November 13, 2019

1. If points x in the left camera and x' in the right camera correspond to each other in a stereo camera system, what is the equation of the epipolar line in the right camera for any point x in the left camera in terms of the Fundamental matrix F? Also, show that x and x' are corresponding points if and only if the condition $x'Fx = 0$ is satisfied. How many degrees of freedom does the F matrix (which is a (3,3) matrix) have?

Answer:

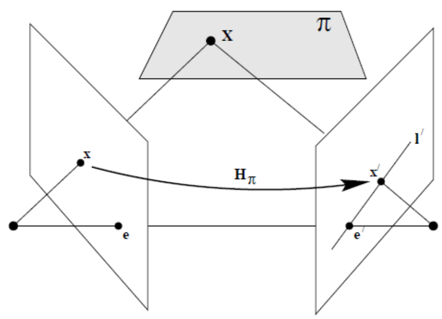

Figure 1: A point x in one image is transferred via the plane π to a matching point x' in the second image. The epipolar line through x' is obtained by joining x' to the epipole e'.

First, transfer via the plane π. Consider a plane π in space not passing through either of the two camera centres. The ray through the first camera centre corresponding to the point x meets the plane π in a point X. This point X is then projected to a point x' in the second image. Since X lies on the ray corresponding to x, the projected point x' must lie on the epipolar line l' corresponding to the image of this ray, The points x and x' are both images of the 3D point X lying on a plane. The set of all such points x_i in the first image and the corresponding points x'_i in the second image are projectively equivalent, since they are each projectively equivalent to the planar point set X_i. Thus there is a 2D homography H_π mapping each x_i to x'_i.

Then, Given the point x' the epipolar line l' passing through x' and the epipole e' can be written as $l' = e' \times x' = [e']_\times x'$ Where $e' = \{e'_1, e'_2, e'_3\}^T$

$$[e']_\times = \begin{bmatrix} 0 & -e'_3 & e'_2 \\ e'_3 & 0 & -e'_1 \\ -e'_2 & e'_1 & 0 \end{bmatrix} \tag{1}$$

Since x' may be written as $x' = H_\pi x$, we have

$$l' = [e']_\times H_\pi x = Fx \tag{2}$$

where we define $F = [e']_\times H_\pi$ the fundamental matrix. if points x and x' correspond, then x' lies on the epipolar line $l' = Fx$ corresponding to the point x. In other words, $0 = x'^T l' = x'^T Fx$. Conversely, if image points satisfy the relation $x'^T Fx = 0$ then the rays defined by these points are coplanar. This is a necessary condition for points to correspond.

F has seven degrees of freedom: a 3×3 homogeneous matrix has eight independent ratios (there are nine elements, and the common scaling is not significant); however, F also satisfies the constraint $det F = 0$ which removes one degree of freedom.

2. Write down the linear system of equations (put it in matrix vector form) required to solve for the Fundamental matrix given $(x_i, y_i, 1)$ and $(x'_i, y'_i, 1), i = 1. \cdots, n; n \geq 8$, the pairs of correspondences. Why is the solution to the ensuing linear least squares problem taken to be the singular vector corresponding to the smallest singular value?

Answer:

$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0 \tag{3}$$

Denote by f the 9-vector made up of the entries of F in row-major order. Then the equation.3 can be expressed as a vector inner product

$$(x'x, x'y, x', y'x, y'y, y', x, y, 1)f = 0. \tag{4}$$

From a set of n point matches, we obtain a set of linear equations of the form

$$Af = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} f = 0 \tag{5}$$

This is a homogeneous set of equations, and f can only be determined up to scale. For a solution to exist, matrix A must have rank at most 8, and if the rank is exactly 8, then the solution is unique (up to scale), and can be found by linear methods – the solution is the generator of the right null-space of A.

If the data is not exact, because of noise in the point coordinates, then the rank of A may be greater than 8 (in fact equal to 9, since A has 9 columns). In this case, one finds a least-squares solution. Apart from the specific form of the equations the problem is essentially the same as the estimation problem: Given $n \geq 4$ 2D to 2D point correspondences $x_i \leftrightarrow x'_i$, determine the 2D homography matrix H such that $x'_i = Hx_i$. The least-squares solution for f is the singular vector corresponding to the smallest singular value of A, that is, the last column of V in the SVD $A = UDV^T$. The solution vector f found in this way minimizes $\|Af\|$ subject to the condition $\|f\| = 1$.

3. In optical flow, if you assume the flow to be a constant over a patch, with the constraint that pixels far from the center of the patch are weighted less than those closer to it, derive a closed form expression for this weighted flow. Explicitly set up the linear system for the weighted flow i.e., show the entries of the matrix in the linear system and the right hand side vector in the linear system. Assume the patch size to be (n, n) and the weight matrix to be W.

Answer: The Lucas–Kanade method assumes that the displacement of the image contents between two nearby instants (frames) is small and approximately constant within a neighborhood of the point p under consideration. Thus the optical flow equation can be assumed to hold for all pixels within a window centered at p. Namely, the local image flow (velocity) vector (V_x, V_y) must satisfy

$$\begin{aligned} I_x(q_1)V_x + I_y(q_1)V_y &= -I_t(q_1) \\ I_x(q_2)V_x + I_y(q_2)V_y &= -I_t(q_2) \\ \vdots I_x(q_n)V_x + I_y(q_n)V_y &= -I_t(q_n) \end{aligned} \tag{6}$$

where $q_1, q_2, \dots, q_n$ are the pixels inside the window, and $I_x(q_i), I_y(q_i), I_t(q_i)$ are the partial derivatives of the image I with respect to position x, y and time t, evaluated at the point q_i and at the current time. These equations can be written in matrix form $Av = b$, where

$$A = \begin{bmatrix} I_x(q_1) & I_y(q_1) \\ I_x(q_2) & I_y(q_2) \\ \vdots & \vdots \\ I_x(q_n) & I_y(q_n) \end{bmatrix} \quad (7) \qquad v = \begin{bmatrix} V_x \\ V_y \end{bmatrix} \quad (8) \qquad b = \begin{bmatrix} -I_t(q_1) \\ -I_t(q_2) \\ \vdots \\ -I_t(q_n) \end{bmatrix} \quad (9)$$

This system has more equations than unknowns and thus it is usually over-determined. The Lucas–Kanade method obtains a compromise solution by the least squares principle. Namely, it solves the 2×2 system

$$A^TAv = A^Tb \quad or \quad v = (A^TA)^{-1}A^Tb \tag{10}$$

where A^T is the transpose of matrix A. That is, it computes

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum_i I_x(q_i)^2 & \sum_i I_x(q_i)I_y(q_i) \\ \sum_i I_y(q_i)I_x(q_i) & \sum_i I_y(q_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i I_x(q_i)I_t(q_i) \\ -\sum_i I_y(q_i)I_t(q_i) \end{bmatrix} \tag{11}$$

where the central matrix in the equation is an Inverse matrix. The sums are running from $i = 1$ to n.

The matrix A^TA is often called the structure tensor of the image at the point p.

The plain least squares solution above gives the same importance to all n pixels q_i in the window. In practice it is usually better to give more weight to the pixels that are closer to the central pixel p. For that, one uses the weighted version of the least squares equation,

$$A^TWAv = A^TWb \quad or \quad v = (A^TWA)^{-1}A^TWb \tag{12}$$

where W is an $n \times n$ diagonal matrix containing the weights $W_{ii} = w_i$ to be assigned to the equation of pixel q_i. That is, it computes

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum_i w_iI_x(q_i)^2 & \sum_i w_iI_x(q_i)I_y(q_i) \\ \sum_i w_iI_y(q_i)I_x(q_i) & \sum_i w_iI_y(q_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i w_iI_x(q_i)I_t(q_i) \\ -\sum_i w_iI_y(q_i)I_t(q_i) \end{bmatrix} \tag{13}$$

4. (bonus question) Brightness Constancy is assumed in many optical flow algorithms. However, for applications with substantial illumination variation, this assumption is often violated and therefore, it often gives incorrect flow estimates. Given two images I1 and I2. One possible idea to account for the illumination effect is through the following simple linear model:

$$I_2(x', y') = a(x, y)I_1(x, y) + b.$$

That is, if $(x', y') \leftrightarrow (x; y)$ are the corresponding pixels (points) on I_2 and I_1 respectively, their intensities are related as above with $a(x, y)$ the spatially-varying contrast function and a global constant b.

With this model, we have to estimate the flow field $\boldsymbol{u} = (u(x, y), v(x, y))$, the contrast $a(x, y)$ and the constant b. We can extend the usual variational approach by incorporating the above model into the cost function and using L2-regularization on the derivatives of u, v and a respectively.

Write down the cost function E(u, a, b) for computing the flow field u using the idea described above.

Derive the Euler-Lagrange Equations to minimize E(u, a, b).

Answer:

First, I think if b stands for a global constant, then when use differential, it will equal to 0 and hence is useless. I will generalize it and substitute it with b(x, y), if I misunderstand it, just cancel the term involve b(x, y) and all the derivation remains right.

(1)Rewriting the brightness change constraint equation, we obtain

$$I(x + \delta x, y + \delta y, t) = a(x, y, t)I(x, y, t) + b(x, y, t) \tag{14}$$

where a is the multiplier and b is the offset functions in the linear transformation. For small δt, we expect a to be close to 1, and b to be close to 0. Since we are dealing with incremental changes, we can let $a = 1 + \delta A$ and $b = \delta B$. In fact, A and B are the quantities of interest to us. Noting that $\delta A \to 0$ and $\delta B \to 0$ as $\delta t \to 0$, we can define time derivatives, A_t and B_t,

$$\begin{aligned} A_t &= \lim_{\delta t \to 0} \frac{\delta A}{\delta t} \\ B_t &= \lim_{\delta t \to 0} \frac{\delta B}{\delta t} \end{aligned} \tag{15}$$

Using Tayler series to expand the left hand side as follows:

$$\begin{aligned} I(x + \delta x, y + \delta y, t) &= I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + O(\epsilon) \\ &= I(x, y, t) + I_x\delta x + I_y\delta y + I_t\delta t + O(\epsilon) \end{aligned} \tag{16}$$

Substituting this into the constraint equation and simplifying, we have

$$I_x\delta x + I_y\delta y + I_t\delta t - I\delta A - \delta B + O(\epsilon) = 0 \tag{17}$$

Finally, dividing through by δt and taking the limit as $\delta t \to 0$, we arrive at

$$I_t + I_x \frac{dx}{dt} + I_y \frac{dy}{dt} - IA_t - B_t = 0 \tag{18}$$

written with elements in problem

$$I_t + I_x u + I_y v - Ia_t - b_t = 0 \tag{19}$$

where $u = \frac{dx}{dt}$ and $v = \frac{dy}{dt}$. If b in problem just stands for a constant. then b_t can be ignored.

the measure of departure from smoothness that is to be minimized is written

$$e_s = \iint \|\nabla \boldsymbol{r}_t\|_2^2 dxdy \tag{20}$$

where

$$\nabla \boldsymbol{r}_t = \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{r}} = \begin{pmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{pmatrix} \tag{21}$$

Here, $\|\cdot\|_2^2$ denote the Euclidean or Frobenius norm of a matrix, which is the sum of the square of all the elements of the matrix.

similarly, smoothness deviations can be defined for the transformation fields

$$\begin{aligned} e_a &= \iint \|\nabla a_t\|_2^2 dxdy \\ e_b &= \iint \|\nabla b_t\|_2^2 dxdy \end{aligned} \tag{22}$$

The image brightness constraint and the smoothness constraints can be combined by defining a single functional that weighs each contribution. Rather than enforcing the brightness change constraint exactly, we use a penalty term that measures the square of the error in the constraint equation over the whole image:

$$e_c = \iint (I_t + I_x u + I_y v - Ia_t - b_t)^2 dxdy \tag{23}$$

To ensure that the optical flow and the transformation fields(approximately) satisfy the optical flow constraint equation, we want e_c to be small.

All together, the problem can be formulated as that of minimizing the functional

$$E(\boldsymbol{u}, a, b) = e_c + \lambda_s e_s + \lambda_a e_a + \lambda_b e_b \tag{24}$$

where $\lambda_s, \lambda_a, \lambda_b$ weigh the total error contributed by each term. If b is independent with x, y, then just cancel the term with respect to b

(2)the Euler-Lagrange Equations is solved by the formula

$$\mathbf{\Psi}_f - \frac{\partial}{\partial x}\mathbf{\Psi}_{f_x} - \frac{\partial}{\partial y}\mathbf{\Psi}_{f_y} = 0 \tag{25}$$

where $\mathbf{\Psi}$ is the integrand in the cost functional and f is each of $u, v, a_t or b_t$, in turn. Applying the above formula, we obtain

$$\begin{aligned}
\nabla^2 u &= \frac{I_x}{\lambda_s}(I_t + I_x u + I_y v - I a_t - b_t) \\
\nabla^2 v &= \frac{I_y}{\lambda_s}(I_t + I_x u + I_y v - I a_t - b_t) \\
\nabla^2 a_t &= \frac{-I}{\lambda_s}(I_t + I_x u + I_y v - I a_t - b_t) \\
\nabla^2 b_t &= \frac{-1}{\lambda_s}(I_t + I_x u + I_y v - I a_t - b_t)
\end{aligned} \tag{26}$$

For a well-posed problem, we need to specify the appropriate boundary conditions. In the absence of fixed boundary conditions, we need to specify natural boundary conditions. For our problem, the natural boundary condition is $(f_x, f_y)^T \cdot \hat{\boldsymbol{n}} = 0$, where $\hat{\boldsymbol{n}}$ is a unit vector perpendicular to the boundary. Again, f can be any one of u, v, a_t or b_t, in turn.

The Answer of 1, 2 is referenced to [1]. The answer of 3, 4 is referenced to [2]. 4 is also referenced to [3], [4], [5]

References

[1] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision.* Cambridge university press, 2003.

[2] B. Horn, B. Klaus, and P. Horn, *Robot vision.* MIT press, 1986.

[3] Y.-H. Kim, A. M. Martínez, and A. C. Kak, "Robust motion estimation under varying illumination," *Image and Vision Computing*, vol. 23, no. 4, pp. 365–375, 2005.

[4] Y.-H. Kim and A. C. Kak, "Error analysis of robust optical flow estimation by least median of squares methods for the varying illumination model," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 9, pp. 1418–1435, 2006.

[5] M. A. Gennert and S. Negahdaripour, "Relaxing the brightness constancy assumption in computing optical flow," MASSACHUSETTS INST OF TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE LAB, Tech. Rep., 1987.