MATH 425 Fall 2024 Final Project: Calculation of a Fundamental Matrix

Miguel Antonio Logarta

Due: Sunday December 22, 2024

1 Introduction

In computer vision, using two cameras is a common way to understand the layout of a 3D scene. This approach solves the problem of missing information when using just one image. With a single image, a computer cannot accurately determine the 3D layout because it lacks details like depth and distance between objects. This is where epipolar geometry becomes important.

Epipolar geometry explains the relationship between two images taken from different positions by two cameras viewing the same 3D scene. The fundamental matrix is helpful in this context because it lets us reconstruct a 3D scene using two uncalibrated cameras. This can be done with just a few matching points between the two images, without needing additional information about the 3D scene.

Before we dive into the code, I'll first explain the basics of Epipolar Geometry.
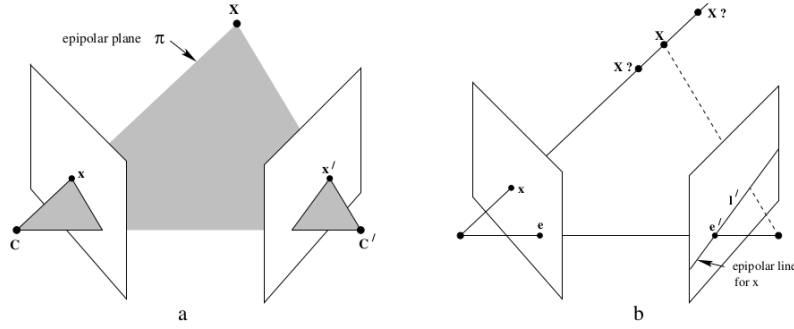
2　Epipolar Geometry

Fig. 9.1. **Point correspondence geometry.** *(a) The two cameras are indicated by their centres* **C** *and* **C′** *and image planes. The camera centres, 3-space point* **X**, *and its images* **x** *and* **x′** *lie in a common plane* **π**. *(b) An image point* **x** *back-projects to a ray in 3-space defined by the first camera centre,* **C**, *and* **x**. *This ray is imaged as a line* **l′** *in the second view. The 3-space point* **X** *which projects to* **x** *must lie on this ray, so the image of* **X** *in the second view must lie on* **l′**.

Imagine a 3D point in space called **X**, which is projected onto two images as points x and x'. Both x and x' lie on the same plane, called the epipolar plane. These images represent the views from two different cameras.

The points along the line connecting the 3D point **X** to x, as seen from camera **C**, create a line l' in the view of the other camera **C'**. This line l' is called an epipolar line. Similarly, the points connecting **X** to x', as seen from camera **C'**, form another epipolar line l in the view of camera **C**.

A key property of epipolar lines is that they intersect at specific points called epipoles, e and e'. These epipoles are the points where the line connecting the two camera centers (known as the baseline) intersects the image planes.

3　The Fundamental Matrix

The Fundamental matrix relates the corresponding points x and x' of the two cameras in which the epipolar lines are written as

$$l' = [e']_\times H_\pi x = Fx$$

$$l = [[e']_\times H_\pi x]^T x' = F^T x'$$

In short, it maps the points in one image to their corresponding epipolar lines in the other image. In addition, the explanation from the textbook also shows

that the second camera is just a transformation of the first camera in the 3D scene. It satisfies the equation

$$x'^T F x = 0$$

where F is a 3x3 matrix.

4   Calculating the Fundamental Matrix

To find the fundamental matrix, we use the normalized eight point algorithm. The normalized eight point algorithm calculates the fundamental matrix with 8 matching point correspondences from two matching images. The algorithm from the textbook is as follows:

282   *11 Computation of the Fundamental Matrix* F

Objective

Given $n \geq 8$ image point correspondences $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i\}$, determine the fundamental matrix F such that $\mathbf{x}'^{\top}_i \mathsf{F} \mathbf{x}_i = 0$.

Algorithm

  (i) **Normalization:** Transform the image coordinates according to $\hat{\mathbf{x}}_i = \mathsf{T}\mathbf{x}_i$ and $\hat{\mathbf{x}}'_i = \mathsf{T}'\mathbf{x}'_i$, where T and T' are normalizing transformations consisting of a translation and scaling.

 (ii) Find the fundamental matrix $\hat{\mathsf{F}}'$ corresponding to the matches $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$ by

     (a) **Linear solution:** Determine $\hat{\mathsf{F}}$ from the singular vector corresponding to the smallest singular value of $\hat{\mathsf{A}}$, where $\hat{\mathsf{A}}$ is composed from the matches $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$ as defined in (11.3).

     (b) **Constraint enforcement:** Replace $\hat{\mathsf{F}}$ by $\hat{\mathsf{F}}'$ such that $\det \hat{\mathsf{F}}' = 0$ using the SVD (see section 11.1.1).

(iii) **Denormalization:** Set $\mathsf{F} = \mathsf{T}'^{\top}\hat{\mathsf{F}}'\mathsf{T}$. Matrix F is the fundamental matrix corresponding to the original data $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$.

---

5   Translating Math Into Code

5.1   Getting Point Correspondences

For this project, I used two images of my toaster to calculate the fundamental matrix. I manually selected 8 points from both images and stored the point correspondences in **x** and **x'** where each coordinate $x_i = (x, y, 1)^T$ and $x'_i = (x', y', 1)^T$. The points are highlighted in red.

Toaster 1   **Toaster 2**
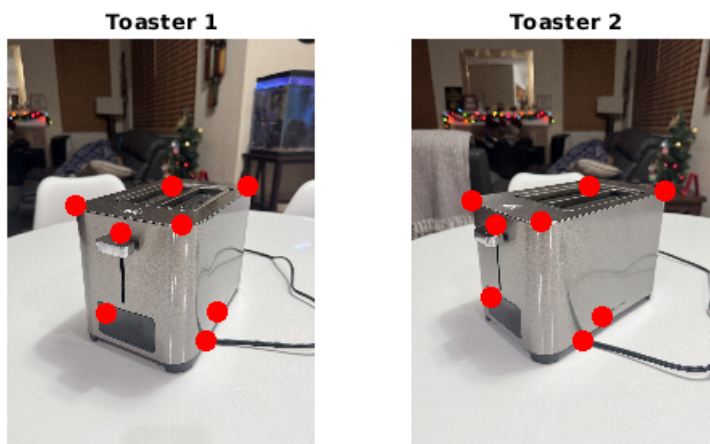
5.2   Normalizing the Points

Next, I needed to normalize the points. Normalization is an important step because modern cameras produce images with very large pixel values. Without normalization, these large numbers can make the calculation of the fundamental matrix less accurate.

The textbook recommended scaling each point using the RMS (root mean square) distance formula, where the origin of the points are centered around (0,0), with the average distance from the origin being $\sqrt{2}$. However, the explanation in the book lacked clear examples, so I had to research further. After some digging, I learned that each point should be scaled using the following formula:

$$s = \left(\frac{2N}{\sum_{i=1}^{N} \|x_i - \bar{x}\|^2} \right)^{1/2}$$

After some more digging, I used the scale to construct a transformation matrix T and applied it to every point in **x** and **x'**

$$T = \begin{pmatrix} s & 0 & -s * \bar{x} \\ 0 & s & -s * \bar{y} \\ 0 & 0 & 1 \end{pmatrix}$$

4

5.3   Main Algorithm

These matching points are in **x** and **x'** where each coordinate $x_i = (x, y, 1)^T$ and $x_i' = (x', y', 1)^T$. We then try to solve for the system of linear equations $A\mathbf{f} = \mathbf{0}$:

$$A\mathbf{f} = \begin{bmatrix} x_1'x_1 & x_1'y_1 & x_1' & y_1'x_1 & y_1'y_1 & y_1' & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n'x_n & x_n'y_n & x_n' & y_n'x_n & y_n'y_n & y_n' & x_n & y_n & 1 \end{bmatrix} \mathbf{f} = 0.$$

The linear solution **f** is found in the <u>last column of V^T</u> in the SVD of A where $SVD(A) = UDV^T$. We end up with a 9x1 vector which we then have to reshape into a 3x3 matrix which is our Fundamental Matrix.

Now we have to enforce a constraint on F. F must be of rank-2 and singular $(det(F) = 0)$. This ensures that the epipolar lines in both images meet at a common epipole.

To enforce this constraint, we get the SVD of F again where $SVD(F) = UDV^T$. We take the last diagonal of the 3x3 matrix D, and set it to zero to force F to be of rank 2. We recompute F' as F' $= F' = U * diag(r, s, 0) * V^T$

5.4   Denormalization

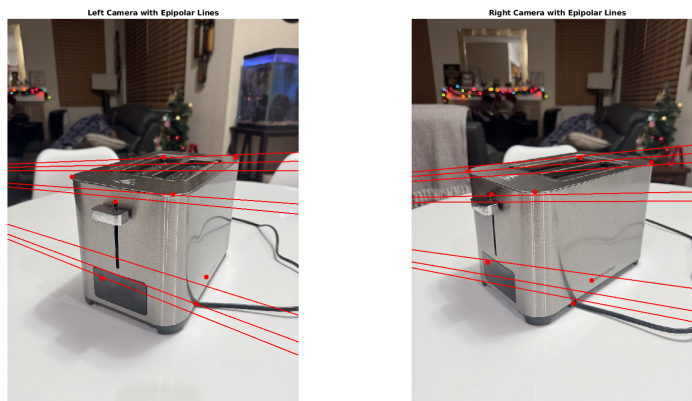We denormalize F by applying the transformation matrix T

$$F = T' * F' * T;$$

6   Results

Now that we have the fundamental matrix, we can plot the epipolar lines on both of our images. Since we denormalized F, we apply F to our denormalized coordinates instead of our normalized coordinates. The equations for our epipolar lines are:

$$l' = Fx$$
$$l = F^T x'$$

7   Remarks

Here are the results. Unfortunately, while some of my epipolar lines align with the corresponding points, many of them do not. I believe this is due to an inaccurate fundamental matrix. When checking x'Fx=0x'Fx=0, the value was around 0.002, which might not be precise enough. After reviewing my code to ensure the correspondence points were properly normalized and the transformation was applied correctly, I couldn't find the issue before the deadline.

For future work, I plan to learn how the RANSAC algorithm works so it can automatically select key points in both images. I would also like to include examples where the cameras are parallel (so the epipolar lines are parallel) and a case where the epipole is inside the image, as the previous example showed the epipole outside the image.

References

[1] Hartley, Richard, and Andrew Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2004.
Chapter 9, 10, and 11 are the main sources of information regarding the calculation of the Fundamental matrix. It contains a very detailed background of epipolar geometry complete with proofs and algorithms.

[2] Hata, Kenji, and Silvio Savarese. "CS231A Course Notes 3: Epipolar Geometry." Stanford.Edu, `web.stanford.edu/class/cs231a/course_notes/03-epipolar-geometry.pdf`. Accessed 23 Dec. 2024.
This is an external source that contains information on epipolar geometry. I used this source due to its simplified explanation of epipolar geometry and the eight-point algorithm.