CSE 252B: Computer Vision II

Lecturer: Serge Belongie
Scribe: Haowei Liu

LECTURE 16
Structure from Motion from Tracked Points

16.1. Introduction

In the last lecture we learned how to track point features through an image sequence. In this lecture we will see how to use these tracked features to infer the scene geometry and camera motion. It is a very famous algorithm in the computer vision community called the "Factorization Method" (by Tomasi & Kanade). This approach assumes an orthographic camera model.

16.2. Orthographic Camera Model

The camera model we have studied so far is the projective model. The orthographic camera model is an approximation of the projective camera that projects 3D points to the image plane simply by dropping the Z coordinate:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \rightarrow \begin{bmatrix} X \\ Y \end{bmatrix}$$

[1] Department of Computer Science and Engineering, University of California, San Diego.

May 26, 2004

Rule of thumb for orthographic camera model assumptions

Suppose Z_{avg} is the average distance of an object from the camera and d_{avg} is the average width of the object (measured along the optical axis of the camera). Then the orthographic camera model is reasonable to use when $Z_{avg} \geq 10d_{avg}$.

Figure 1 demonstrates the conditions under which the orthographic camera model becomes applicable. On the right we see a scene containing a rectangular sign viewed by a projective camera as the photographer moves farther away from the sign. On the left, the portion of each image containing the sign is cropped and resized so that the sign occupies approximately the same area. It is evident as the camera moves farther away that the perspective effects diminish. In particular, the vanishing point for the parallel lines on the top and bottom of the sign moves increasingly close to infinity.

Alfred Hitchcock pioneered the use of this effect (the so-called "Hitchcock zoom") in the movie *Vertigo.*

16.3. Structure From Motion Theorems

Now we consider two structure from motion theorems that are pertinent to the factorization method.

Ullman's SFM theorem (1979)

Given three orthographic views of four rigid points in general position, the structure and camera motion consistent with the points is uniquely determined.

This is intuitive if you think of the way engineering drawings often depict 3D objects, i.e., three (carefully selected) orthographic views.

Koenderink & Van Doorn's Affine SFM theorem (1991)

Two views of four points in general position give the affine structure. We can think of one of the points as the origin and three other points as defining the coordinate axes of an affine coordinate system.

Koenderink & Van Doorn's paper introduced the idea of *stratification* (from affine to Euclidean). To go from affine to Euclidean, we need one more view.

16.4. Factorization Method

Tomasi & Kanade's method first estimates the affine structure and then upgrades it to Euclidean. It operates on a *batch* of at least 3 frames with known correspondences (i.e., with tracked features).

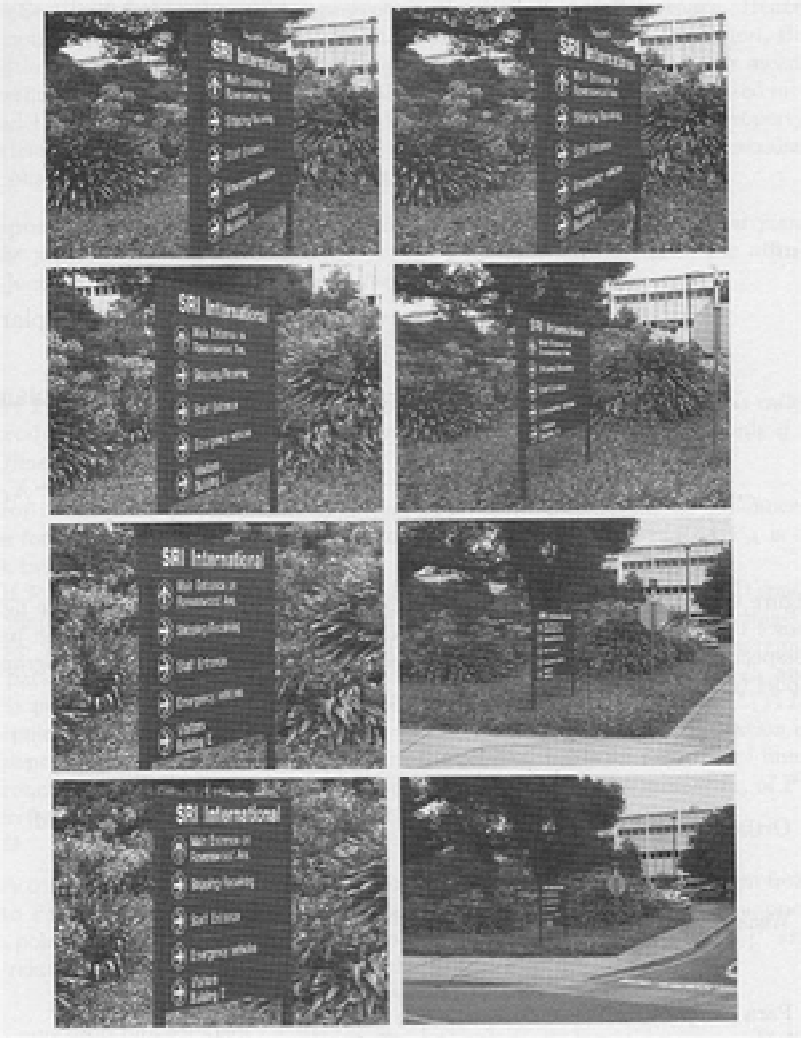

Figure 1. The orthographic camera model becomes valid with increasing distance from an object. [Faugeras, Luong, and Papadopoulo]

16.4.1. Initial Step

Suppose that we have tracked n points of a rigid scene over $N \geq 3$ frames, as shown in Figure 2. The scene points $\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_n$ are projected to the image points $\boldsymbol{p}_{ij} = (x_{ij}, y_{ij})^\top$, which represents the image of jth point in the ith frame. Let $\boldsymbol{T}_i$ be the vector from the world frame origin to the origin of the ith camera frame, and use $\boldsymbol{i}_i, \boldsymbol{j}_i, \boldsymbol{k}_i$ to denote the unit axis vectors of the ith camera frame.

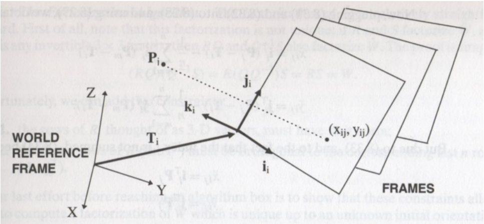

Figure 2. World reference and camera frames used in Factorization method derivation. [Trucco & Verri]

First put all the x_{ij}'s and y_{ij}'s into two $N \times n$ matrices X and Y and form the $2N \times n$ **measurement matrix** W.

$$W = \begin{bmatrix} X \\ Y \end{bmatrix}$$

Next, center it to get

$$\tilde{W} = \begin{bmatrix} \tilde{X} \\ \tilde{Y} \end{bmatrix}$$

where

$$\tilde{x}_{ij} = x_{ij} - \bar{x}_i, \qquad \tilde{y}_{ij} = y_{ij} - \bar{y}_i$$

$\bar{\boldsymbol{p}}_i = (\bar{x}_i, \bar{y}_i)^\top$ is the centroid of the points in the ith frame.

16.4.2. Rank Theorem

Notice that we don't need to represent $\boldsymbol{k}_i$ explicitly since we can get it by computing $\boldsymbol{k}_i = \boldsymbol{i}_i \times \boldsymbol{j}_i$. The coordinates of the points in the ith image plane are given by

$$\begin{aligned} x_{ij} &= \boldsymbol{i}_i^\top(\boldsymbol{P}_j - \boldsymbol{T}_i) \\ y_{ij} &= \boldsymbol{j}_i^\top(\boldsymbol{P}_j - \boldsymbol{T}_i) \end{aligned}$$

Performing the centering operation gives us:

$$\begin{aligned} \tilde{x}_{ij} &= \boldsymbol{i}_i^\top(\boldsymbol{P}_j - \boldsymbol{T}_i) - \frac{1}{n}\sum_{m=1}^{n} \boldsymbol{i}_i^\top(\boldsymbol{P}_m - \boldsymbol{T}_i) \\ \tilde{y}_{ij} &= \boldsymbol{j}_i^\top(\boldsymbol{P}_j - \boldsymbol{T}_i) - \frac{1}{n}\sum_{m=1}^{n} \boldsymbol{j}_i^\top(\boldsymbol{P}_m - \boldsymbol{T}_i) \end{aligned}$$

If we assume an object centered reference frame, then we have

$$\sum_{i=1}^{n} \boldsymbol{P}_i = \mathbf{0}$$

that is, the centroid of the points lies at the world origin. From this, it follows that

$$\tilde{x}_{ij} = \boldsymbol{i}_i^\top \boldsymbol{P}_j \qquad \text{and} \qquad \tilde{y}_{ij} = \boldsymbol{j}_i^\top \boldsymbol{P}_j$$

Now define the $2N \times 3$ matrix R to be

$$R^\top = [\boldsymbol{i}_1, \boldsymbol{i}_2, \cdots, \boldsymbol{i}_n, \boldsymbol{j}_1, \boldsymbol{j}_2, \cdots, \boldsymbol{j}_n]$$

and S to be

$$S = [\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_n]$$

Then $\tilde{W} = RS$.

The **Rank Theorem** says that $\tilde{W}$ has rank 3 (in absence of noise). This is evident by construction, since $\tilde{W}$ is formed by a product of two matrices with 3 rows or columns.

How do we get the poses of camera? If we can estimate $[\boldsymbol{i}_i, \boldsymbol{j}_i, \boldsymbol{i}_i \times \boldsymbol{j}_i]$, we can recover the rotational component of the camera pose. Note that we can not recover translation along the Z axis using the orthographic camera model. Moreover, we eliminated the other components of translation by centering the data. Thus we are only recovering the rotation of the camera.

16.4.3. Metric Constraints

The factorization of the matrix $\tilde{W}$ is not unique: if $\tilde{W}$ is factorized by R and S, the RQ and $Q^{-1}S$ also factorize $\tilde{W}$ for $Q \in GL(3)$, that is, $(RQ)(Q^{-1}S) = RS = \tilde{W}$

Fortunately, R is not comprised of arbitrary 3×1 vectors stacked side by side; they need to be orthonormal. Therefore, we have two constraints on R:

(a) The rows of R must have unit norm
(b) The $\boldsymbol{i}_i$'s must be perpendicular to the $\boldsymbol{j}_i$'s

Using these constraints on R, we can get a factorization of $\tilde{W}$ that is unique up to an unknown initial orientation of the world reference frame.

16.4.4. Algorithm based on SVD

(a) Compute $\tilde{W} = UDV^\top$, with D's diagonal sorted in descending order.
(b) When noise is present, define the **adjusted matrices**:

$$\begin{aligned} D' &= D(1:3, 1:3) \\ U' &= U(:, 1:3) \\ V' &= V(:, 1:3) \end{aligned}$$

and construct

$$\hat{R} = U'D'^{1/2}, \qquad \hat{S} = D'^{1/2}V'^\top$$

(c) Enforce constraints on R: we need to find $Q \in GL(3)$ such that

$$\begin{aligned} \hat{\boldsymbol{\imath}}_i^\top Q Q^\top \hat{\boldsymbol{\imath}}_i &= 1 \\ \hat{\boldsymbol{\jmath}}_i^\top Q Q^\top \hat{\boldsymbol{\jmath}}_i &= 1 \\ \hat{\boldsymbol{\imath}}_i^\top Q Q^\top \hat{\boldsymbol{\jmath}}_i &= 0 \end{aligned}$$

We want to find a Q that will solve this quadratic system of equations and then form $R = \hat{R}Q$ and $S = Q^{-1}\hat{S}$. To solve for Q, we can

- use Newton's method to find Q directly, or
- define $C = QQ^\top$ and solve the linear system of equations for $C = QQ^\top$ and use Cholesky to get Q. (Note that the recovered C may not be positive semidefinite; see Appendix.)

16.4.5. Conclusion

The algorithm works under two assumptions: orthographic camera model and tracked feature points.

Figure 3 demonstrates the recovered structure of a building using the algorithm.

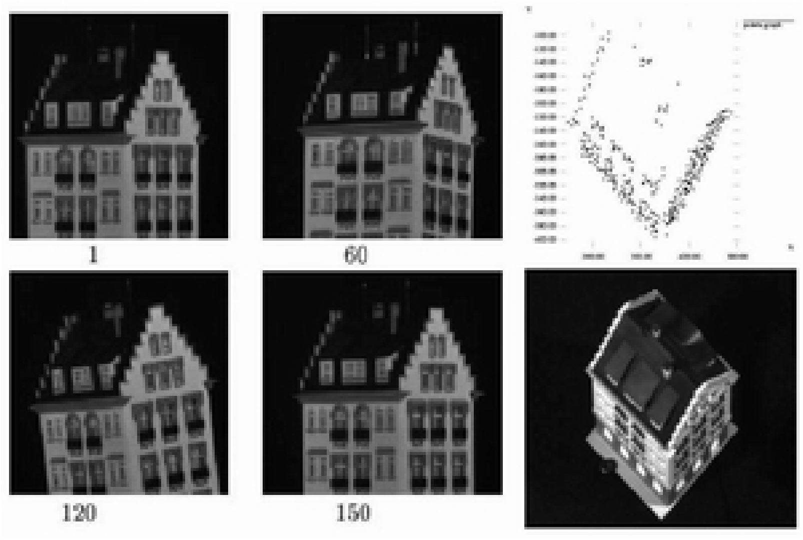

Figure 3. Recovered structure of a building [Tomasi and Kanade]

The extension to projective case is also possible, but not closed form [Sturm & Triggs 1996].

Appendix: enforcing positive semidefiniteness

When attempting to estimate C using the linear method, we might obtain a matrix that is not positive semidefinite. Sameer found a result that addresses this problem in a paper[1] by Nick Higham.

Let $\widetilde{C}$ represent our estimate that is supposed to be positive semidefinite. First, compute the symmetric component of $\widetilde{C}$ via the expression

$$\widetilde{C}_{sym} = \frac{\widetilde{C} + \widetilde{C}^\top}{2}$$

Now diagonalize $\widetilde{C}_{sym}$:

$$\widetilde{C}_{sym} = U \Lambda U^\top$$

and form the matrix Λ_+ by setting any negative eigenvalues to zero.

The positive semidefinite matrix that is closest to $\widetilde{C}$ in Frobenius norm is then given by

$$C_{psd} = U \Lambda_+ U^\top$$

The proof appears in Higham's paper.

If, as in the case of the metric upgrade step of Tomasi-Kanade, we just need the positive definite square root, we can form it simply as $U\Lambda_+^{1/2}$.

[1] N. Higham, "Computing a Nearest Symmetric Positive Semidefinite Matrix," Linear Algebra and Appl., 103:103-118, 1988.