Structure from Motion

Paul Heckbert, Nov. 1999

15-869, Image-Based Modeling and Rendering

Approach

Problem:

Reconstruct scene geometry and camera motion from two or more images

Typically, assume

- static scene (model camera motion only)
- diffuse, opaque surfaces (simplifies feature tracking)
- orthographic projection, for starters, then generalize to projective

Steps:

- shoot *f* frames of video (or sequence of stills)
- track *n* feature points from frame to frame
- build large matrix of image feature coordinates
- factor this matrix to extract motion and shape
- build 3-D model by connecting the features with triangles

http://www.ius.cs.cmu.edu/IUS/mbvc0/www/modeling.html

Image Features

Good features are spots, corners, and other points where the image, regarded as a terrain, has high curvature.

Good features are image windows that can be tracked well [Tomasi-Kanade 92]

Pick good features using the gradient covariance matrix [Lucas-Kanade 81]:

$$\sum_{\text{window}} \nabla I \cdot \nabla I^{\mathrm{T}} = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_y I_x & \sum I_y^2 \end{bmatrix}$$

How many large eigenvalues does this matrix have?

- 2 - good feature (spot)
- 1 - poor (edge - aperture problem)
- 0 - poor (flat)

Orthographic Projection

$$\underset{2\times1}{\mathbf{u}} = \underset{2\times3}{\boldsymbol{\Pi}}\ \underset{3\times1}{\mathbf{X}} + \underset{2\times1}{\mathbf{t}}$$

image point — projection matrix — scene point — image offset

Trick

Choose scene origin to be centroid of 3D points

Choose image origins to be centroid of 2D points

Allows us to drop the camera translation:

$$\underset{2\times1}{\mathbf{u}} = \underset{2\times3}{\boldsymbol{\Pi}}\ \underset{3\times1}{\mathbf{X}}$$

Tomasi-Kanade Orthographic Method

projection of *n* features in one image:

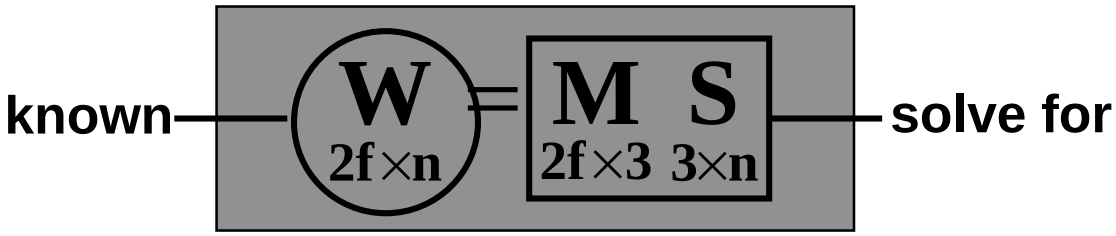
$$\underset{2\times n}{\begin{bmatrix} u_1 & u_2 & \cdots & u_n \end{bmatrix}} = \underset{2\times n}{\prod} \underset{3\times n}{\begin{bmatrix} X_1 & X_2 & \cdots & X_n \end{bmatrix}}$$

projection of *n* features in *f* images

$$\underset{2f\times n}{\begin{bmatrix} u_1^1 & u_2^1 & \cdots & u_n^1 \\ u_1^2 & u_2^2 & \cdots & u_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ u_1^f & u_2^f & \cdots & u_n^f \end{bmatrix}} = \underset{2f\times 3}{\begin{bmatrix} \Pi^1 \\ \Pi^2 \\ \vdots \\ \Pi^f \end{bmatrix}} \underset{3\times n}{\begin{bmatrix} X_1 & X_2 & \cdots & X_n \end{bmatrix}}$$

W measurement **M** motion **S** shape

Tomasi-Kanade Orthographic Method



SVD Factorization Technique

W is at most rank 3 (assuming no noise)

We can use *singular value decomposition* to factor W:

$$\underset{2f\times n}{W} = \underset{2f\times 3}{M'}\,\underset{3\times n}{S'}$$

S' differs from *S* by a linear transformation *A*:

$$W = M'S' = (MA^{-1})(AS)$$

Solve for *A* by enforcing constraints on *M*

Problems with Basic Orthographic Method, 1

- Measurement matrix **W** might have many voids, since occlusion and noise cause features to appear and disappear.

  Solution: take linear combinations of rows & columns to "hallucinate" entries.

- Features can be mis-tracked:

  Solution: maintain tree of multiple hypotheses

- SVD is slow: O($fn \cdot \min(f,n)$).

  Solution 1: solve bilinear equation by alternating between:

  - freeze S and solve $W=MS$ for M
  - freeze M and solve $W=MS$ for S

  This is faster than SVD.

  Solution 2: don't compute full SVD, but partial one

Problems with Basic Orthographic Method, 2

- Feature set too sparse, doesn't yield a good surface model.

  Solution: Don't track as a preprocess. Instead solve for correspondence as you solve for motion and shape.

- Orthographic projection is poor approximation for nearby objects.

  Solution: model perspective, but this is more complex.

  Projective factorization: [Poelman 95] refined orthographic solution using nonlinear optimization (Levenberg-Marquardt)

Triggs' Projective Method

[Triggs CVPR '96] generalized orthographic method by using homogeneous coordinates.

Image coordinates **u** have a third, unknown DOF, projective depth.

This yields a larger system of equations, but very similar approach.

Steps:

- track features
- find fundamental matrices **F** between successive frames
- use **F**'s to solve for projective depth
- solve projective factorization equation $\mathbf{W}_{3f \times n} = \mathbf{M}_{3f \times 3}\, \mathbf{S}_{3 \times n}$