Affine Structure-from-Motion: A lot of frames (1)

$$
\begin{pmatrix}
u_1^1 & u_2^1 & . & . & . & u_n^1 \\
v_1^1 & v_2^1 & & & & v_n^1 \\
u_1^2 & u_2^2 & & & & u_n^2 \\
v_1^2 & v_2^2 & & & & v_n^2 \\
. & . & & & & . \\
. & . & & & & . \\
. & . & & & & . \\
u_1^m & u_2^m & & & & u_n^m \\
v_1^m & v_2^m & . & . & . & v_n^m
\end{pmatrix}
=
\begin{pmatrix}
s_{1,1}^1 & s_{1,2}^1 & s_{1,3}^1 & t_x^1 \\
s_{2,1}^1 & s_{2,2}^1 & s_{2,3}^1 & t_y^1 \\
s_{1,1}^2 & s_{1,2}^2 & s_{1,3}^2 & t_x^2 \\
s_{2,1}^2 & s_{2,2}^2 & s_{2,3}^2 & t_y^2 \\
 & . & & \\
 & . & & \\
 & . & & \\
s_{1,1}^m & s_{1,2}^m & s_{1,3}^m & t_x^m \\
s_{2,1}^m & s_{2,2}^m & s_{2,3}^m & t_y^m
\end{pmatrix}
\begin{pmatrix}
x_1 & x_2 & . & . & . & x_n \\
y_1 & y_2 & & & & y_n \\
z_1 & z_2 & & & & z_n \\
1 & 1 & & & & 1
\end{pmatrix}
$$

I S P

First Step: Solve for Translation (1)

- This is trivial, because we can pick a simple origin.
  - World origin is arbitrary.
  - Example: We can assume first point is at origin.
    - Rotation then doesn't effect that point.
    - All its motion is translation.
  - Better to pick center of mass as origin.
    - Average of all points.
    - This also averages all noise.

Specifically, we can never tell where the world points were to begin with. Adding one to every x coordinate in P and then subtracting 1 in every tx is undetectable.

So, wlog we can assume that sum(P(k,:)) = 0 for k from 1 to 3, ie., sum(x1 … xn) = 0, sum(y1…yn) = 0, sum(z1 … zn) = 0.

Rotation doesn't move the origin, which is now the center of mass. Neither does scaled orthographic projection. So, this only moves from translation.

Explicitly, we assume sum(p) = (0,0,0)^T. Then: sum(s*R(p)) = s*R(sum(p)) = s*R(0,0,0)^T = (0,0,0)^T. (^T means transpose).

More explicitly, suppose sum(p) = (0,0,0,n)^T. Then, sum(R*P) = R*(sum(P)) = R*(0,0,0,n)^T = (0,0,0,n)^T. Sum(T*R*P) = T*(0,0,0,n)^T = (ntx,nty,ntz,n)^T. (Or just look at the 2x4 projection matrix). If we subtract tx or ty from every row, then the residual is (s11,s12,s13;s21,s22,s23)*P. I = s part of matrix + t part of matrix.

$$
s\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}
\begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \end{pmatrix}
\begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} & 0 \\ r_{2,1} & r_{2,2} & r_{2,3} & 0 \\ r_{3,1} & r_{3,2} & r_{3,3} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} P
$$

Even more explicitly. Consider the first row of the image matrix I. Average together all the entries in this row. This gives us:

sum((s{1,1},s{1,2},s{1,3})*(x_i,y_i,z_i) + tx)/n

= (s{1,1},s{1,2},s{1,3})*sum(x_i,y_i,z_i)/n + tx

= (s{1,1},s{1,2},s{1,3})*(0,0,0) + tx = tx.

So we've solved for tx. If we subtract tx from every element in the first row of I, we remove the effects of translation.

First Step: Solve for Translation (2)

$$
\text{WLOG}: \sum_{i=1}^{n} \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}
$$

$$
\overline{u}^i = \sum_{k=1}^{n} u_k^i \Big/ n
$$

$$
\overline{v}^i = \sum_{k=1}^{n} v_k^i \Big/ n
$$

$$
\tilde{I} = \begin{pmatrix}
u_1^1 - \overline{u}^1 & u_2^1 - \overline{u}^1 & . & . & . & u_n^1 - \overline{u}^1 \\
v_1^1 - \overline{v}^1 & v_2^1 - \overline{v}^1 & & & & v_n^1 - \overline{v}^1 \\
u_1^2 - \overline{u}^2 & u_2^2 - \overline{u}^2 & & & & u_n^2 - \overline{u}^2 \\
v_1^2 - \overline{v}^2 & v_2^2 - \overline{v}^2 & & & & v_n^2 - \overline{v}^2 \\
. & . & & & & . \\
. & . & & & & . \\
. & . & & & & . \\
u_1^m - \overline{u}^m & u_2^m - \overline{u}^m & & & & u_n^m - \overline{u}^m \\
v_1^m - \overline{v}^m & v_2^m - \overline{v}^m & . & . & . & v_n^m - \overline{v}^m
\end{pmatrix}
$$

First Step: Solve for Translation (3)

$$\begin{pmatrix} \tilde u_1^1 & \tilde u_2^1 & \cdots & \tilde u_n^1 \\ \tilde v_1^1 & \tilde v_2^1 & & \tilde v_n^1 \\ \tilde u_1^2 & \tilde u_2^2 & & \tilde u_n^2 \\ \tilde v_1^2 & \tilde v_2^2 & & \tilde v_n^2 \\ \vdots & \vdots & & \vdots \\ \tilde u_1^m & \tilde u_2^m & & \tilde u_n^m \\ \tilde v_1^m & \tilde v_2^m & \cdots & \tilde v_n^m \end{pmatrix} = \begin{pmatrix} s_{1,1}^1 & s_{1,2}^1 & s_{1,3}^1 \\ s_{2,1}^1 & s_{2,2}^1 & s_{2,3}^1 \\ s_{1,1}^2 & s_{1,2}^2 & s_{1,3}^2 \\ s_{2,1}^2 & s_{2,2}^2 & s_{2,3}^2 \\ & \vdots & \\ s_{1,1}^m & s_{1,2}^m & s_{1,3}^m \\ s_{2,1}^m & s_{2,2}^m & s_{2,3}^m \end{pmatrix} \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & & y_n \\ z_1 & z_2 & & z_n \end{pmatrix}$$

As if by magic, there's no translation.

Rank Theorem

$$\underbrace{\begin{pmatrix} \tilde u_1^1 & \tilde u_2^1 & \cdots & \tilde u_n^1 \\ \tilde v_1^1 & \tilde v_2^1 & & \tilde v_n^1 \\ \tilde u_1^2 & \tilde u_2^2 & & \tilde u_n^2 \\ \tilde v_1^2 & \tilde v_2^2 & & \tilde v_n^2 \\ \vdots & \vdots & & \vdots \\ \tilde u_1^m & \tilde u_2^m & & \tilde u_n^m \\ \tilde v_1^m & \tilde v_2^m & \cdots & \tilde v_n^m \end{pmatrix}}_{\tilde I} = \underbrace{\begin{pmatrix} s_{1,1}^1 & s_{1,2}^1 & s_{1,3}^1 \\ s_{2,1}^1 & s_{2,2}^1 & s_{2,3}^1 \\ s_{1,1}^2 & s_{1,2}^2 & s_{1,3}^2 \\ s_{2,1}^2 & s_{2,2}^2 & s_{2,3}^2 \\ & \vdots & \\ s_{1,1}^m & s_{1,2}^m & s_{1,3}^m \\ s_{2,1}^m & s_{2,2}^m & s_{2,3}^m \end{pmatrix}}_{S} \underbrace{\begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & & y_n \\ z_1 & z_2 & & z_n \end{pmatrix}}_{P}$$

$\tilde I$ has rank 3.

This means there are 3 vectors such that every row of $\tilde I$ is a linear combination of these vectors. These vectors are the rows of P.

Solve for S

- SVD is made to do this.

$$\tilde I = UDV$$

D is diagonal with non-increasing values.

U and V have orthonormal rows.

Ignoring values that get set to 0, we have U(:,1:3) for S, and

D(1:3,1:3)*V(1:3,:) for P.

Linear Ambiguity (as before)

$\tilde I$ = U(:,1:3) * D(1:3,1:3) * V(1:3,:)

$\tilde I$ = (U(:,1:3) * A) * (inv(A) *D(1:3,1:3) * V(1:3,:))

Noise

- $\tilde I$ has full rank.
- Best solution is to estimate I that's as near to $\tilde I$ as possible, with estimate of I having rank 3.
- Our current method does this.

Weak Perspective Motion

$$\underbrace{\begin{pmatrix} \tilde u_1^1 & \tilde u_2^1 & \cdots & \tilde u_n^1 \\ \tilde v_1^1 & \tilde v_2^1 & & \tilde v_n^1 \\ \tilde u_1^2 & \tilde u_2^2 & & \tilde u_n^2 \\ \tilde v_1^2 & \tilde v_2^2 & & \tilde v_n^2 \\ \vdots & \vdots & & \vdots \\ \tilde u_1^m & \tilde u_2^m & & \tilde u_n^m \\ \tilde v_1^m & \tilde v_2^m & \cdots & \tilde v_n^m \end{pmatrix}}_{\tilde I} = \underbrace{\begin{pmatrix} s_{1,1}^1 & s_{1,2}^1 & s_{1,3}^1 \\ s_{2,1}^1 & s_{2,2}^1 & s_{2,3}^1 \\ s_{1,1}^2 & s_{1,2}^2 & s_{1,3}^2 \\ s_{2,1}^2 & s_{2,2}^2 & s_{2,3}^2 \\ & \vdots & \\ s_{1,1}^m & s_{1,2}^m & s_{1,3}^m \\ s_{2,1}^m & s_{2,2}^m & s_{2,3}^m \end{pmatrix}}_{S} \underbrace{\begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & & y_n \\ z_1 & z_2 & & z_n \end{pmatrix}}_{P}$$

$\tilde I$ =(U(:,1:3)*A)*(inv(A) *D(1:3,1:3)*V(1:3,:))

Row 2k and 2k+1 of S should be orthogonal. All rows should be unit vectors.

(Push all scale into P).

Choose A so (U(:,1:3) * A) satisfies these conditions.

Related problems we won't cover

- Missing data.
- Points with different, known noise.
- Multiple moving objects.

Final Messages

- Structure-from-motion for points can be reduced to linear algebra.
- Epipolar constraint reemerges.
- SVD important.
- Rank Theorem says the images a scene produces aren't complicated (also important for recognition).

This document was created with Win2PDF available at http://www.daneprairie.com.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.