3D Structure and Motion from 2D Motion

Adel H. Fakih

afakih@engmail.uwaterloo.ca

May 25, 2009

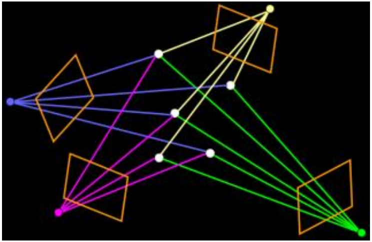
Outline

Outline

1. Introduction
2. Structure from Motion Constraints
3. Instantaneous Constraint Approaches
4. Discrete Constraint Approaches
   - Two Views
   - Extension to Multiple Frames
5. Factorization Methods

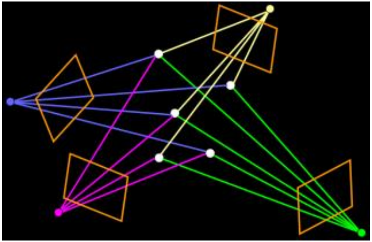
The Structure from Motion Problem

- Input:
  - $\longrightarrow$ Image projection: $\overrightarrow{x}_i$
- Output:
  - $\longrightarrow$ Camera motion: R, $\overrightarrow{T}$, $\overrightarrow{\omega}$, $\overrightarrow{V}$
  - $\longrightarrow$ Structure: Positions of a set of N $3D$ points $\overrightarrow{X}_i$



The Structure from Motion Problem

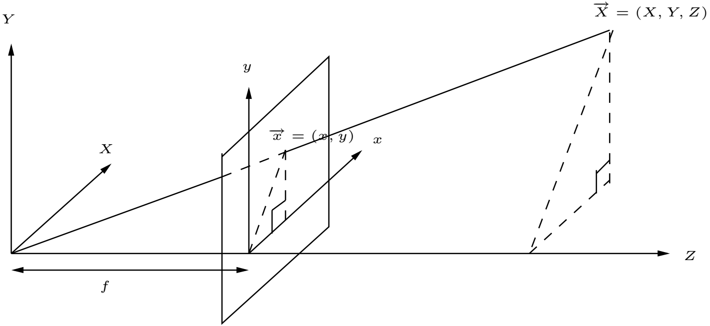
- Input:
  - $\longrightarrow$ Image projection: $\overrightarrow{x}_i$
- Output:
  - $\longrightarrow$ Camera motion: R, $\overrightarrow{T}$, $\overrightarrow{\omega}$, $\overrightarrow{V}$
  - $\longrightarrow$ Structure: Positions of a set of N $3D$ points $\overrightarrow{X}_i$



- Optimal solution: Minimizes the reprojection error in all the images
  - $\longrightarrow$ Known as Bundle Adjustment
  - $\longrightarrow$ Needs to be done offline
  - $\longrightarrow$ Computationally expensive

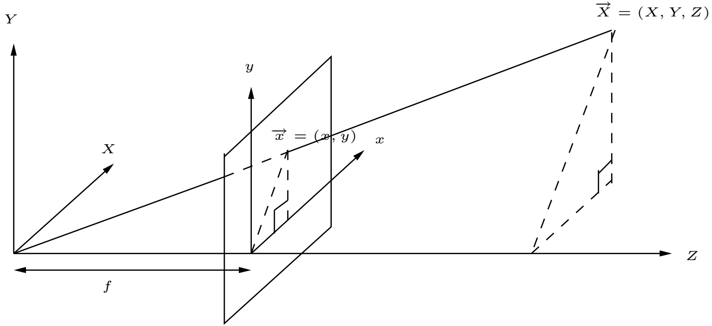
University of Waterloo

Perspective projection



$$\begin{bmatrix} x \\ y \end{bmatrix} = K \begin{bmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \end{bmatrix}$$

Perspective projection



$$\begin{bmatrix} x \\ y \end{bmatrix} = K \begin{bmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \end{bmatrix} \qquad K = \begin{bmatrix} f_1 & \alpha_c f_1 & c_1 \\ 0 & f_2 & c_2 \\ 0 & 0 & 1 \end{bmatrix}$$

University of Waterloo

Outline

SFM Constraints

- Image projections are the result of the camera pose and the 3D positions of the points

SFM Constraints

- Image projections are the result of the camera pose and the 3D positions of the points
- Matching Constraints:
  - $\longrightarrow$ Link 3D parameters to image projections
  - $\longrightarrow$ Equation of motion of rigid bodies + projection equation

SFM Constraints

- Image projections are the result of the camera pose and the 3D positions of the points
- Matching Constraints:
  - $\longrightarrow$ Link 3D parameters to image projections
  - $\longrightarrow$ Equation of motion of rigid bodies + projection equation

	3D motion	Projection	SFM constraint
Discrete	$\overrightarrow{X}_c = R\overrightarrow{X} + \overrightarrow{T}$	$x_c = \frac{X_c}{Z_c}$	$\overrightarrow{x}_c = \frac{[R\overrightarrow{X}+\overrightarrow{T}]_{1,2}}{[R\overrightarrow{X}+\overrightarrow{T}]_3}$
Instantaneous	$\frac{d\overrightarrow{X}_c}{dt} = \overrightarrow{w} \times \overrightarrow{X}_c + \overrightarrow{V}$	$y_c = \frac{Y_c}{Z_c}$	$\dot{\overrightarrow{x}}_c(t) = \frac{A(\overrightarrow{x}_c)\overrightarrow{V}}{Z_c} - B(\overrightarrow{x_c})\overrightarrow{\omega}$

$$A(\overrightarrow{x}_c) = \begin{bmatrix} 1 & 0 & -x_c \\ 0 & 1 & -y_c \end{bmatrix} \quad B(\overrightarrow{x}_c) = \begin{bmatrix} -x_c y_c & 1 + x_c^2 & -y_c \\ -1 - y_c^2 & x_c y_c & y_c \end{bmatrix}$$

University of Waterloo

Outline

Get Rid of the Depth

Algebraic manipulation

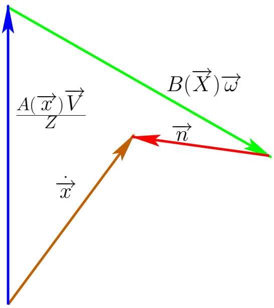
$$\vec{V}^T(\vec{x} \times \dot{\vec{x}}) + (\vec{V} \times \vec{x})(\vec{x} \times \vec{\omega}) = 0$$

Get Rid of the Depth

Algebraic manipulation

$$\vec{V}^T(\vec{x} \times \dot{\vec{x}}) + (\vec{V} \times \vec{x})(\vec{x} \times \vec{\omega}) = 0$$

Geometrical perspective

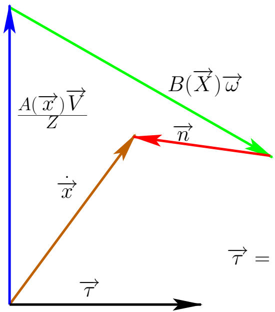


Get Rid of the Depth

Algebraic manipulation

$$\overrightarrow{V}^T(\overrightarrow{x} \times \dot{\overrightarrow{x}}) + (\overrightarrow{V} \times \overrightarrow{x})(\overrightarrow{x} \times \overrightarrow{\omega}) = 0$$

Geometrical perspective



$$\overrightarrow{\tau} = \frac{([A(\overrightarrow{x})\overrightarrow{V}]_y, -[A(\overrightarrow{x})\overrightarrow{V}]_x)^T}{||A(\overrightarrow{x})\overrightarrow{V}||}$$

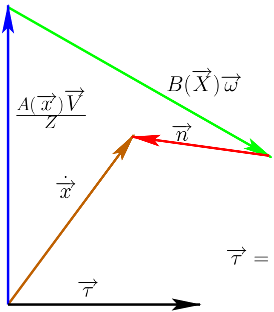
University of Waterloo

Get Rid of the Depth

Algebraic manipulation

$$\overrightarrow{V}^T(\overrightarrow{x} \times \dot{\overrightarrow{x}}) + (\overrightarrow{V} \times \overrightarrow{x})(\overrightarrow{x} \times \overrightarrow{\omega}) = 0$$

Geometrical perspective



$$\overrightarrow{\tau} = \frac{([A(\overrightarrow{x})\overrightarrow{V}]_y, -[A(\overrightarrow{x})\overrightarrow{V}]_x)^T}{||A(\overrightarrow{x})\overrightarrow{V}||}$$

$$\overrightarrow{\tau}^T(\dot{\overrightarrow{x}} - B(\overrightarrow{X})\overrightarrow{\omega}) = 0$$

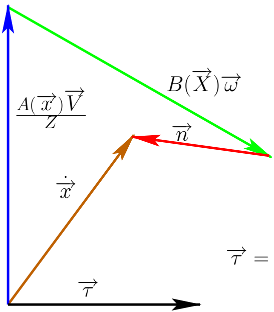
Get Rid of the Depth

Algebraic manipulation

$$\vec{V}^T(\vec{x} \times \dot{\vec{x}}) + (\vec{V} \times \vec{x})(\vec{x} \times \vec{\omega}) = 0$$

$$||A(\vec{x})\vec{V}||\vec{\tau}^T(\dot{\vec{x}} - B(\vec{X})\vec{\omega}) = 0$$

Geometrical perspective



$$\vec{\tau} = \frac{([A(\vec{x})\vec{V}]_y, -[A(\vec{x})\vec{V}]_x)^T}{||A(\vec{x})\vec{V}||}$$

$$\vec{\tau}^T(\dot{\vec{x}} - B(\vec{X})\vec{\omega}) = 0$$

Approaches Using the Weighted Constraint

- Bruss and Horn
  - $\longrightarrow$ a least-squares estimate of $\overrightarrow{\omega}$ is obtained as a function of $\overrightarrow{V}$
  - $\longrightarrow$ Substitute $\overrightarrow{\omega}$ back into the bilinear constraints
  - $\longrightarrow$ Minimize the obtained non-linear constraint subject to $|\overrightarrow{V}| = 1$

Approaches Using the Weighted Constraint

- Bruss and Horn
  - $\longrightarrow$ a least-squares estimate of $\overrightarrow{\omega}$ is obtained as a function of $\overrightarrow{V}$
  - $\longrightarrow$ Substitute $\overrightarrow{\omega}$ back into the bilinear constraints
  - $\longrightarrow$ Minimize the obtained non-linear constraint subject to $|\overrightarrow{V}| = 1$
- Linear Subspaces (Jepson and Heeger)
  - $\longrightarrow$ Define vectors $\overrightarrow{\tau}_i = \sum_{k=1}^{N} c_{ik} [\dot{\overrightarrow{x}}^k \times \overrightarrow{x}^k]$
  - $\longrightarrow$ $\tau_i \overrightarrow{V} = \sum_{k=1}^{N} c_{ik} \overrightarrow{V} [\overrightarrow{x}^k \times [\overrightarrow{x}^k \times \overrightarrow{\omega}]]$
  - $\longrightarrow$ Choose the $\overrightarrow{c}_i$ vectors to be perpendicular to all quadratic polynomials on image plane
  - $\longrightarrow$ $\overrightarrow{\tau}_i \overrightarrow{V} = 0$
  - $\longrightarrow$ Solution: Smallest eigenvalue of $\sum \overrightarrow{\tau}_i \overrightarrow{\tau}_i^T$

Approaches Using the Un-weighted Constraint

- Zhang and Tomasi:
  - ⟶ Gauss-Newton minimization of $\sum(\dot{\overrightarrow{x}} - \frac{A(\overrightarrow{x}_c)\overrightarrow{V}}{Z_c} - B(\overrightarrow{x_c})\overrightarrow{\omega})^2$
  - ⟶ At each step:
    - Determine $\overrightarrow{V}$, $\overrightarrow{\omega}$, and Z for every point
    - Re-determine $\overrightarrow{\omega}$ from the unweighted constraint using $\overrightarrow{V}$
    - Re-determine the Z from the full instantaneous constraint
  - ⟶ Initialize from 15 different locations to avoid local-minima

University of Waterloo

Approaches Using the Un-weighted Constraint

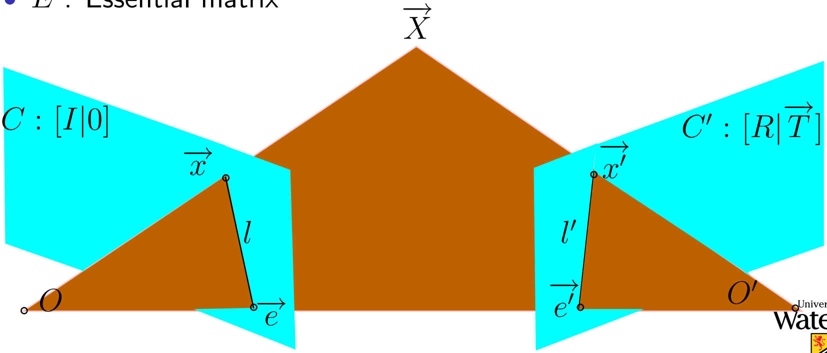
- Zhang and Tomasi:
  - $\longrightarrow$ Gauss-Newton minimization of $\sum(\dot{\overrightarrow{x}} - \frac{A(\overrightarrow{x}_c)\overrightarrow{V}}{Z_c} - B(\overrightarrow{x_c})\overrightarrow{\omega})^2$
  - $\longrightarrow$ At each step:
    - Determine $\overrightarrow{V}$, $\overrightarrow{\omega}$, and Z for every point
    - Re-determine $\overrightarrow{\omega}$ from the unweighted constraint using $\overrightarrow{V}$
    - Re-determine the Z from the full instantaneous constraint
  - $\longrightarrow$ Initialize from 15 different locations to avoid local-minima
- Pauwels and Van-Hulle:
  - $\longrightarrow$ Minimizes $(||A(\overrightarrow{x})\overrightarrow{V}||^{(1-\rho)}\overrightarrow{\tau}^T(\dot{\overrightarrow{x}} - B(\overrightarrow{x})\overrightarrow{\omega}))^2$
  - $\longrightarrow$ $\rho = 0 \Rightarrow$ weighted
  - $\longrightarrow$ $\rho = 1 \Rightarrow$ unweighted

Outline

Two Views

- $\overrightarrow{x'}^T [\overrightarrow{T}]_\times R \overrightarrow{x} = 0$
- $E = [\overrightarrow{T}]_\times R$
- E : Essential matrix

$$[\overrightarrow{T}]_\times \overset{\text{def}}{=} \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix}$$



Essential Matrix

- $E\overrightarrow{e} = 0$ and $\overrightarrow{e'}E = 0$
- E is singular
- E has two equal non-zero singular values
- $SVD : E = UWV^T$

  $\longrightarrow W = \begin{pmatrix} \alpha & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 0 \end{pmatrix}$

  $\longrightarrow V, U \in SO_3$

Essential Matrix Determination

$$(x', y', 1) \begin{pmatrix} (E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0$$

$$(xx', xy', x, yx', yy', y, x', y') \begin{pmatrix} E_{11} \\ E_{12} \\ E_{13} \\ E_{21} \\ E_{22} \\ E_{23} \\ E_{31} \\ E_{32} \end{pmatrix} = -1$$

Minimize
$\sum_n (\overrightarrow{x}^T E \overrightarrow{x}')^2$
subject to $\|E\| = 1$

University of Waterloo

5points Algorithm

- Combine

$$EE^TE - \frac{1}{2}trace(EE^T)E = 0$$

with the 5 equations:

$$\overrightarrow{x_i'}^T E \overrightarrow{x_i} = 0, \ \forall\, i \in \{1...5\}$$

$\longrightarrow$ Get a 10^{th} order polynomial equation

- Gives up to 10 solutions

Random Sample Consensus (RANSAC)

- Select 5 data items at random
- Estimate the correspondent essential matrix
- Find the number of inliers k
- If K is big enough, accept and exit.
- Repeat L times
- L is determined based on the expected number of outliers and the desired probability of success

University of Waterloo

Random Sample Consensus (RANSAC)

- Select 5 data items at random
- Estimate the correspondent essential matrix
- Find the number of inliers k
- If K is big enough, accept and exit.
- Repeat L times
- L is determined based on the expected number of outliers and the desired probability of success
- Preemptive Ransac techniques

  $\longrightarrow$ Breadth first

Motion from Essential Matrix

- SVD: $E = UWV^T$

Motion from Essential Matrix

- SVD: $E = UWV^T$
- Four possible solutions:

  $\longrightarrow$ $[T_1]_\times = UR_z(\frac{\pi}{2})WU^T$

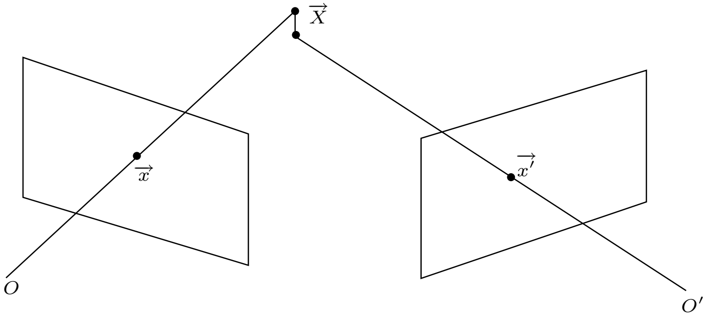
  $\longrightarrow$ $[T_2]_\times = UR_z(-\frac{\pi}{2})WU^T$

  $\longrightarrow$ $R_1 = UR_z(\frac{\pi}{2})V^T$

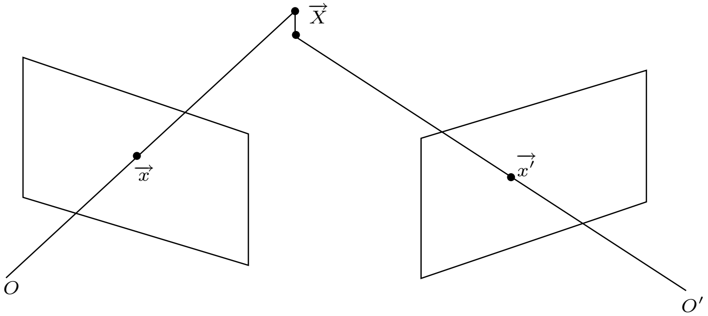
  $\longrightarrow$ $R_2 = UR_z(-\frac{\pi}{2})V^T$

- $R_z(\frac{\pi}{2}) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Triangulation

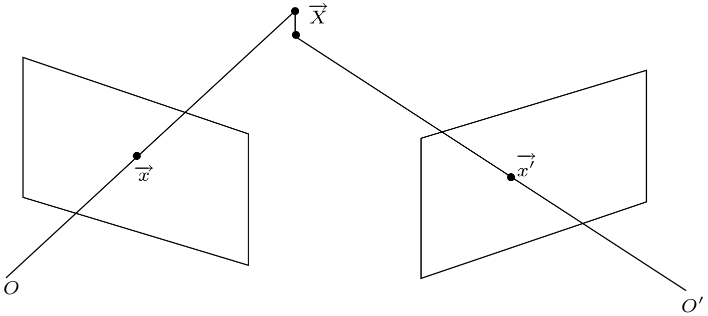


Triangulation



- Linear: From the projection equations get a system of the form: $A\overrightarrow{X} = 0$

Triangulation

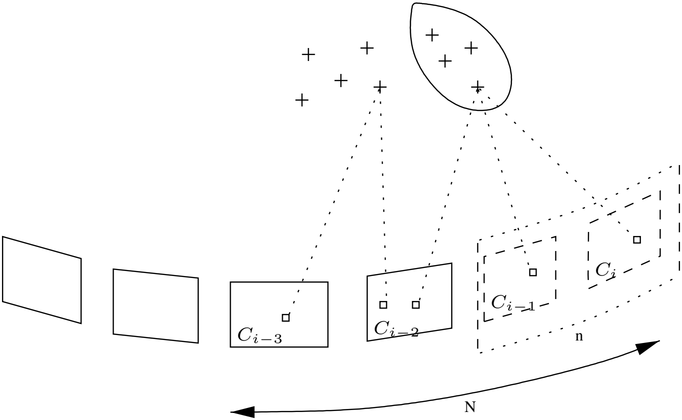


- Linear: From the projection equations get a system of the form: $A\vec{X} = 0$
- Non-Linear: Iteratively find $\vec{\hat{X}}$ that minimizes $(\vec{x} - \vec{\hat{x}})^2 + (\vec{x'} - \vec{\hat{x'}})$
  $\longrightarrow$ $\vec{\hat{x}}$ and $\vec{\hat{x'}}$ are the projections of $\vec{\hat{X}}$

University of Waterloo

Local Bundle Adjustment

- Royer and Lhuillier



Filter Based Approaches

- Organize the time evolution of the 3D parameters as a dynamical system

State Equations

$$\begin{cases} \overrightarrow{X}_i(t+1) = \overrightarrow{X}_i(t) \\ \overrightarrow{T}(t+1) = e^{\hat{\overrightarrow{\omega}}(t)}\overrightarrow{T}(t) + \overrightarrow{V}(t) \\ R(t+1) = e^{\hat{\overrightarrow{\omega}}(t)}R(t) \\ \overrightarrow{V}(t+1) = \overrightarrow{V}(t) + \overrightarrow{a}_V(t) \\ \overrightarrow{\omega}(t+1) = \overrightarrow{\omega}(t) + \overrightarrow{a}_\omega(t) \end{cases}$$

Measurement equations

$$\overrightarrow{x}_i(t) = \Pi(R(t)\overrightarrow{X}_i(t) + \overrightarrow{T}(t))$$

University of Waterloo

Filter Based Approaches

- Organize the time evolution of the 3D parameters as a dynamical system

State Equations

$$\begin{cases} \overrightarrow{X}_i(t+1) = \overrightarrow{X}_i(t) \\ \overrightarrow{T}(t+1) = e^{\hat{\overrightarrow{\omega}}(t)}\overrightarrow{T}(t) + \overrightarrow{V}(t) \\ R(t+1) = e^{\hat{\overrightarrow{\omega}}(t)}R(t) \\ \overrightarrow{V}(t+1) = \overrightarrow{V}(t) + \overrightarrow{a}_V(t) \\ \overrightarrow{\omega}(t+1) = \overrightarrow{\omega}(t) + \overrightarrow{a}_\omega(t) \end{cases}$$

Measurement equations

$$\overrightarrow{x}_i(t) = \Pi(R(t)\overrightarrow{X}_i(t) + \overrightarrow{T}(t))$$

- Can be solved using an Extended Kalman Filter

University of Waterloo

Filter Based Approaches

- Organize the time evolution of the 3D parameters as a dynamical system

State Equations

$$\begin{cases} \overrightarrow{X}_i(t+1) &= \overrightarrow{X}_i(t) \\ \overrightarrow{T}(t+1) &= e^{\hat{\overrightarrow{\omega}}(t)}\overrightarrow{T}(t) + \overrightarrow{V}(t) \\ R(t+1) &= e^{\hat{\overrightarrow{\omega}}(t)}R(t) \\ \overrightarrow{V}(t+1) &= \overrightarrow{V}(t) + \overrightarrow{a}_V(t) \\ \overrightarrow{\omega}(t+1) &= \overrightarrow{\omega}(t) + \overrightarrow{a}_\omega(t) \end{cases}$$

Measurement equations

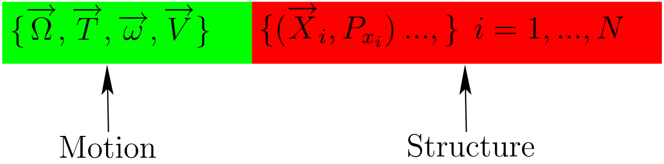
$$\overrightarrow{x}_i(t) = \Pi(R(t)\overrightarrow{X}_i(t) + \overrightarrow{T}(t))$$

- Can be solved using an Extended Kalman Filter
  $\longrightarrow$ Problem: Dimensionality explosion (quadratic in the number of features)

University of Waterloo

Rao-BlackWellized Recursive Structure from Motion

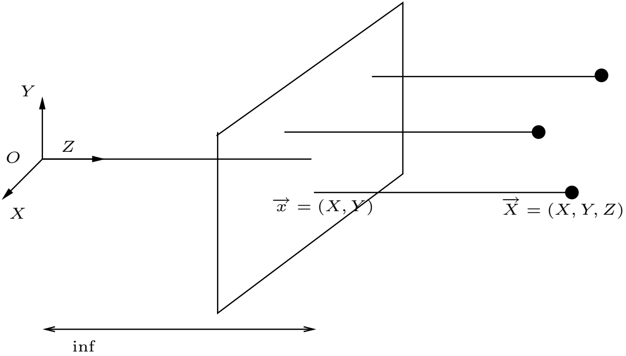
- Particle filtering based
- Splits the state vector into two parts
  - ⟶ Motion: estimated using particle filtering
  - ⟶ Structure: The 3D points in every particle are estimated analytically conditioned on the motion in the sample



Outline

Tomasi-Kanade Factorization (1)

- Applies to orthographic projections



$\longrightarrow$ Constraint : $\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} [R\vec{X} + \vec{T}]_x \\ R\vec{X} + \vec{T}]_y \end{bmatrix} = A\vec{X} + \vec{b}$

$\longrightarrow$ A is 2×3 and $\vec{b}$ is 2×1

Tomasi-Kanade Factorization (2)

- M cameras $(A_j, \overrightarrow{b}_j)$
- N points $(\overrightarrow{X}_i)$
- $\overrightarrow{x}_{ij}$: projection of $\overrightarrow{X}_i$ on the j^{th} camera
- Take one of the points (or their center of mass) as the origin
  $\longrightarrow$ $A_j\overrightarrow{X}_i + \overrightarrow{b}_j \Leftrightarrow A_j\overrightarrow{X}_i$
- Define D,P,A:

$$D \stackrel{\text{def}}{=} \begin{bmatrix} \overrightarrow{x}_{11} & \dots & \overrightarrow{x}_{N1} \\ . & . & . \\ . & . & . \\ . & . & . \\ \overrightarrow{x}_{1M} & \dots & \overrightarrow{x}_{NM} \end{bmatrix}$$

$$P \stackrel{\text{def}}{=} \begin{bmatrix} \overrightarrow{X}_1 & \dots & \overrightarrow{X}_N \end{bmatrix}$$

$$A \stackrel{\text{def}}{=} \begin{bmatrix} A_1 \\ . \\ . \\ A_M \end{bmatrix}$$

- $D = A \times P$ (D at most rank 3!)

Tomasi-Kanade Factorization (3)

- How to recover A and P from D
- SVD: $D = UWV^T$
- U is $2M \times 2M$, W is $2M \times N$ and V is $N \times N$
- W is diagonal with diagonal values $\lambda_1, ..., \lambda_{2M}$
- The best rank 3 approximation of D is
$$\begin{bmatrix} U_1 & U_2 & U_3 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \\ V_3^T \end{bmatrix}$$
- Take $A = \begin{bmatrix} U_1 & U_2 & U_3 \end{bmatrix}$ and
$$P = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \\ V_3^T \end{bmatrix}$$

University of Waterloo

The Perspective Case

- D in this case is $\begin{bmatrix} Z_{11}\overrightarrow{x}_{11} & \dots & Z_{N1}\overrightarrow{x}_{N1} \\ . & . & . \\ . & . & . \\ . & . & . \\ Z_{1M}\overrightarrow{x}_{1M} & \dots & Z_{NM}\overrightarrow{x}_{NM} \end{bmatrix}$

The Perspective Case

- D in this case is $\begin{bmatrix} Z_{11}\overrightarrow{x}_{11} & \dots & Z_{N1}\overrightarrow{x}_{N1} \\ . & . & . \\ . & . & . \\ . & . & . \\ Z_{1M}\overrightarrow{x}_{1M} & \dots & Z_{NM}\overrightarrow{x}_{NM} \end{bmatrix}$
- Sturm and Triggs:
  - $\longrightarrow$ Guess the depths
  - $\longrightarrow$ Factorize D to get M and P
  - $\longrightarrow$ Iterate

University of Waterloo

Thank you

Questions???