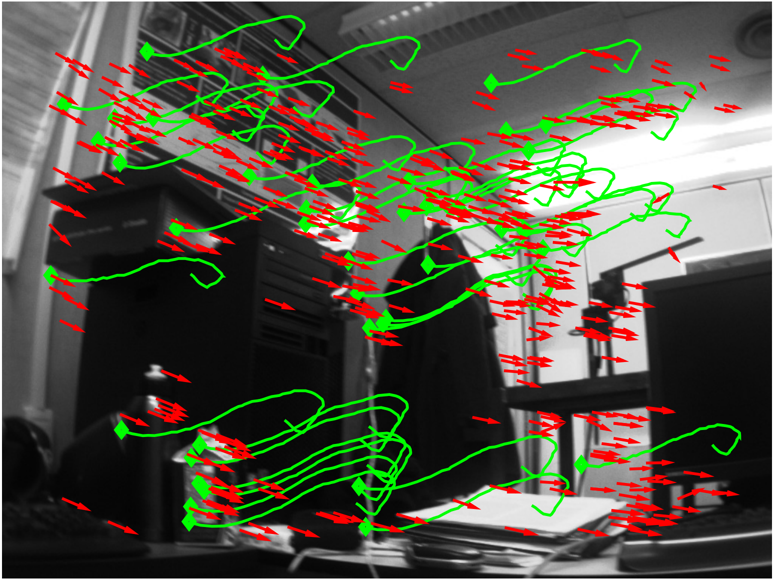
Efficient Augmentation of the EKF Structure from Motion with Frame-to-Frame Features

Adel Fakih and John Zelek

{afakih,jzelek}@engmail.uwaterloo.ca

June 2, 2010



Outline

Outline

Introduction

- Frame-to-Frame Features
  - Provide additional constraints on the velocity.
  - Abundant (Under Gaussian noise assumption, consistent estimation leads to higher accuracy).
  - Have been used earlier in:
    - *bundle adjustment*.
    - Particle Filtering (only in weighting the particles).
  - Their direct insertion in the EKF is very costly (cubic complexity).

Problem Formulation

- 3D Parameters to estimate
  - Motion:
    - Pose $[\overrightarrow{\Omega}; \overrightarrow{T}]$
    - Velocity $[\overrightarrow{\omega}; \overrightarrow{V}]$
  - Structure ($3D$ points): $\overrightarrow{\mathbf{X}} = [\overrightarrow{X}^1; ...; \overrightarrow{X}^N]$
  - $\overrightarrow{S} = [\overrightarrow{S}^1; \overrightarrow{S}^2]$
  - $\overrightarrow{S}^1 = [\overrightarrow{\Omega}; \overrightarrow{T}; \overrightarrow{\mathbf{X}}]$
  - $\overrightarrow{S}^2 = [\overrightarrow{\omega}; \overrightarrow{V}]$
- Measurements
  - Tracked Features: $\overrightarrow{\mathbf{y}}(t) = [\overrightarrow{y}^1(t); ...; \overrightarrow{y}^N(t)]$
  - Frame-To-Frame features:
    $\overrightarrow{\mathbf{z}}(t) = [\overrightarrow{z}^1(t-1); \overrightarrow{z}^1(t); ...; \overrightarrow{z}^K(t-1); \overrightarrow{z}^K(t)]$

University of Waterloo

Dynamical System

- Transition Equations:

$$\begin{cases} \overrightarrow{\mathbf{X}}(t+1) & = & \overrightarrow{\mathbf{X}}(t) \\ \overrightarrow{T}(t+1) & = & e^{[\overrightarrow{\omega}(t)]_\times} \overrightarrow{T}(t) + \overrightarrow{V}(t) \\ \overrightarrow{\Omega}(t+1) & = & Log_{SO3}(e^{[\overrightarrow{\omega}(t)]_\times} e^{[\overrightarrow{\Omega}(t)]_\times}) \\ \overrightarrow{V}(t+1) & = & \overrightarrow{V}(t) + \overrightarrow{a}_V(t) \\ \overrightarrow{\omega}(t+1) & = & \overrightarrow{\omega}(t) + \overrightarrow{a}_\omega(t) \end{cases}$$

- Measurement Equations:

$$\begin{cases} \overrightarrow{y}^i(t) & = & g(R(t), \overrightarrow{T}(t), \overrightarrow{X}^i) & = & proj(R(t)\overrightarrow{X}^i + \overrightarrow{T}) \\ & & h(\overrightarrow{V}, \overrightarrow{\omega}, \overrightarrow{z}(t-1), \overrightarrow{z}(t)) & = & \frac{r}{\nabla r} = 0 \end{cases}$$

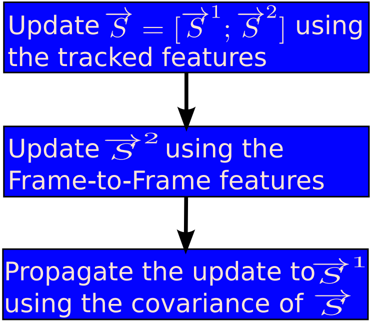
$$r = \overrightarrow{z}(t-1)^T \underbrace{[\overrightarrow{V}]_\times e^{[\overrightarrow{V}]_\times}}_{\text{Essential Matrix}} \overrightarrow{z}(t)$$

University of Waterloo

Outline

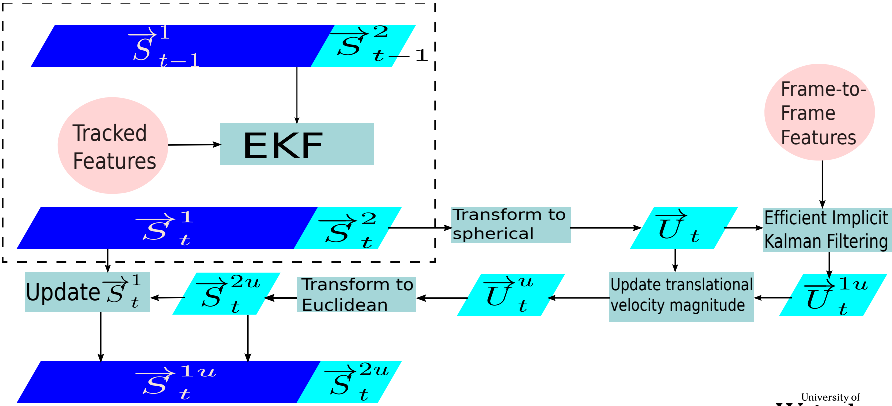
Proposed Approach

- Fold the Frame-to-Frame information in a separate filtering Step
  - By capitalizing on the special structure of the covariance matrix, the computational complexity can be reduced from cubic to linear
  - Can be divided into several steps
  - Can be done in a Random Sample Consensus way to get rid of outliers
  - Steps can be computed in parallel



University of Waterloo

Flowchart


$\overrightarrow{S}^1_{t-1}$
$\overrightarrow{S}^2_{t-1}$
Tracked Features
EKF
Frame-to-Frame Features
$\overrightarrow{S}^1_t$
$\overrightarrow{S}^2_t$
Transform to spherical
$\overrightarrow{U}_t$
Efficient Implicit Kalman Filtering
Update $\overrightarrow{S}^1_t$
$\overrightarrow{S}^{2u}_t$
Transform to Euclidean
$\overrightarrow{U}^u_t$
Update translational velocity magnitude
$\overrightarrow{U}^{1u}_t$
$\overrightarrow{S}^{1u}_t$
$\overrightarrow{S}^{2u}_t$

Outline

Efficient Implicit Extended Kalman Filtering

- Implicit measurement equation: $h(\overrightarrow{\mathbf{z}}, \overrightarrow{U}^1) = 0$

$$\overrightarrow{U}^{1u} = \overrightarrow{U}^1 + Lh(\overrightarrow{U}^1, \overrightarrow{\mathbf{z}})$$
$$\Sigma^u_{\overrightarrow{U}^1} = \Gamma \Sigma_{\overrightarrow{U}^1} \Gamma^T + L R_{\overrightarrow{Z}} L^T$$
$$R_{\overrightarrow{Z}} = H_{\overrightarrow{z}} \Sigma_{\overrightarrow{z}} H^T_{\overrightarrow{z}}$$

$$\Lambda = H_{\overrightarrow{U}^1} \Sigma_{\overrightarrow{U}^1} H^T_{\overrightarrow{U}^1} + R_{\overrightarrow{Z}}$$
$$L = -\Sigma_{\overrightarrow{U}^1} H^T_{\overrightarrow{U}^1} (\Lambda)^{-1}$$
$$\Gamma = I_K - LH$$

- Sherman-Morrison-Woodbury formula + some manipulations:

$$L = -A + B(I_6 + B)^{-1} A$$
$$LH_{\overrightarrow{U}^1} = -B + B(I_6 + B)^{-1} B$$

$$A = (\Sigma_{\overrightarrow{U}^1} H^T_{\overrightarrow{U}^1}) R^{-1}_{\overrightarrow{\mathbf{z}}}$$
$$B = (\Sigma_{\overrightarrow{U}^1} H^T_{\overrightarrow{U}^1})(R^{-1}_{\overrightarrow{\mathbf{z}}} H_{\overrightarrow{U}^1})$$

Reduced Cost: Number of Multiplication Operations

$R_{\vec{\mathbf{z}}}^{-1} H_{\vec{U}1}$	$6K$
$\Sigma_{\vec{U}1} H_{\vec{U}1}^{T}$	$36K$
$A = (\Sigma_{\vec{U}1} H_{\vec{U}1}^{T}) R_{\vec{\mathbf{z}}}^{-1}$	$6K$
$B = (\Sigma_{\vec{U}1} H_{\vec{U}1}^{T})(R_{\vec{\mathbf{z}}}^{-1} H_{\vec{U}1})$	$36K$
L	$36K + 216$
$L H_{\vec{U}1}$	432
$\Sigma_{\vec{U}1}^{u}$	$42K + 432$
$\vec{U}^{1u}$	$6K$
Total	$162K + 1080$

Update Propagation Using the Covariance Matrix

- $\vec{\mu}$ and Σ partitioned as follows:

$$\vec{\mu} = \begin{bmatrix} \vec{\mu}^{1} \\ \vec{\mu}^{2} \end{bmatrix} \qquad \Sigma = \begin{bmatrix} \Sigma^{11} & \Sigma^{12} \\ \Sigma^{21} & \Sigma^{22} \end{bmatrix}$$

- then if $\vec{\mu}^{2}$ and Σ^{22} are updated to be $\vec{\mu}^{2u}$ and Σ^{22u}, then $\vec{\mu}^{11}$,Σ^{11} and Σ^{12} should be updated as follows:

$$\vec{\mu}^{1u} = \vec{\mu}^{1} + \Sigma^{12}(\Sigma^{22})^{-1}(\vec{\mu}^{2u} - \vec{\mu}^{2})$$

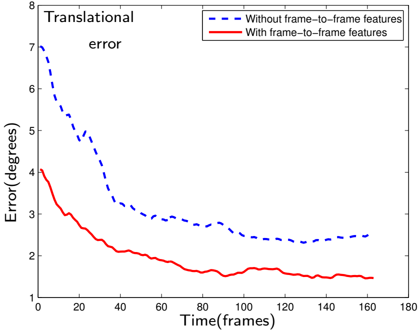
$$\Sigma^{12u} = \Sigma^{12}(\Sigma^{22})^{-1}\Sigma^{22u}$$

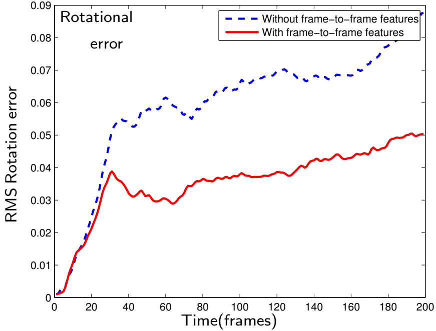
$$\Sigma^{11u} = \Sigma^{11} - \Sigma^{12}(\Sigma^{22})^{-1}(\Sigma^{22} - \Sigma^{22u})(\Sigma^{22})^{-1}\Sigma^{21}$$

University of Waterloo

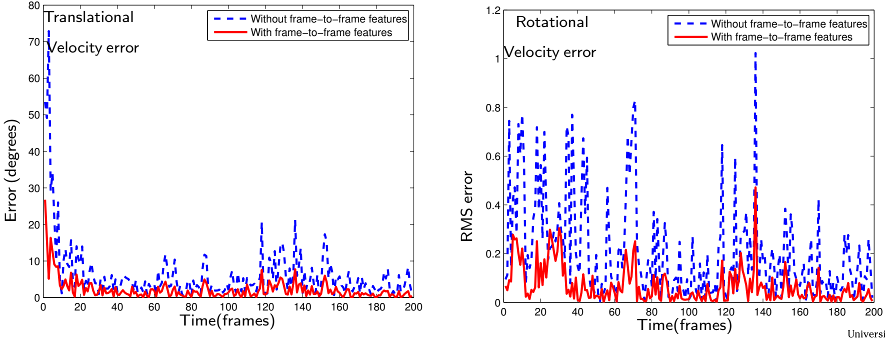
Outline

Pose Error

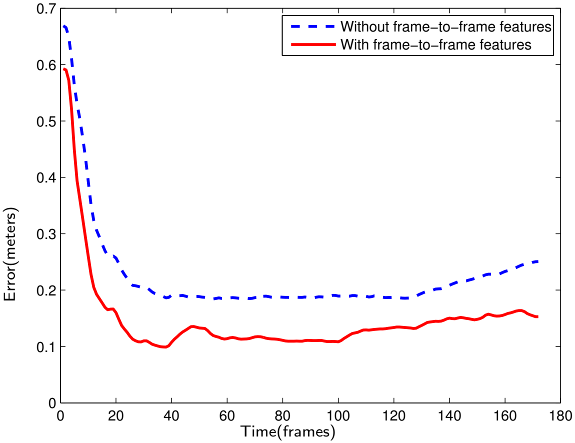

Translational
error
Without frame–to–frame features
With frame–to–frame features
Error(degrees)
Time(frames)
8
7
6
5
4
3
2
1
0
20
40
60
80
100
120
140
160
180


Rotational
error
Without frame–to–frame features
With frame–to–frame features
RMS Rotation error
Time(frames)
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
20
40
60
80
100
120
140
160
180
200

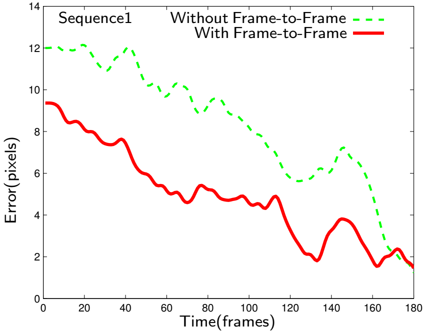
Velocity Error

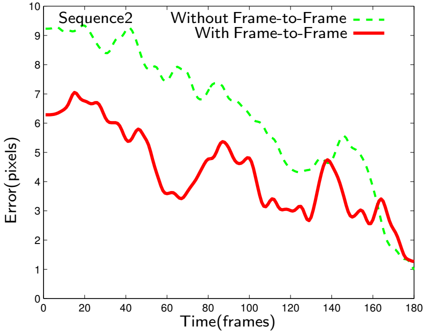


3D Points Error



Reprojection Error on Real Images





Thank you

Questions and comments?