Vision-based Dynamic Target Trajectory and Ego-motion Estimation Using Incremental Light Bundle Adjustment

Michael Chojnacki
Under the supervision of Asst. Prof. Vadim Indelman
and co-supervision of Prof. Ehud Rivlin

Graduate Seminar, December 2016

TECHNION Israel Institute of Technology
ANPL Autonomous Navigation and Perception Lab
TECHNION AUTONOMOUS SYSTEMS PROGRAM

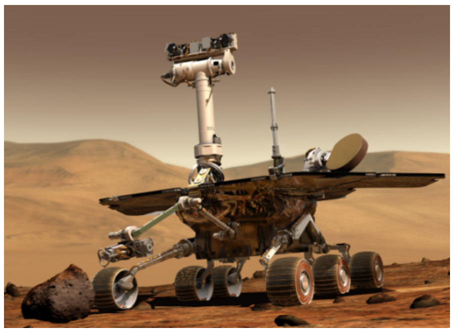
Overview

- Motivations
- Problem Formulation
- iLBA and Dynamic Target Tracking
- Optimization method
- Experiments Results
- Conclusions

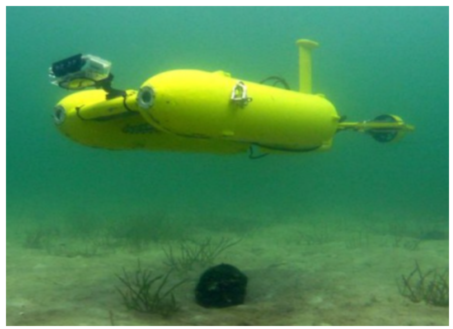
→ Why Motion Estimation ?

- Autonomous Navigation

Space Exploration



Sub-marine exploration



Autonomous Driving

Indoor Operation

- Others

Virtual/Augmented Reality

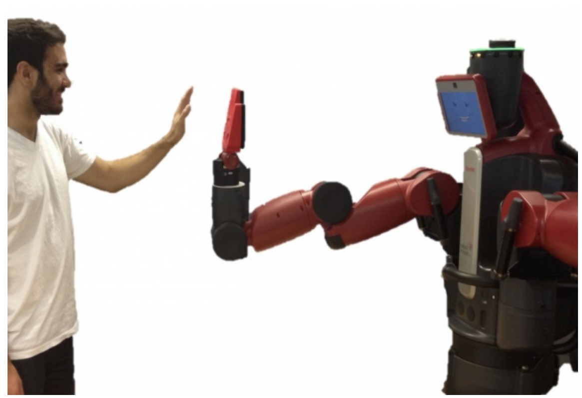
Pointing Devices

→ Why Target Tracking ?

Surveillance

Military

Robot – Human interaction



Scenario

- Unknown environment
- No prior information about platform's trajectory
- No prior information about target's trajectory although motion model is assumed

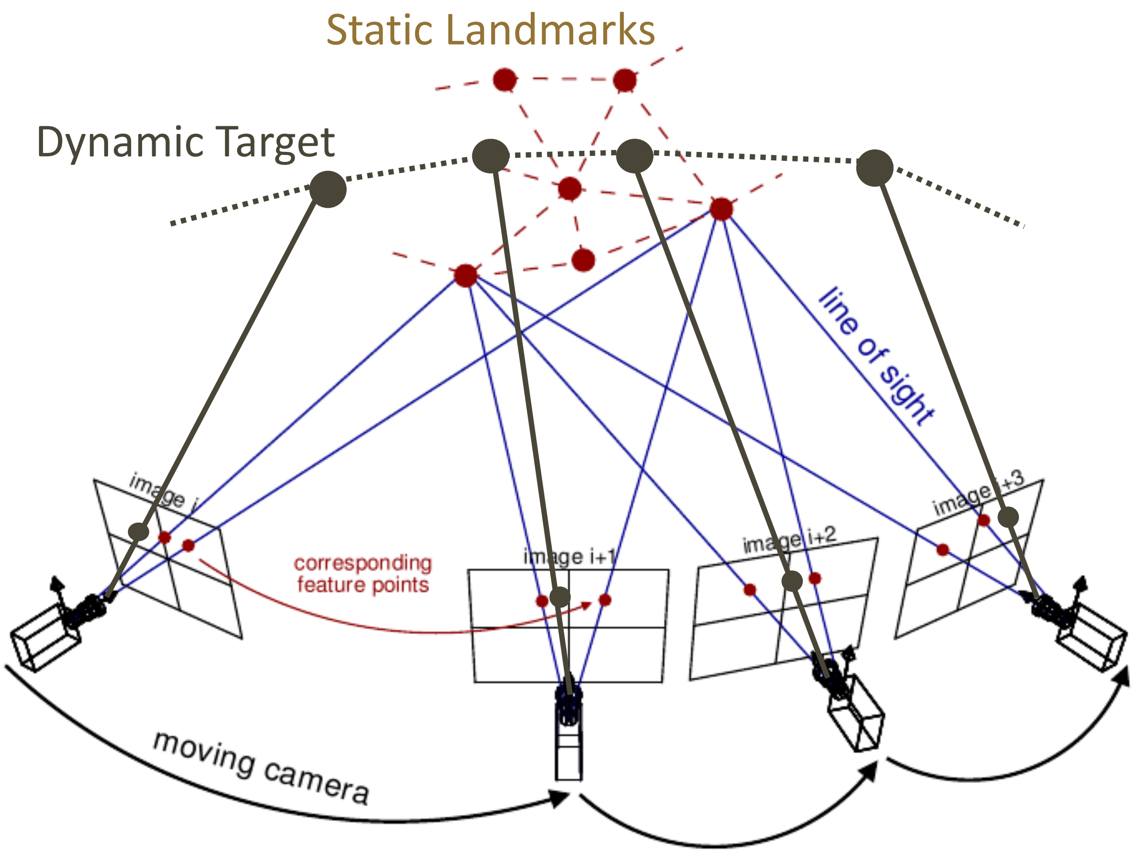
- Interested in on-line operation
- No Global Positioning System

→ Use of onboard sensors : <u>Monocular Camera</u>

Scenario

Efficiently and simultaneously estimate ego-motion and target trajectory

Simultaneous Localization and Mapping (SLAM)



Related Work

- Target Tracking (or DATMO) : [Y. Bar-Shalom, 1988], [M. Breitenstein, 2009]
  - Assume **known/highly predictable sensor location**

- Combined SLAM and DATMO : [J. S. Ortega , 2007], [C. Wang, 2004], [T. D. Vu, 2009]
  - Different techniques : EKF, PF, …
  - All involve optimization over the camera's state, the target's state **and the observed 3D structure !**

- "Structure-Less" BA : [Steffen et al., 2010], [Indelman, 2012]
  - All perform **batch optimization**

[www.cs.cmu.edu]

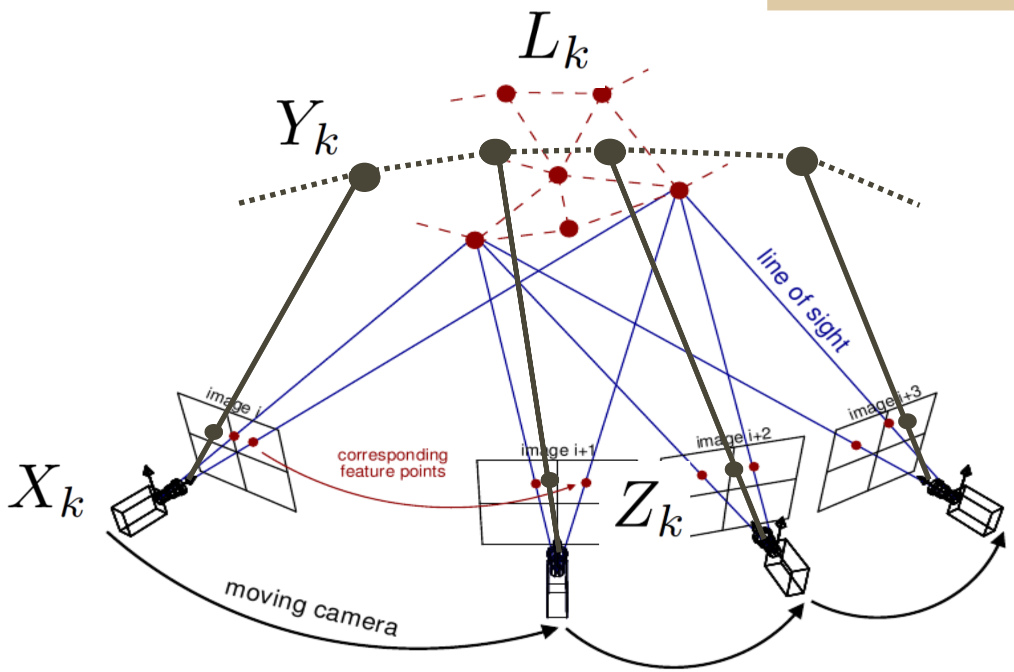
Contributions

- Present Ego-motion estimation and Target tracking as a **combined optimization problem**

- **Integrate target tracking into efficient “structure-less” BA framework :**
  Use Incremental Light Bundle Adjustment (iLBA **[Indelman et al., 2015]**) to :
  - Improve computational efficiency compared to BA
  - Incremental optimization : Re-use calculations from previous steps

- Show results from **simulations** and **real-imagery experiments** performed at ANPL

Notations



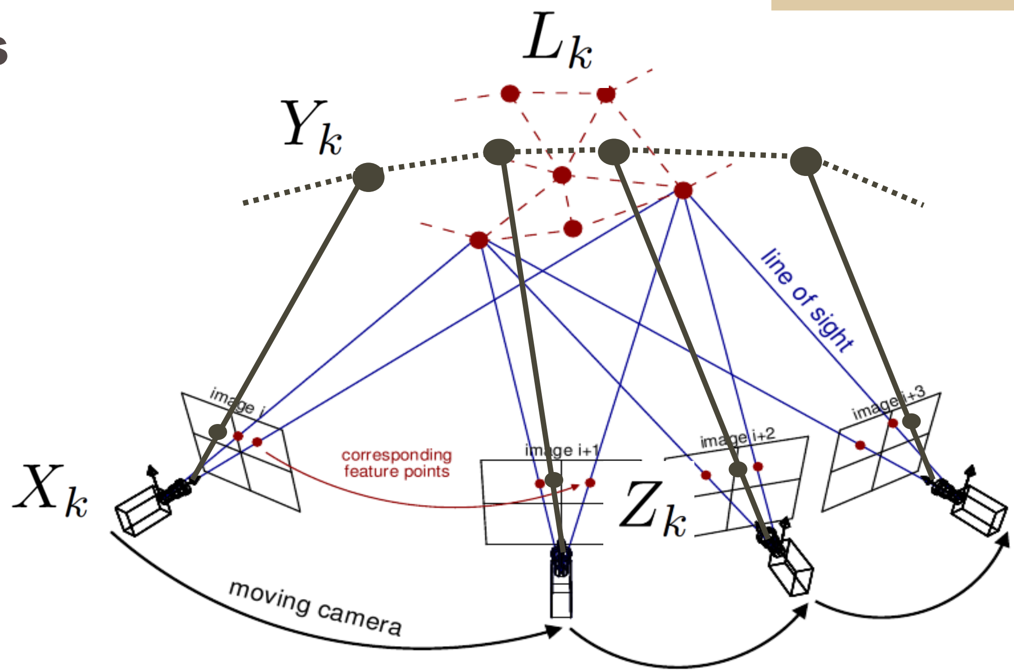
$X_k \doteq \{ \; x_1 \quad \ldots \quad x_k \; \}$ where x_k is the 6DOF camera pose at time *k*

$Y_k \doteq \{ \; y_1 \quad \ldots \quad y_k \; \}$ where $y_k = \begin{bmatrix} y_k^T \\ \dot{y}_k^T \end{bmatrix}$

$L_k \doteq \{ \; l_1 \quad \ldots \quad l_n \; \}$

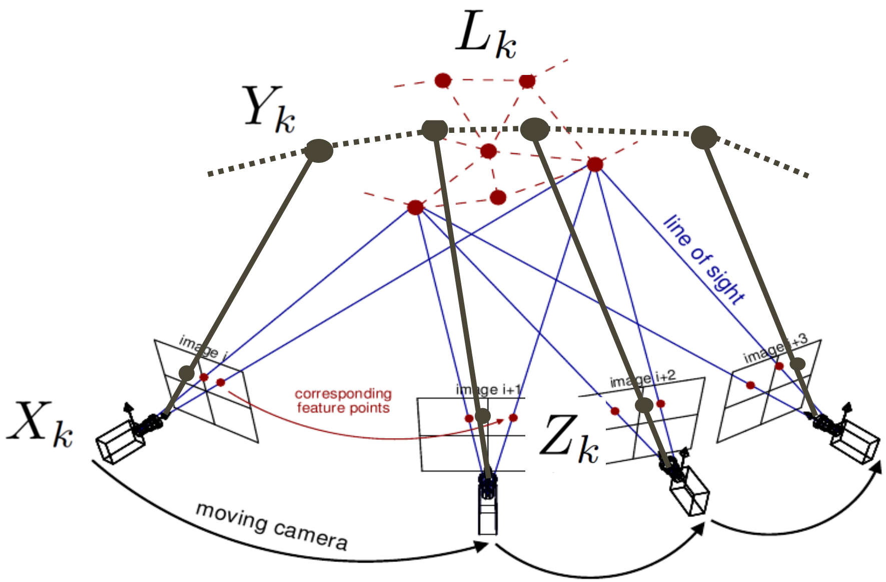
$Z_k \doteq \{ \; z_1 \quad \ldots \quad z_k \; \}$ where $z_k = \left[z_k^1, \ldots, z_k^i \right]$

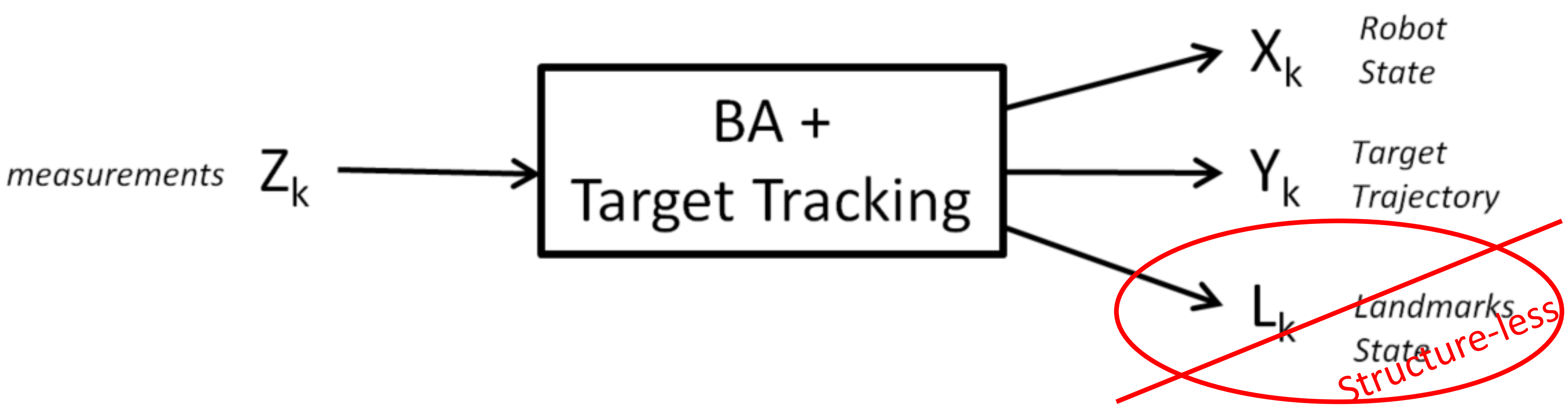
Assumptions



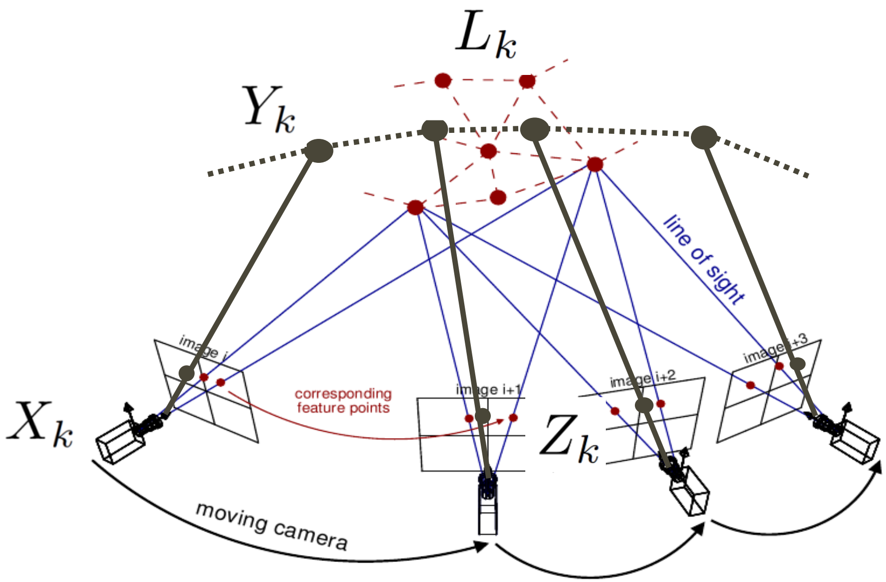
- Known image correspondences for landmarks & target
- White-Gaussian Noises
- Markov process : Models depend only on the current state and previous state
- Target is represented by a single landmark
- Prior information on first camera pose and target state

Problem Formulation : BA and Target Tracking





Problem Formulation : BA and Target Tracking



- Joint probability distribution function (pdf)

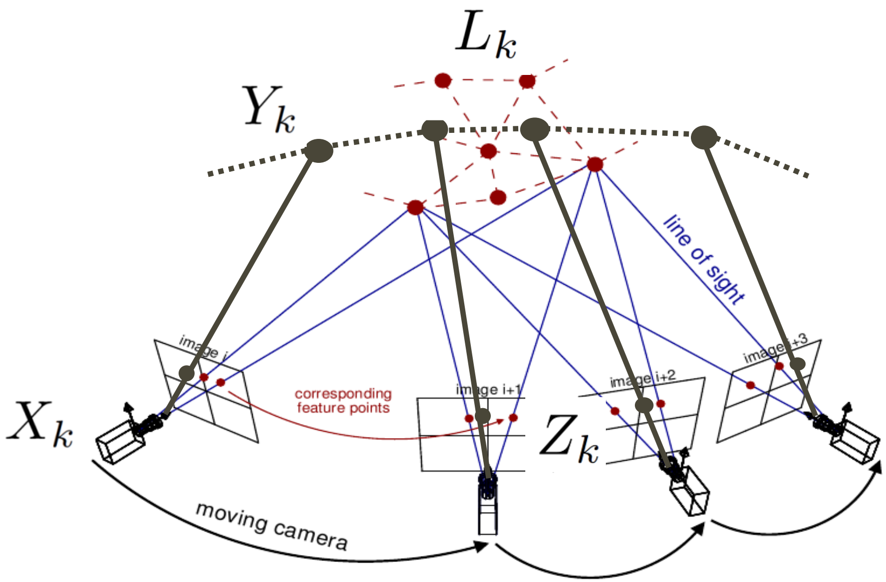
$$P\left(X_k, Y_k, L_k | Z_k\right) \propto \boxed{priors} \prod_{i=1}^{k} \left(\boxed{p\left(y_i | y_{i-1}\right)} \boxed{p\left(z_i^T | x_i, y_i\right)} \prod_{j \in \mathcal{M}_i} \boxed{p\left(z_i^j | x_i, l_j\right)} \right)$$

Prior Information **Motion Model** **Measurement Model**

- Maximum a posteriori (MAP) estimate :

$$X_k^*, Y_k^*, L_k^* = \underset{X_k, Y_k, L_k}{\arg\max}\, P\left(X_k, Y_k, L_k | Z_k\right)$$

Problem Formulation : BA and Target Tracking



- Joint probability distribution function (pdf)

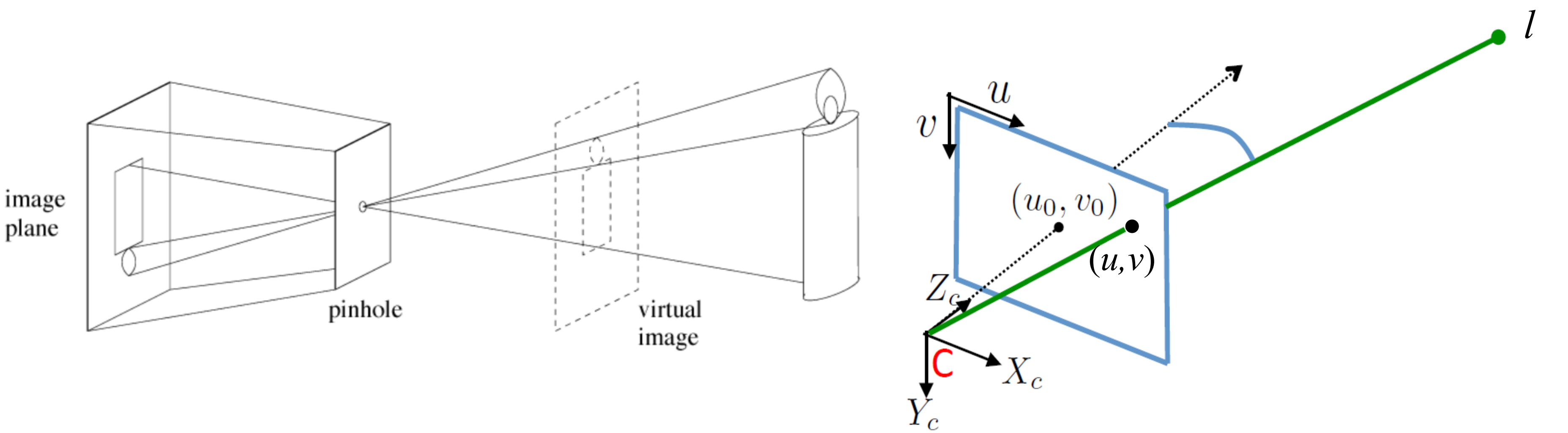
$$P\left(X_k, Y_k, L_k | Z_k\right) \propto priors \cdot \prod_{i=1}^{k} \left(p\left(y_i | y_{i-1}\right) p\left(z_i^T | x_i, y_i\right) \prod_{j \in \mathcal{M}_i} p\left(z_i^j | x_i, l_j\right) \right)$$

Prior Information

Motion Model

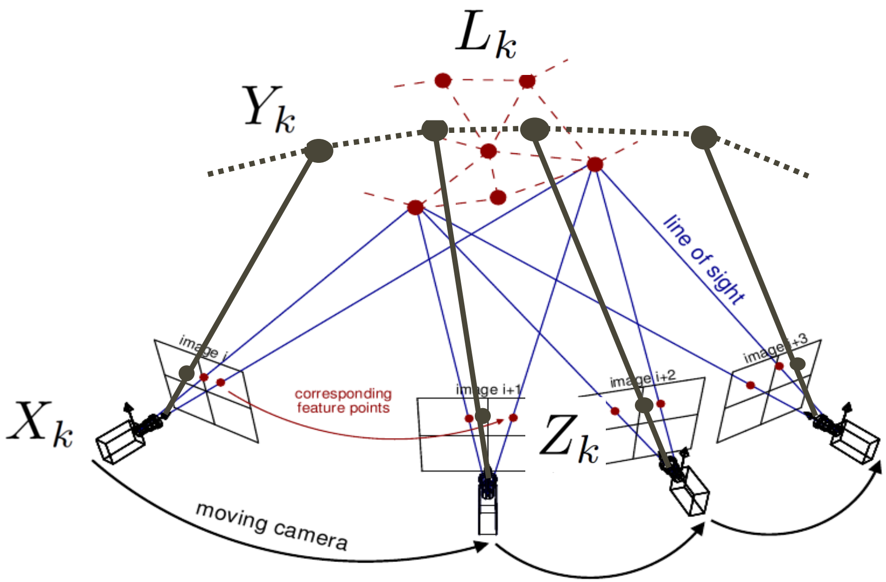
Measurement Model

Measurement Model : Pinhole Camera



- Defining the Projection Operator : $proj(x, l) \doteq K[R \quad t]l$ **[R.I. Hartley, 2004]**
- Observation Model : $z = proj\,(x, l) + v$ where $v \sim \mathcal{N}\,(0, \Sigma_v)$
- Measurement Likelihood : $$p(z|x,l) = \frac{1}{\sqrt{|2\pi\Sigma_v|}} \exp\left(-\frac{1}{2}\|z - proj(x,l)\|^2_{\Sigma_v}\right)$$
  actual (z) **predicted** ($proj(x,l)$)
  Re-projection error

Problem Formulation : BA and Target Tracking



- Joint probability distribution function (pdf)

$$P\left(X_k, Y_k, L_k | Z_k\right) \propto priors \cdot \prod_{i=1}^{k} \left(p\left(y_i | y_{i-1}\right) p\left(z_i^T | x_i, y_i\right) \prod_{j \in \mathcal{M}_i} p\left(z_i^j | x_i, l_j\right) \right)$$

Prior Information

Motion Model

Measurement Model

Motion Model : Constant Velocity

- Target state : $$y_k = \begin{bmatrix} y_k^T \\ \dot{y}_k^T \end{bmatrix}$$

- State Propagation : $$y_{k+1} = \Phi_k y_k + G_k w_k \quad \text{where} \quad w_k \sim \mathcal{N}(0, \Sigma_w)$$

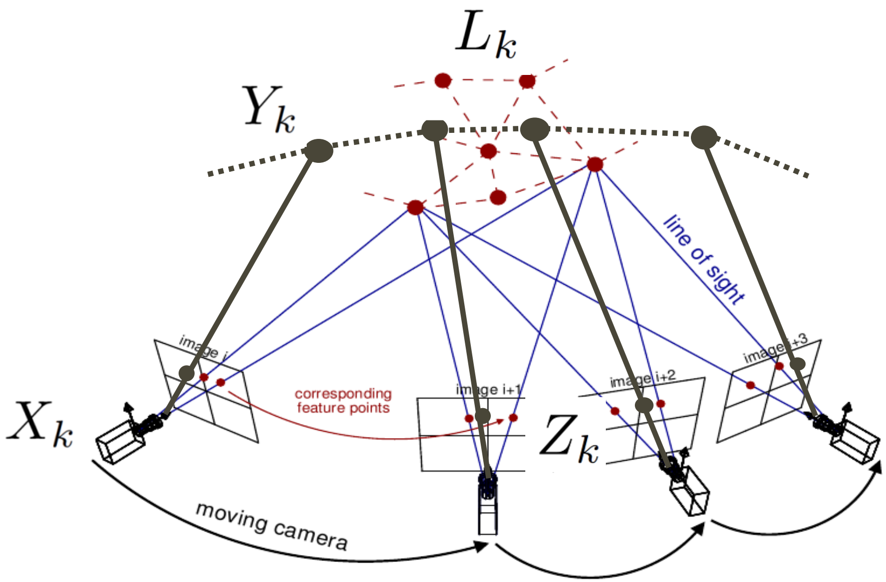
  Φ_k → **Transition Matrix** G_k → **Process noise Jacobian**

- Constant Velocity : $$\Phi_k = \begin{bmatrix} I_{3\times3} & \triangle t \\ 0 & I_{3\times3} \end{bmatrix} \in \mathbb{R}^{6\times6} \quad \text{and} \quad G_k = \begin{bmatrix} 0 \\ I_{3\times3} \end{bmatrix} \in \mathbb{R}^{6\times3}$$

- Probabilistic Representation : $$p(y_{k+1}|y_k) \doteq \frac{1}{\sqrt{|2\pi\Sigma_{mm}|}} \exp\left(-\frac{1}{2}\|y_{k+1} - \Phi_k y_k\|^2_{\Sigma_{mm}}\right)$$

  $$\Sigma_{mm} \doteq G^T \Sigma_w G$$

Problem Formulation : BA and Target Tracking



- Joint probability distribution function (pdf)

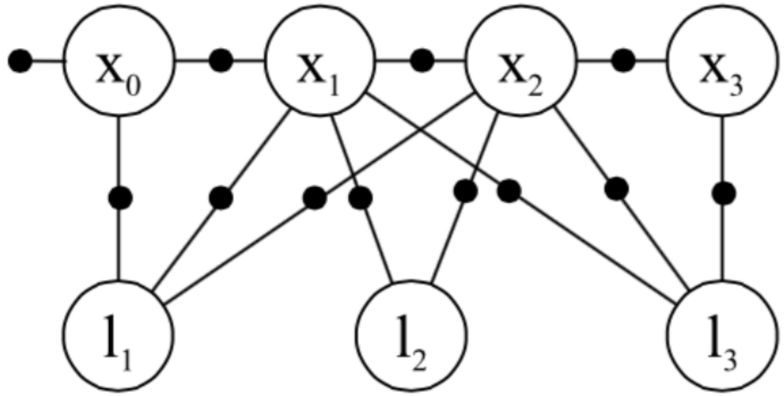
$$P\left(X_k, Y_k, L_k | Z_k\right) \propto priors \cdot \prod_{i=1}^{k}\left(p\left(y_i | y_{i-1}\right) p\left(z_i^T | x_i, y_i\right) \prod_{j \in \mathcal{M}_i} p\left(z_i^j | x_i, l_j\right)\right)$$

$$\propto \exp\left(-\frac{1}{2}\left\|y_i - \Phi_i y_{i-1}\right\|^2_{\Sigma_{mm}}\right) \quad \propto \exp\left(-\frac{1}{2}\left\|z_i^T - proj\left(x_i, y_i\right)\right\|^2_{\Sigma_v}\right) \quad \propto \exp\left(-\frac{1}{2}\left\|z_i^j - proj\left(x_i, l_j\right)\right\|^2_{\Sigma_v}\right)$$

- Pdf can also be represented by graphical models : **Factor Graph**

Factor Graph

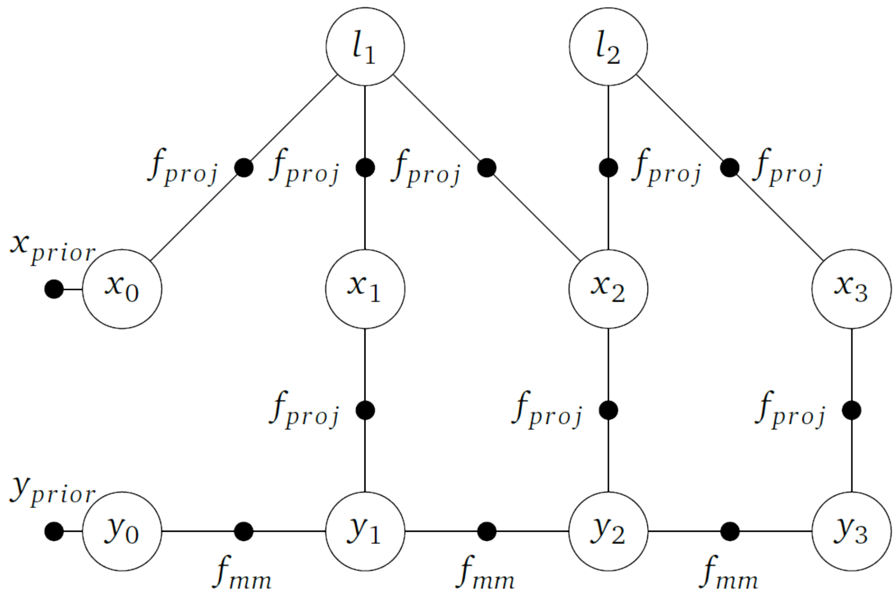
Describes a factorization of a joint pdf in terms of process and measurement models



- Vertices represent the variables
- Nodes represent constrains between variables, also known as **factors**

→ Allows for computationally efficient probabilistic inference

Factor Graph : BA and Target Tracking



- Joint pdf :

$$P(X_k, Y_k, L_k | Z_k) \propto p(x_0)\, p(y_0) \prod_{i=1}^{k} \left(f_{mm}(y_{i+1}, y_i)\, f_{proj}(x_i, y_i) \prod_{j \in \mathcal{M}_i} f_{proj}(x_i, l_j) \right)$$

$$\doteq \exp\left(-\frac{1}{2}\|y_i - \Phi_i y_{i-1}\|^2_{\Sigma_{mm}}\right) \qquad \doteq \exp\left(-\frac{1}{2}\left\|z_i^T - proj(x_i, y_i)\right\|^2_{\Sigma_v}\right) \qquad \doteq \exp\left(-\frac{1}{2}\left\|z_i^j - proj(x_i, l_j)\right\|^2_{\Sigma_v}\right)$$

- Maximum a posteriori (MAP) estimate : $X_k^*, Y_k^*, L_k^* = \underset{X_k, Y_k, L_k}{\arg\max}\, P(X_k, Y_k, L_k | Z_k)$

Involves reconstruction of the 3D structure

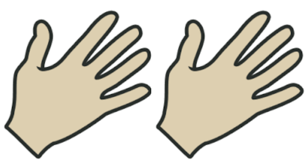
Computational Efficiency

- For BA and target tracking and with N frames observing M landmarks :

  → **12N + 3M** elements to optimize **(6N – Camera, 6N – Target, 3M - Landmarks)**

- Performed Incrementally as new surrounding features are observed

  → Increases the computational complexity of the problem



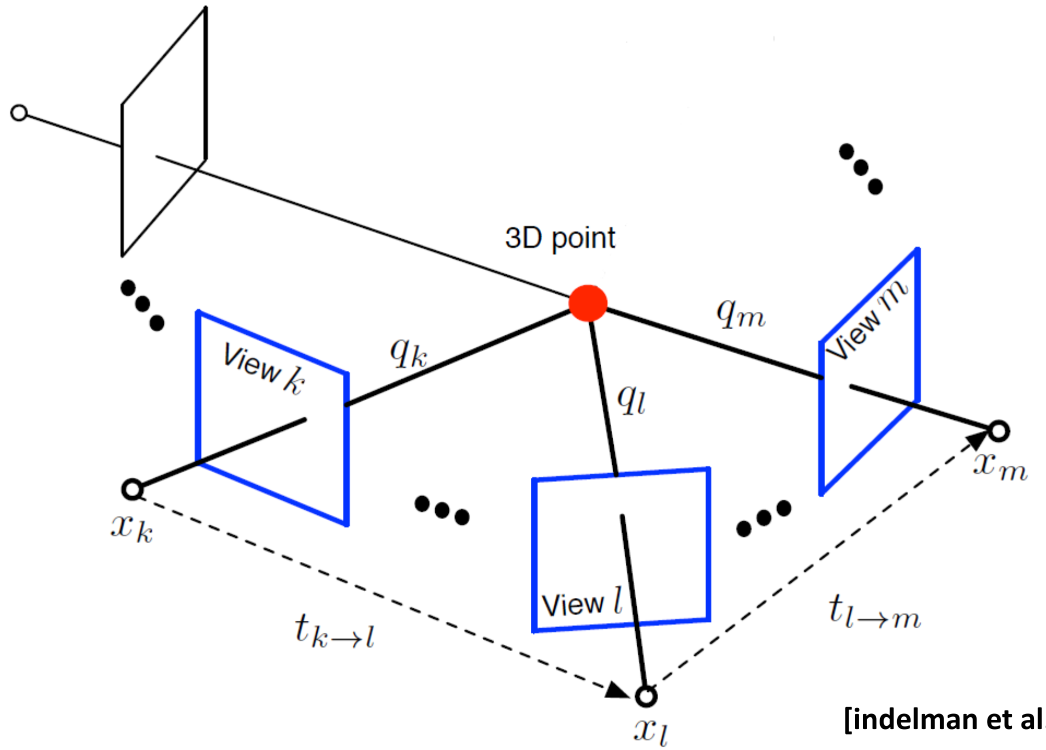
- On-line 3D structure reconstruction is of no interest : $P\left(X_k, Y_k | Z_k\right)$ **: 12N** elements

$$P\left(X_k, Y_k, L_k | Z_k\right) \xrightarrow{\textit{Marginalization}} P\left(X_k, Y_k | Z_k\right) = \int P\left(X_k, Y_k, L_k | Z_k\right) dL_k$$

Computationally Expensive Process !

→ **Light Bundle Adjustment (LBA)**

Incremental Light Bundle Adjustment (iLBA) – [Indelman et al., 2015]



[indelman et al., 2013]

- Intuition : 3 frames from which the same landmark is observed are related by geometrical constraints : **Multiview constraints**
- Allows to algebraically eliminate the landmarks from the optimization

→ Less Variables involved! (No need to calculate full BA first)

Incremental Light Bundle Adjustment (iLBA) – [Indelman et al., 2015]

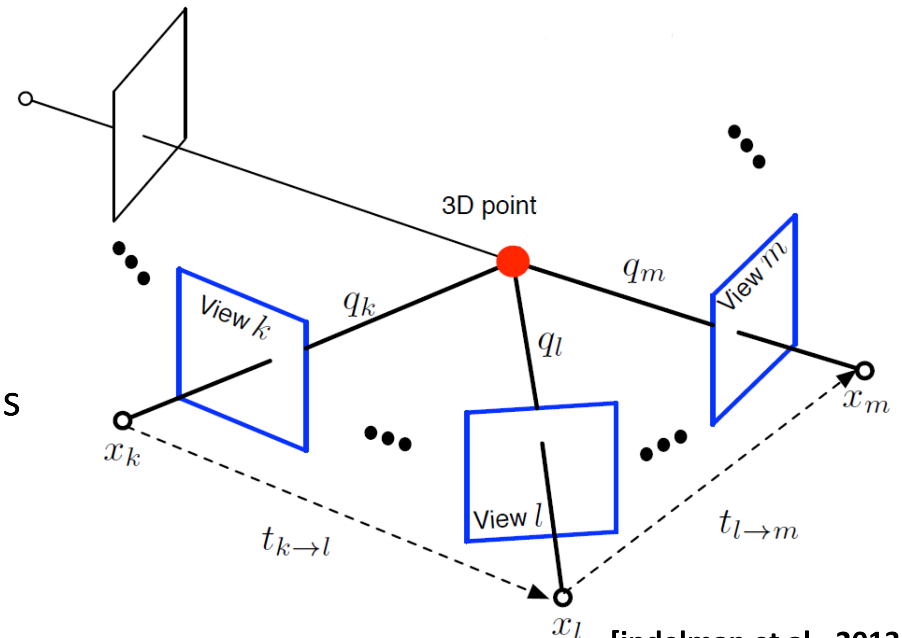
$$g_{2v}\left(x_k, x_l, z_k, z_l\right) = q_k \cdot \left(t_{k\to l} \times q_l\right)$$
$$g_{2v}\left(x_l, x_m, z_l, z_m\right) = q_l \cdot \left(t_{l\to m} \times q_m\right)$$

2 view constraints

$$g_{3v}\left(x_k, x_l, x_m, z_k, z_l, z_m\right) =$$
$$\left(q_l \times q_k\right) \cdot \left(q_m \times t_{l\to m}\right) - \left(q_k \times t_{k\to l}\right) \cdot \left(q_m \times q_l\right)$$

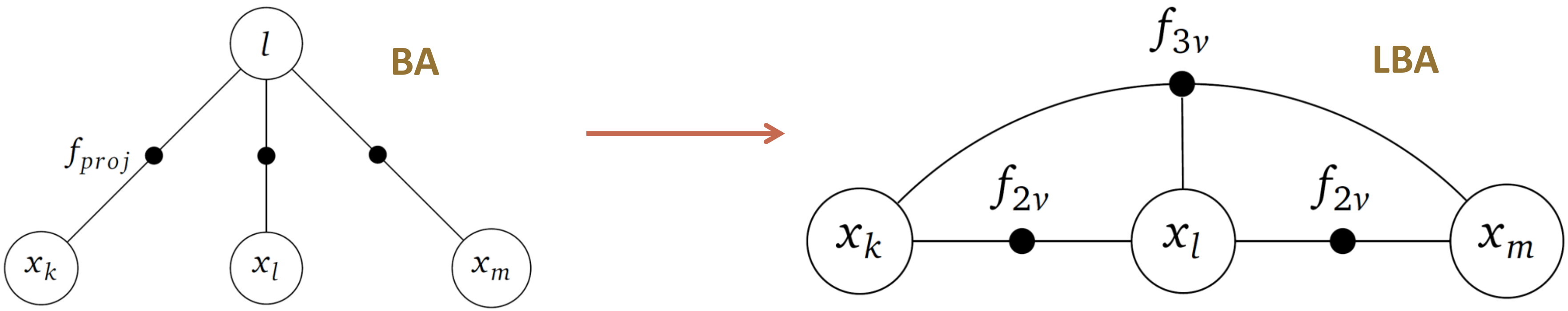
3 view constraint

- **2 view constraints** : epipolar geometry
- **3 view constraints** : relates between the scales of $t_{l\to m}$ and $t_{k\to l}$



[indelman et al., 2013]

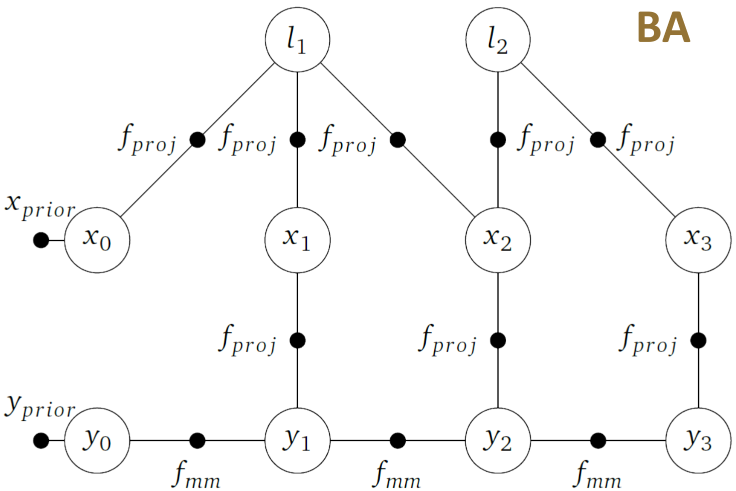
Incremental Light Bundle Adjustment (iLBA) – [Indelman et al., 2015]

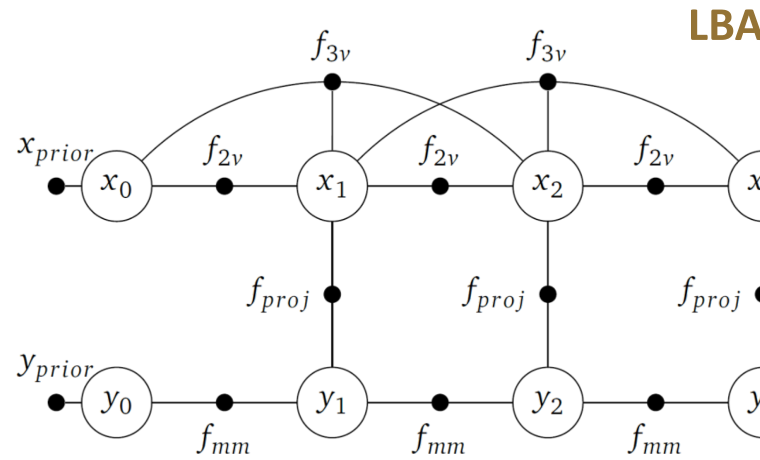


$$f_{2v}(x_k, x_l) \doteq \exp\left(-\frac{1}{2}\left\|g_{2v}(x_k, x_l, z_k, z_l)\right\|^2_{\Sigma_{2v}}\right)$$ → 2 view factor

$$f_{3v}(x_k, x_l, x_m) \doteq \exp\left(-\frac{1}{2}\left\|g_{3v}(x_k, x_l, x_m, z_k, z_l, z_m)\right\|^2_{\Sigma_{3v}}\right)$$ → 3 view factor

… With target tracking



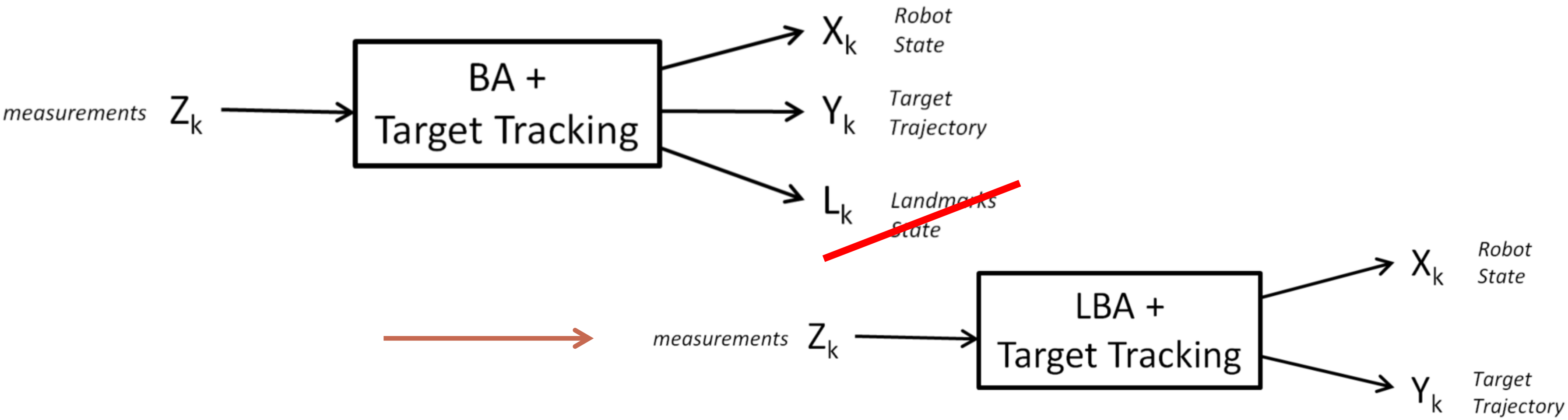


$$P\left(X_k, Y_k, L_k|Z_k\right) \propto priors \prod_{i=1}^{k}\left(f_{mm}\left(y_i, y_{i-1}\right) f_{proj}\left(x_i, y_i\right) \prod_{j\in\mathcal{M}_i} f_{proj}\left(x_i, l_j\right)\right)$$

$$P\left(X_k, Y_k|Z_k\right) \propto p\left(x_0\right) p\left(y_0\right) \prod_{i=1}^{k}\left(f_{mm}\left(y_i, y_{i-1}\right) f_{proj}\left(x_i, y_i\right) \prod_{j=1}^{N} f_{2v/3v}\left(X_j\right)\right)$$

The target is the only re-constructed 3D point !

Up till now



- Less variables to optimize : N frames, M landmarks **12N + 3M** → **12N**

- We solve : $X_k^*, Y_k^*, L_k^* = \arg\max_{X_k, Y_k, L_k} p\left(X_k, Y_k, L_k | Z_k\right)$ → $\boxed{X_k^*, Y_k^* = \arg\max_{X_k, Y_k} p\left(X_k, Y_k | Z_k\right)}$

- **How ?**

LBA and Target Tracking

- Recall : $P\left(X_k, Y_k | Z_k\right) \propto p\left(x_0\right) p\left(y_0\right) \prod_{i=1}^{k} \left(f_{mm}\left(y_i, y_{i-1}\right) f_{proj}\left(x_i, y_i\right) \prod_{j=1}^{N} f_{2v/3v}\left(X_j\right) \right)$

$\doteq \exp\left(-\frac{1}{2}\left\|y_i - \Phi_i y_{i-1}\right\|^2_{\Sigma_{mm}}\right)$ **Motion model**

$\doteq \exp\left(-\frac{1}{2}\left\|z_i^T - proj\left(x_i, y_i\right)\right\|^2_{\Sigma_v}\right)$ **Observation model**

$\doteq \exp\left(-\frac{1}{2}\left\|h_j(X_j, Z_j)\right\|^2_{\Sigma_j}\right)$ **2v/3v constraints (LBA)**

- Find the MAP : $X_k^*, Y_k^* = \arg\max_{X_k, Y_k} p\left(X_k, Y_k | Z_k\right)$

Log is monotonic (same max/min)

- Equivalent to minimizing :

$$J\left(X_k, Y_k\right) = \left\|x_0 - \hat{x}_0\right\|^2_{\Sigma_x} + \left\|y_0 - \hat{y}_0\right\|^2_{\Sigma_y} + \\ + \sum_{i=1}^{k} \left(\left\|y_{i+1} - \Phi_i y_i\right\|^2_{\Sigma_{mm}} + \left\|z_i^T - proj\left(x_i, y_i\right)\right\|^2_{\Sigma_v} + \sum_{j}^{N_h} \left\|h_j(X_j, Z_j)\right\|^2_{\Sigma_j} \right)$$

- Approaches : Gauss-Newton, Levenberg-Marquardt, …

Optimization

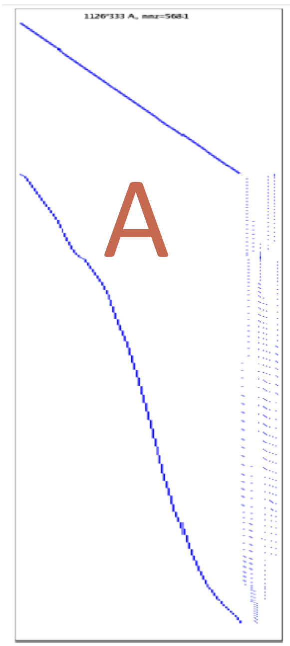
Pictures from [Dellaert et al., 2006]

Recipe for Gauss-Newton :

- Linearize cost function
- Re-arange RHS such that $J\left(\overline{\Theta}+\triangle\Theta\right)\approx\left\|A\triangle\Theta-b\right\|^{2}$
- Solve for $\triangle\Theta$
- Update linearization point $\overline{\Theta}+\triangle\Theta\rightarrow\overline{\Theta}$
- Repeat until convergence

Note :

- *A* contains the Jacobians of all the measurements with respect to the variables - A is large !
- *A* is sparse !



Jacobian matrix

Optimization

Pictures from [Dellaert et al., 2006]

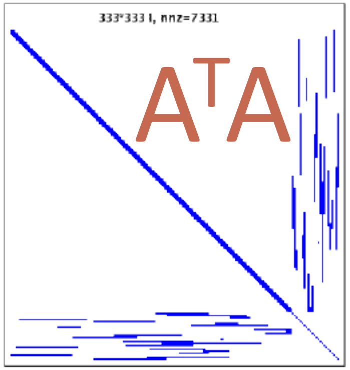
Recipe for Gauss-Newton :

- Linearize cost function
- Re-arange RHS such that $J\left(\overline{\Theta}+\triangle\Theta\right)\approx\left\|A\triangle\Theta-b\right\|^{2}$
- Solve for $\triangle\Theta$
- Update linearization point
- Repeat until convergence

Need to solve $A\triangle\Theta=b$ → $A^{T}A\triangle\Theta=A^{T}b$

$$\triangle\Theta=\left(A^{T}A\right)^{-1}A^{T}b$$

Expensive process !



Information matrix

Optimization

- Two issues :
  1. Naïve approach is **expensive**
  2. The **Entire process needs to be performed from scratch** each time a new variable/measurement is added to the problem

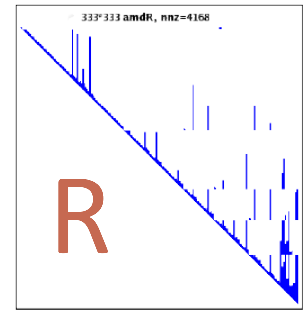
- Recently Developed Techniques :
  - Square Root SAM (Dellaert et al., 2006)
  - Incremental SAM - iSAM (Kaess et al., 2008)
  - iSAM2 (Kaess et al., 2012)

1. Exploits **sparsity** of the involved matrices to **simplify** $\triangle\Theta$ **recovery**
2. Uses **graphical models** to perform **Incremental optimization** : Calculations from previous steps can be reused

Incremental Smoothing and Mapping (iSAM)

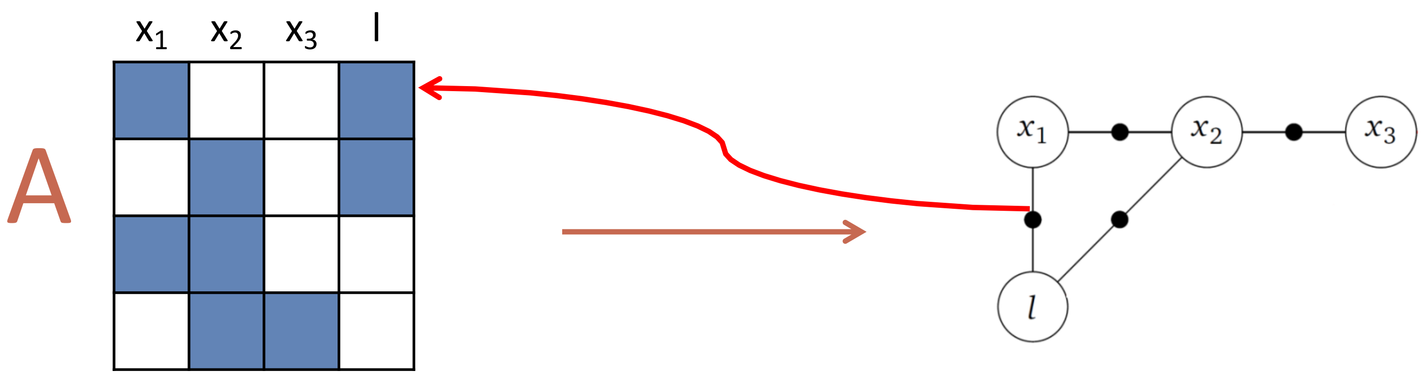
1. Exploiting matrix sparsity
    - Factorization : QR (A), Cholesky (A^TA)

QR : → $R\Delta\Theta = d$

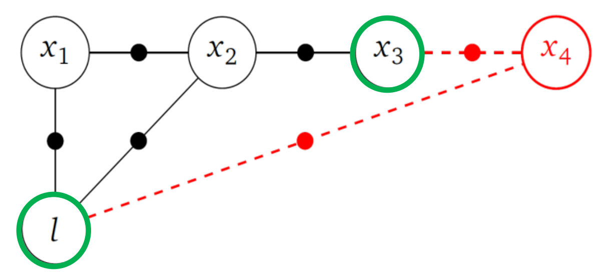


Square-root matrix

2. Using graphical models to allow for incremental optimization

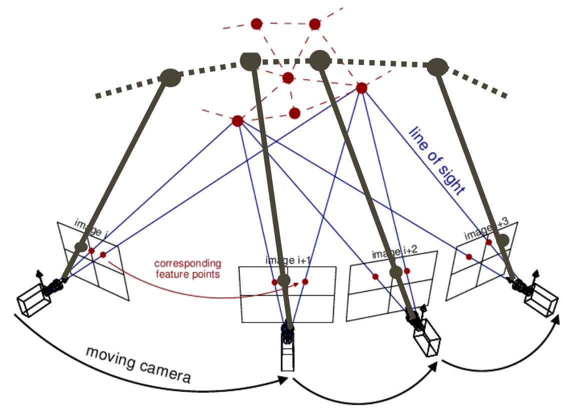


- Identify which variables are **involved in new factors**
- R matrix can be **updated**, not recalculated

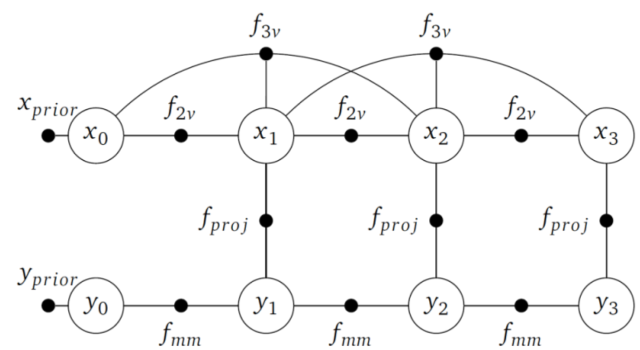


Up till now

- **One big optimization process** including camera and target states



- **Integrated target tracking into iLBA** framework :
  - Involves less variables (structure-less)
  - Performs incremental inference over graphical models

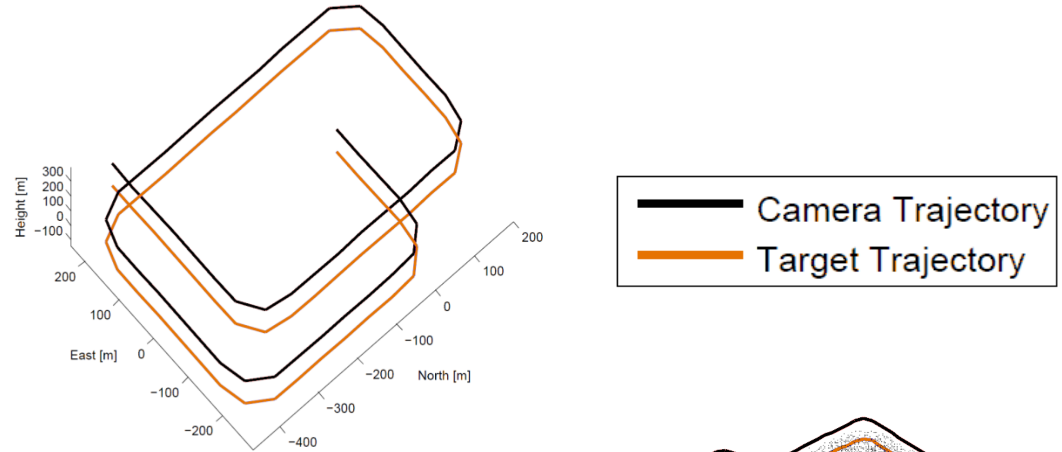


Let's see some results !

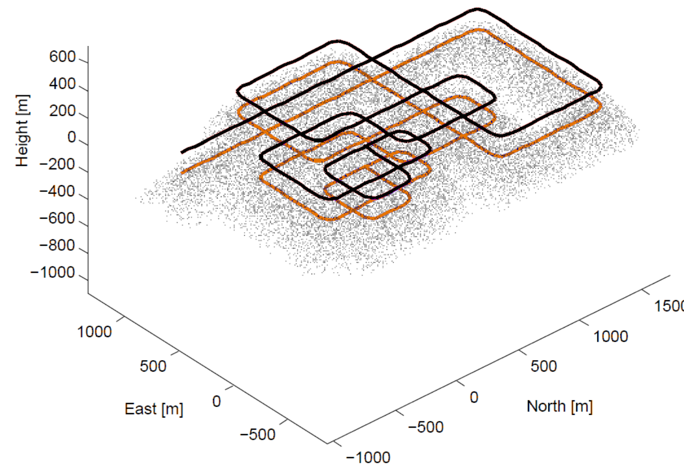
Scenario

Experiments and Simulations

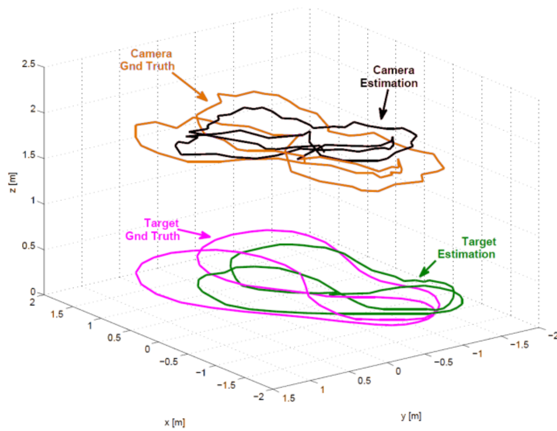
1. Statistical Simulation :
   - Short Scenario

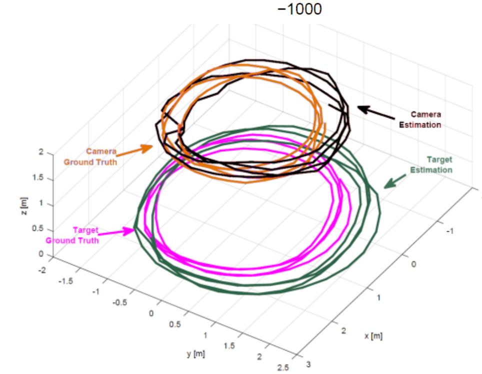


2. Case Study :
   - Large Scale Scenario



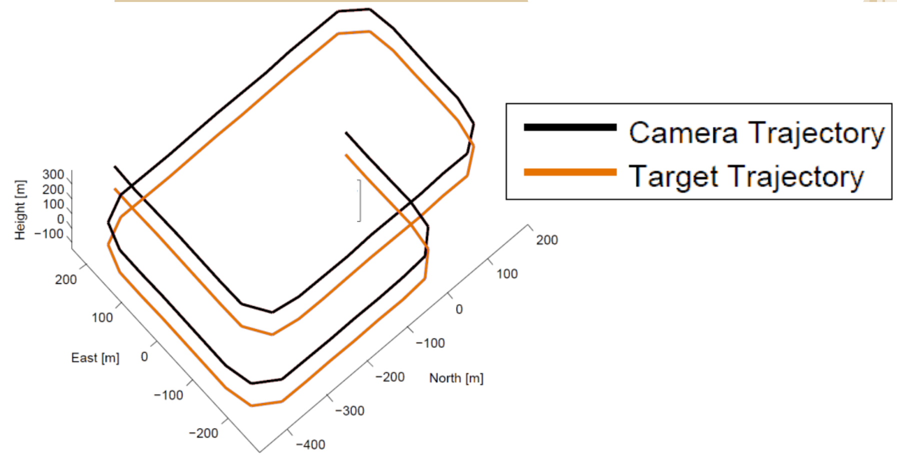
3. Real-Imagery Experiments

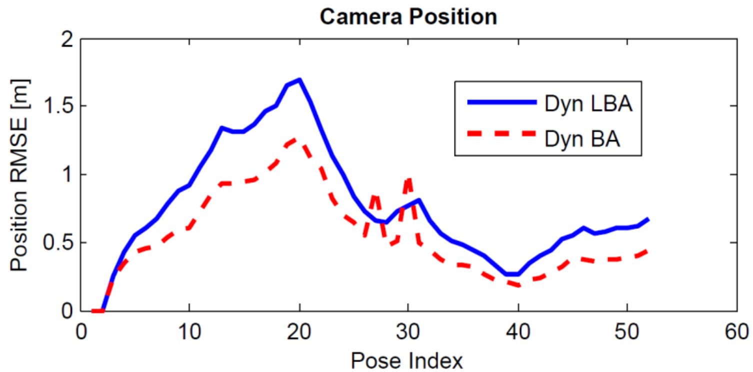


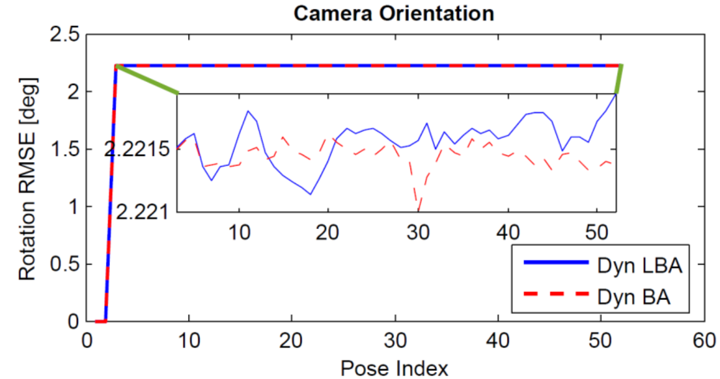


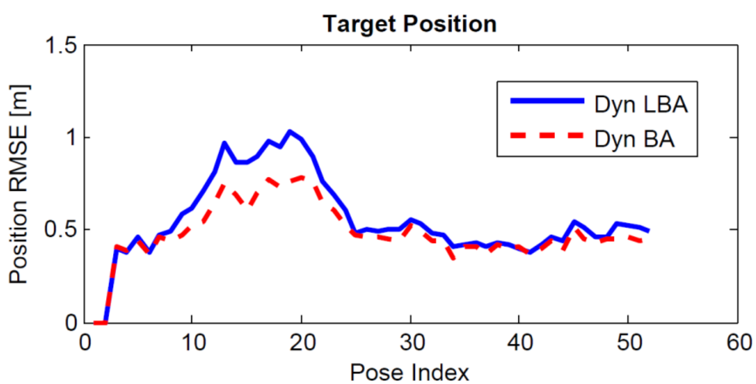
1. Statistical Simulation

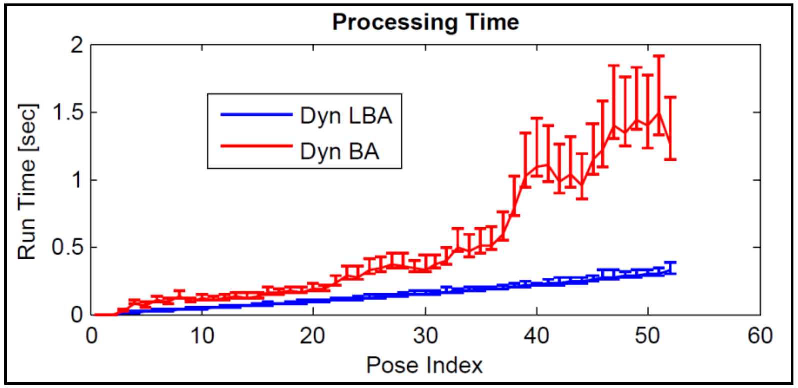
- 45 run Monte-Carlo study
- Short Scenario : 52 frames, 160 seconds
- 2 Loop Closures





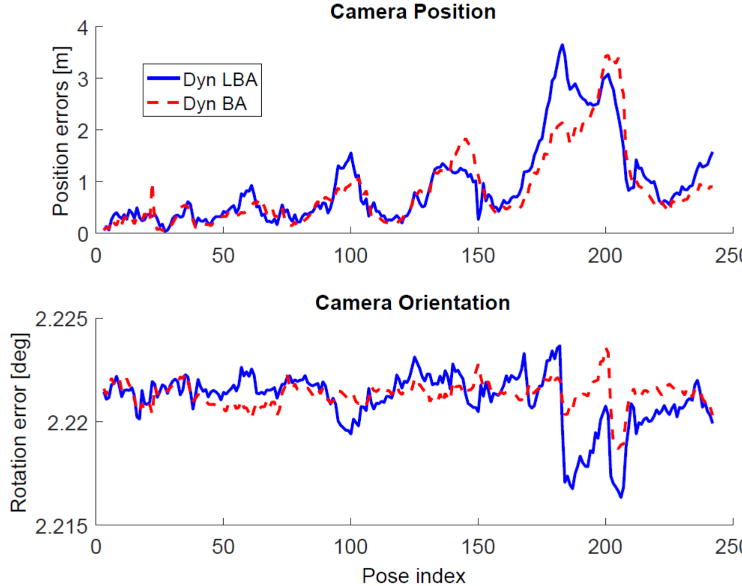


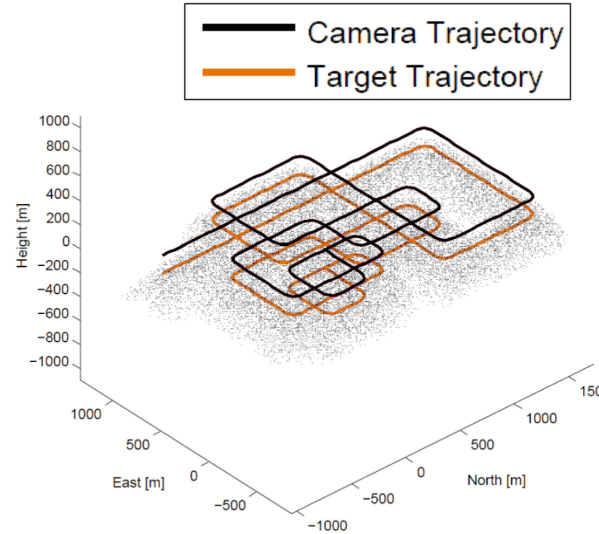


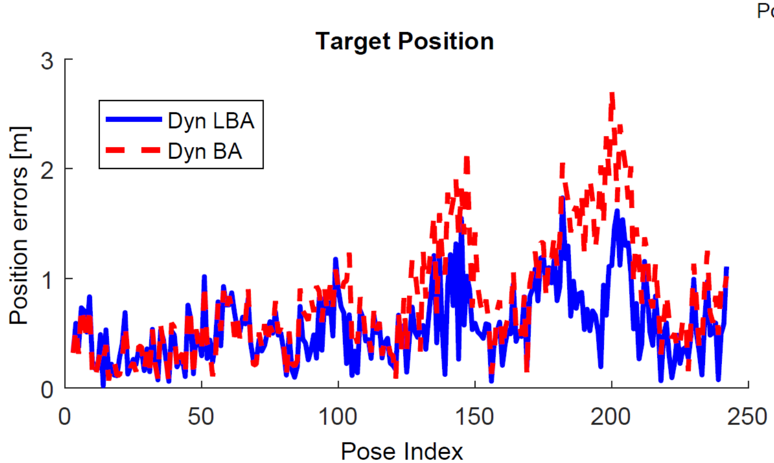


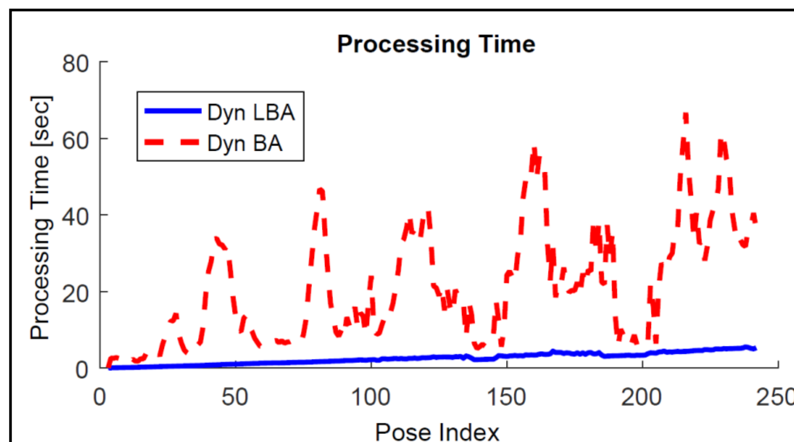
2. Large Scale Scenario Simulation

- Large scenario : 240 frames, 14.5 km
- ~25300 observed Landmarks, ~10 Loop Closures



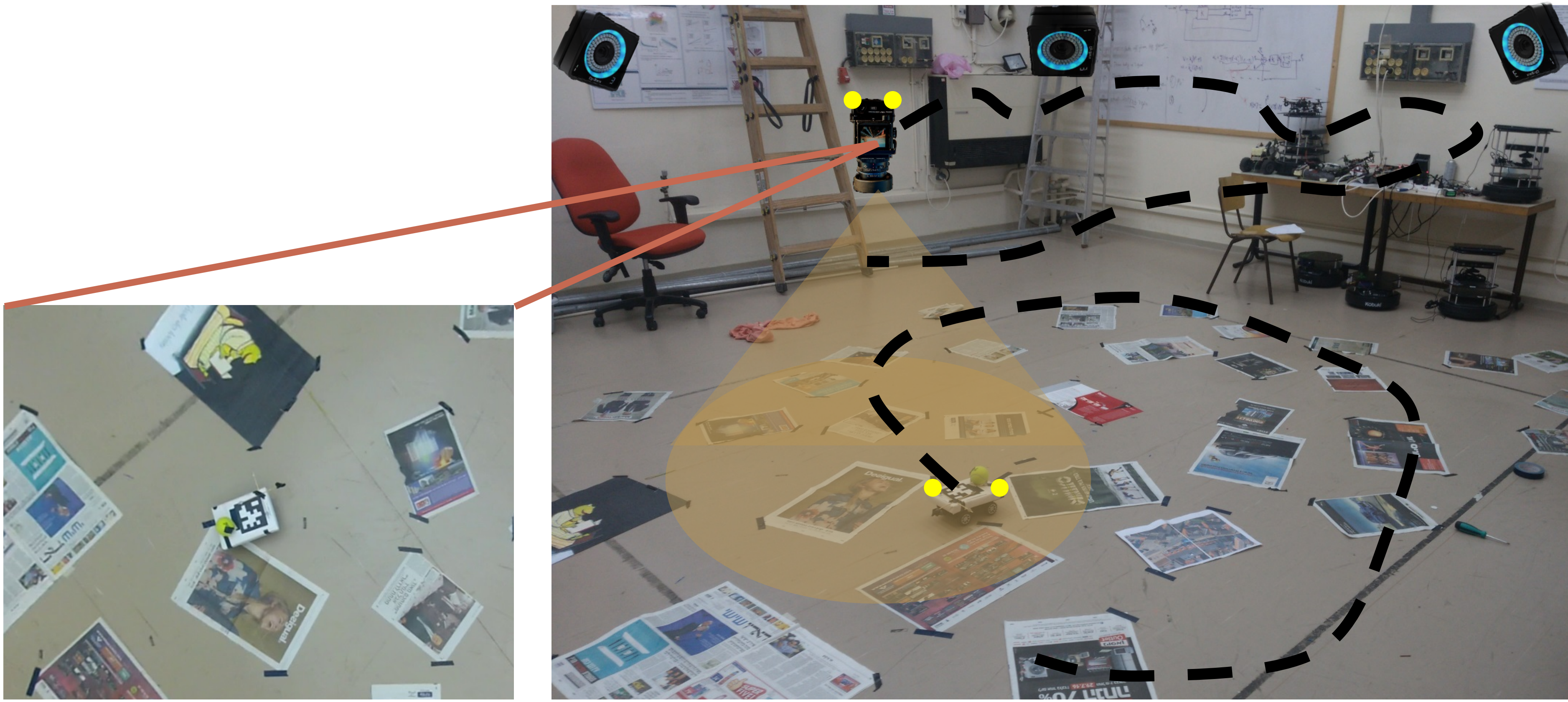






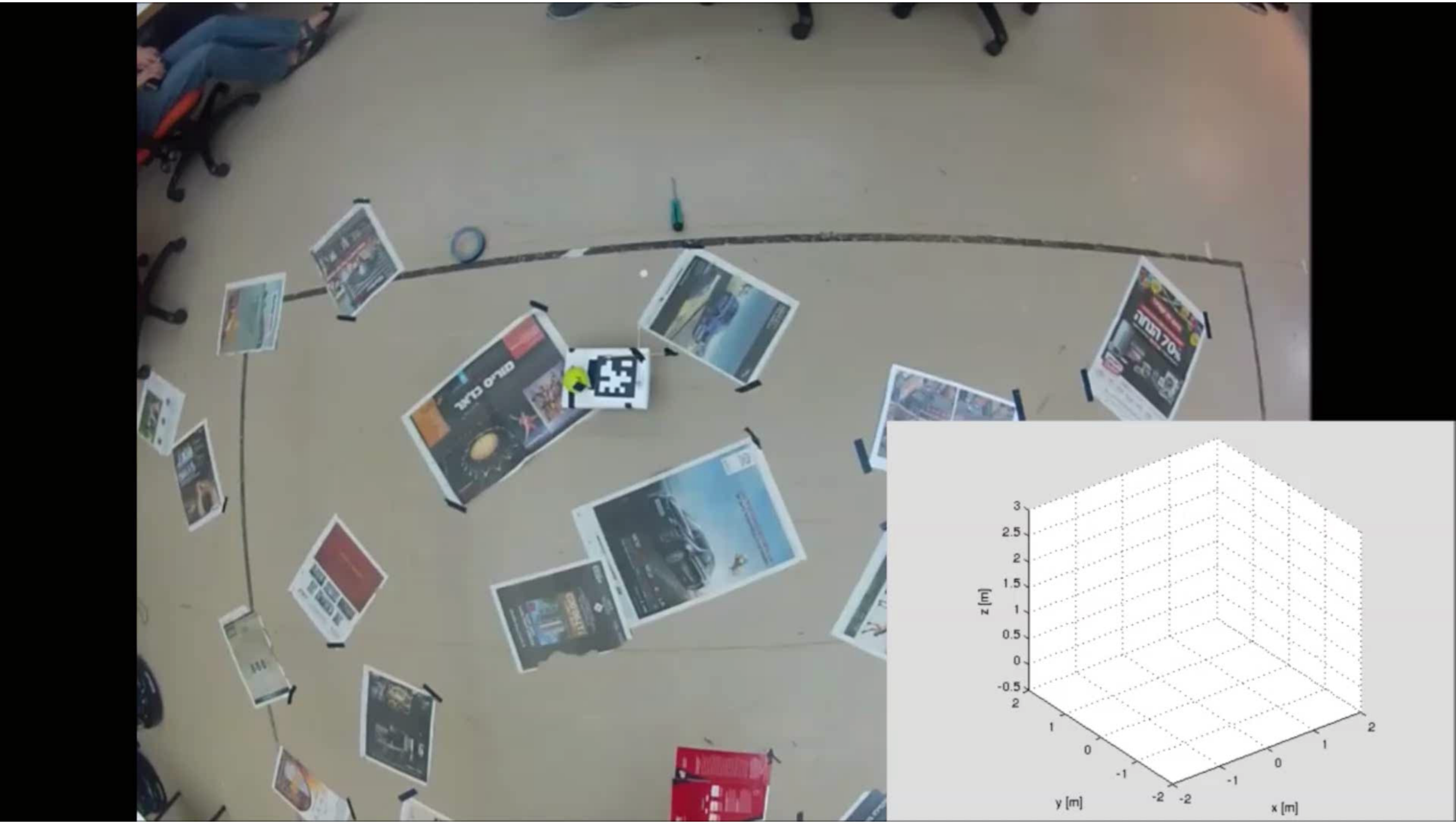
3. Real-Imagery Experiments

- Aerial Scenario : Downward facing camera observing a dynamic target on the ground
- Ground Truth from 6DoF optical tracking system



- Datasets publicly available at : http://vindelman.net.technion.ac.il/software/

3. Real-Imagery Experiments

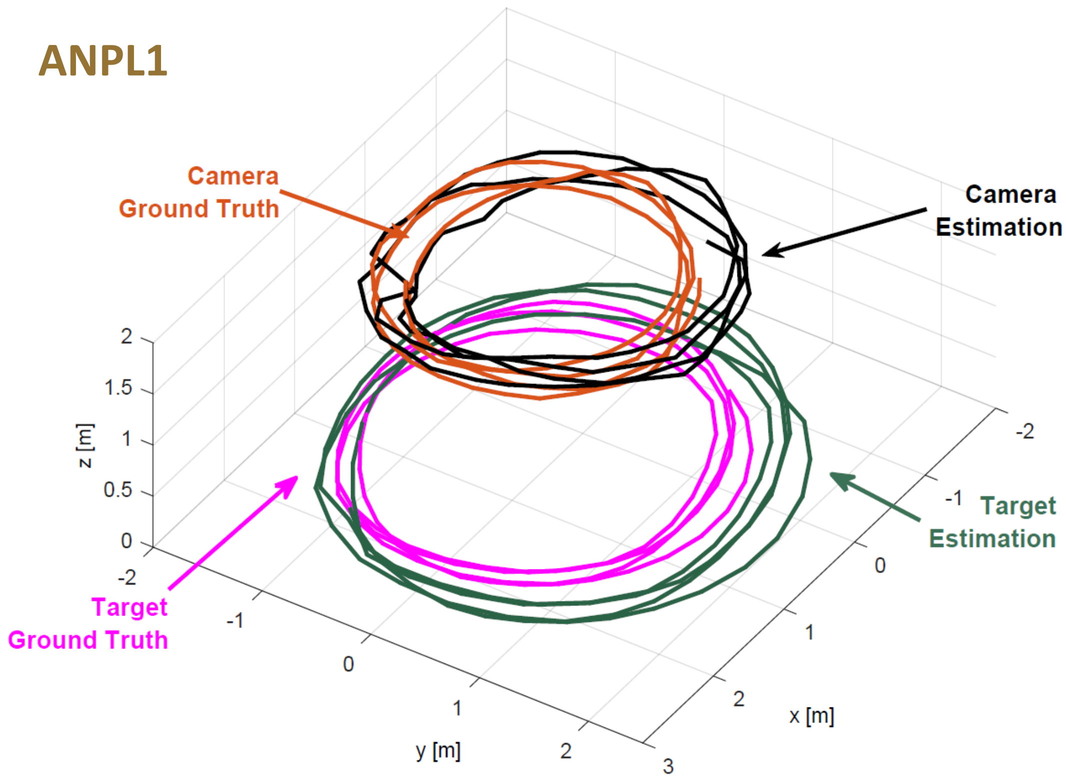


3. Real-Imagery Experiments

- 2 different datasets

	Camera Resolution [pix]	Frames	Duration [sec]	Land-marks	Obser-vations
ANPL1	1280 × 960	80	40	2439	31333
ANPL2	1920 × 1080	73	117	3366	25631

ANPL1



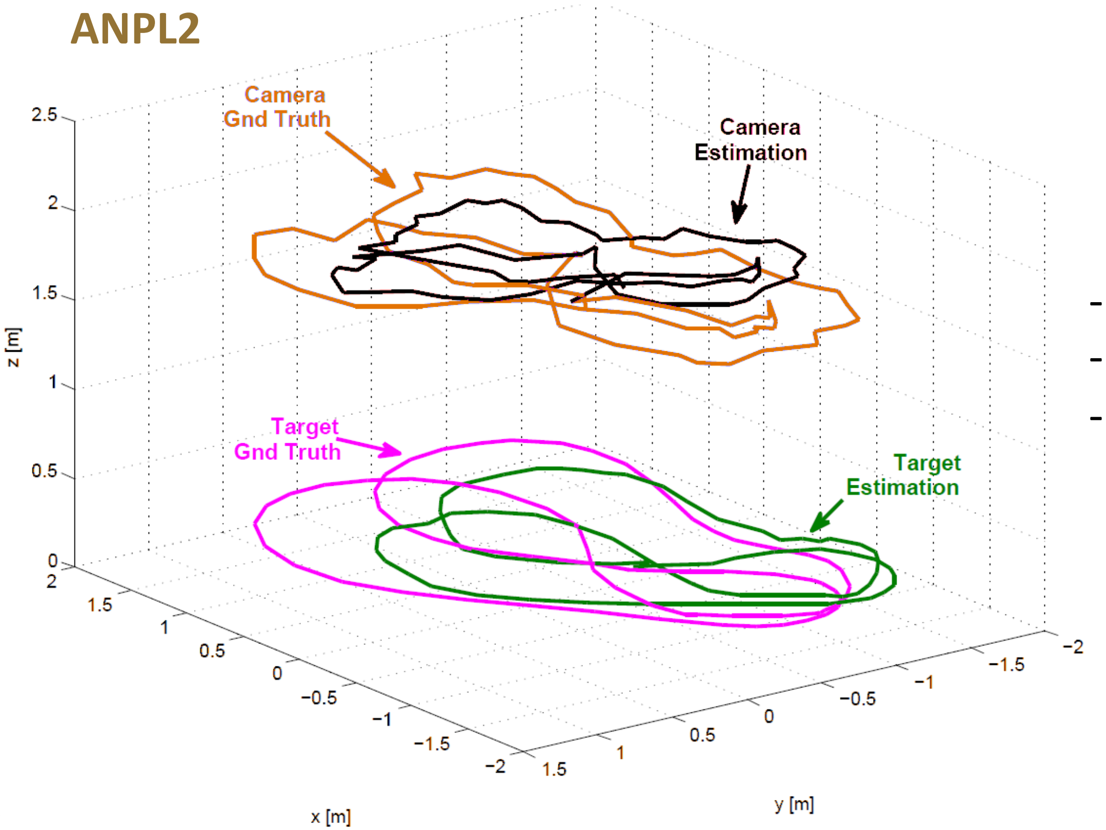
- Circular recurrent path
- Synchronized movements
- Frequent loop closer
- Target always in sight

3. Real-Imagery Experiments

- 2 different datasets

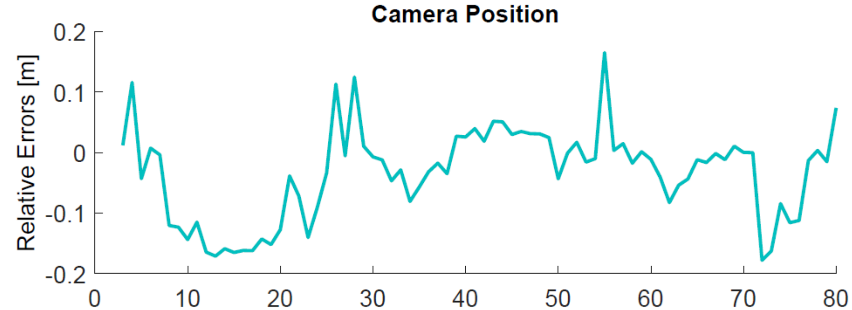
	Camera Resolution [pix]	Frames	Duration [sec]	Land-marks	Obser-vations
ANPL1	1280 × 960	80	40	2439	31333
ANPL2	1920×1080	73	117	3366	25631

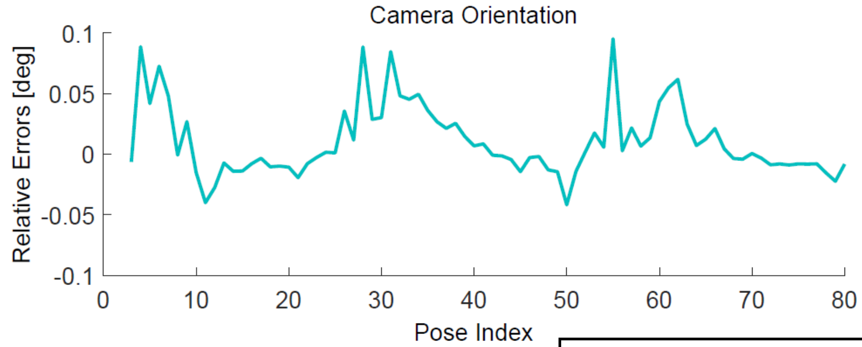
ANPL2

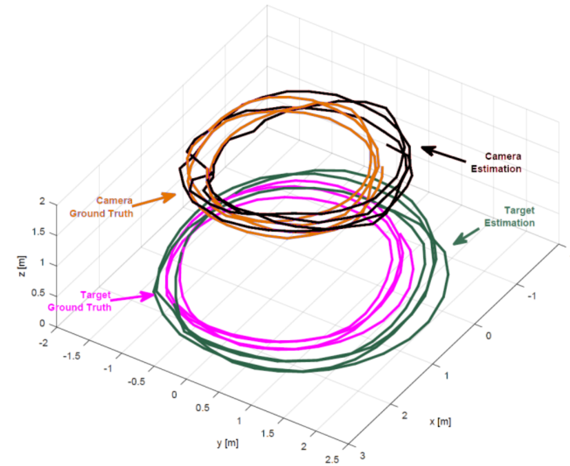


- More complex path
- Unsynchronized movements
- Target not always in sight

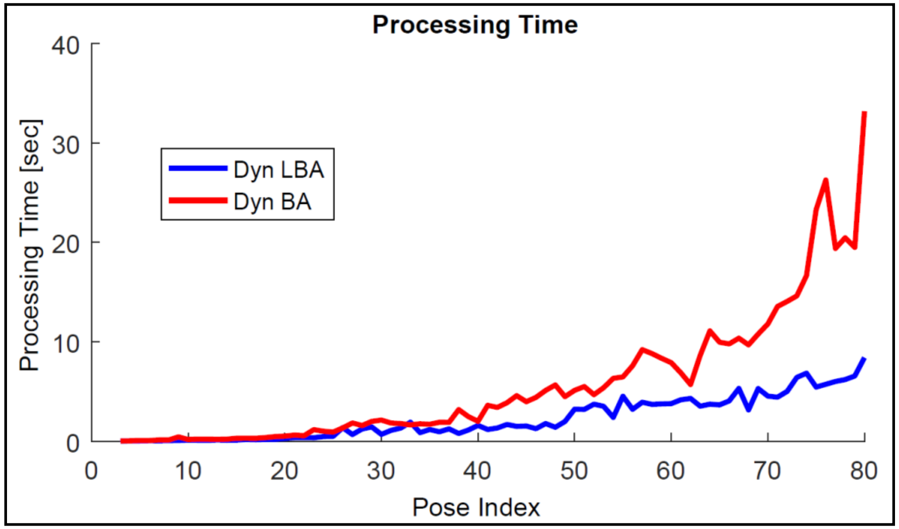
ANPL 1


Camera Position
Relative Errors [m]
0.2
0.1
0
-0.1
-0.2
0 10 20 30 40 50 60 70 80


Camera Orientation
Relative Errors [deg]
0.1
0.05
0
-0.05
-0.1
0 10 20 30 40 50 60 70 80
Pose Index

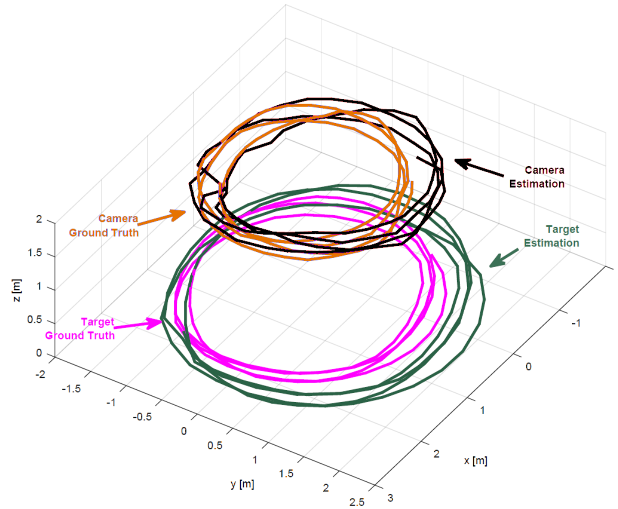

Camera Estimation
Camera Ground Truth
Target Estimation
Target Ground Truth
z [m]
y [m]
x [m]


Target Position
Relative Errors [m]
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
0 10 20 30 40 50 60 70 80
Pose Index

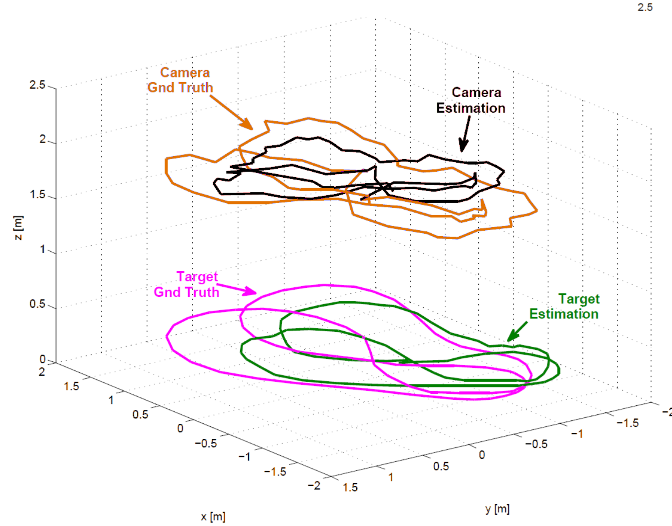

Processing Time
Processing Time [sec]
40
30
20
10
0
Dyn LBA
Dyn BA
0 10 20 30 40 50 60 70 80
Pose Index

Real-Imagery results summary

	Target Rel. Error [m]		Camera Rel. Error [m]	
	Mean	**Max**	**Mean**	**Max**
ANPL1	0.06	0.19	0.01	0.09
ANPL2	0.01	0.42	0.01	0.23



		Processing Time [sec]	
		Mean	**Total**
ANPL1	BA	5.6	447.8
	LBA	2.2	177.1
ANPL2	BA	3.1	222.9
	LBA	1.9	139.4



Conclusions / Future Work

Contributions

- An efficient method for vision-based ego-motion and target trajectory estimation
  → Target tracking problem is integrated into the iLBA framework

- Simulations/Tests show :
  - Considerable gain in computational efforts compared to BA
  - Similar levels of accuracy for both methods

- Publicly available datasets online

Future Work

- Problem extension to multi-robot / multi-target cases
  Challenge : data association

THANK YOU