Image-based Mobile Robot localization using Interval Methods

Ide Flore Kenmogne, Vincent Drevelle

Lagadic Group
Irisa/Inria Rennes Bretagne Atlantique

Swim 2016
19-22 June

UAV navigation in GPS denied environments

Motivation

Observation missions in GPS denied environments

Use of other sensors

Camera (most appropriate in UAVs case: weight & low cost)

Uncertainty quantification

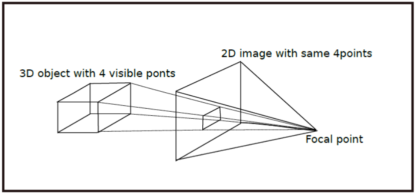
- For mapping acquired data
- For navigation safety

Outline

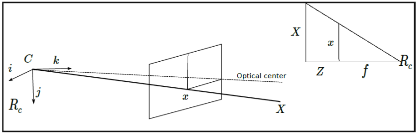
Image-based pose estimation

3D point projection in an image

Perspective projection



Which model for 3D point projection ?

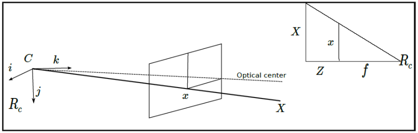


In R_c, the perspective projection of the point $\mathbf{X} = ({}^cX, {}^cY, {}^cZ)$ on the image point $\mathbf{x}=(x,y)$ can be given by :

$$\begin{cases} f\,{}^cX - {}^cZ\,x = 0 \\ f\,{}^cY - {}^cZ\,y = 0 \end{cases}$$

Pinhole Camera model

Let *(u,v)* be an image point coordinates in pixel and *(x,y)* its corresponding in meters.



Complete model

$$\begin{cases} u = u_0 + p_x \frac{^cX}{^cZ} + \delta' \, p_x \frac{^cX}{^cZ} \left(\frac{^cX^2}{^cZ^2} + \frac{^cY^2}{^cZ^2} \right) \\ v = v_0 + p_y \frac{^cY}{^cZ} + \delta' \, p_y \frac{^cY}{^cZ} \left(\frac{^cX^2}{^cZ^2} + \frac{^cY^2}{^cZ^2} \right) \end{cases}$$

Simplified model

$$\begin{cases} u = u_0 + p_x \frac{^cX}{^cZ} \\ v = v_0 + p_y \frac{^cY}{^cZ} \end{cases}$$

Where $p_x = \frac{f}{l_x}$, $p_y = \frac{f}{l_y}$ and $\delta = \delta' * f^2$. And $(u_0, v_0, p_x, p_y, \delta)$ represent the camera intrinsic parameters

Pinhole Camera model

Perspective model: Linear notation in homogeneous coordinates

$$\underbrace{\begin{pmatrix} u \\ v \\ 1 \end{pmatrix}}_{U} \approx \begin{pmatrix} p_x & 0 & u_0 \\ 0 & p_y & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} p_x & 0 & u_0 \\ 0 & p_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{K} \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\Pi} \underbrace{\begin{pmatrix} {}^cX \\ {}^cY \\ {}^cZ \\ 1 \end{pmatrix}}_{X}$$

with,

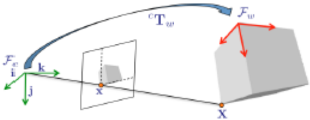
- K: the camera intrinsic parameters matrix
- Π: the perspective projection matrix

Pose Estimation: problem

- What is Pose Estimation ?
  - pose = position and orientation of an object (6DOF)
  - pose estimation = getting the pose of an object from a 2D image
- Given
  - calibrated camera : known intrinsic parameters
  - landmarks positions
  - corresponding points on the image-plane
- What is wanted ?
  - the camera pose

Pose Estimation: problem

Invert the projection model (re-projection) to find the appropriate transformation between camera and World frames.



Transformation ${}^{c}T_w$ between world frame F_w and camera frame F_c and perspective projection[1]

Determining the camera extrinsic parameters.

$$ {}^{c}X = {}^{c}T_w \, {}^{w}X \qquad \Rightarrow \qquad x = K \, \Pi \, {}^{c}T_w \, {}^{w}X $$

Where, ${}^{c}T_w$ is a homogeneous frame transformation matrix expressed as follow :

$$ {}^{c}T_w = \begin{pmatrix} {}^{c}R_w & {}^{c}t_w \\ 0_{3*1} & 1 \end{pmatrix} $$

Camera pose:
${}^{c}R_w$ (rotation matrix)
${}^{c}t_w$ (world frame position).

Pose Estimation: existing solutions[1]

- Algebraic algorithms (Good speed; poor noise filtering & numeric instabilities)
  - Linear 4/N-Point Algorithms
- Optimization (Iterative) Algorithms (numerically stable; dependence on initial guess)
  - Levenberg-Marquardt
- Hybrid Algorithms (numeric stability, speed, efficient handling of noise)
  - POSIT

Point pose estimation methods

Bounded error pose estimation using Interval analysis

Bounded error measurements

- The problem
  - Pixel measurements are subject to errors
  - Complex structured environments ⇒ landmarks positions in the world frame not well known
- Wanted
  - **propagate the uncertainty of the measurements in order to quantify pose estimate uncertainty**
- We assume
  - Bounded errors on Landmarks & Image-features
- Error representation (Intervals)
  - Let x and ${}^{w}X$ be the measurements vectors
  - each x and ${}^{w}X$ component is an *interval vector*, such that $x \in [x]$and ${}^{w}X \in [{}^{w}X]$
  - where the *intervals* represent the **uncertainty**

Interval Analysis as a tool

We can define problem as a Constraint satisfaction problem (CSP) H using:

- ${}^cR_w(\phi, \theta, \psi)$ and ${}^c_!t_w(X, Y, Z)$ as **variables**
- $x_i = K\ \Pi\ {}^cT_w\ {}^wXi$ applied to each measurement correspondence ($i = 1 : N$, with N the number of observed landmarks) as **constraints**
- Additional *geometric constraints* wrt. some UAV parameters limitations, characteristics and movements feasibility

Let $q = (X, Y, Z, \phi, \theta, \psi)$, the robot pose such that ${}^cR_w\ ,^c t_w$ are function of q, i.e.. ${}^cT_w\ =\ {}^cT_w(q)$.
Our CSP can be formulated as :

$$H\ \ : (\ x = K\ \Pi\ {}^cT_w(q)\ {}^wX; q \in [q], x \in [x],^wX \in [{}^wX]\ )$$

The *solution set* of H is defined as:

$$S_q = \{q \in [q]\,|\,\exists\,(x,^wX)\ \in\ [x] \times [{}^wX]\,,\ x\ =\ K\,\Pi\,{}^cT_w\,{}^wX\}$$

Contractor programming

To Contract H means replacing the pose $[q]$ by a smaller domain $[q^{'}]$ such that the solution set S_q remains unchanged.

- Contractor programming

  Consist in considering the algorithm behind H as a function (**Ctc**) by abstracting it from its underlying constraints

$$\textbf{Ctc} \ : \ IR^n \longrightarrow IR^n \ such \ that \ C([q]) \subseteq [q]$$

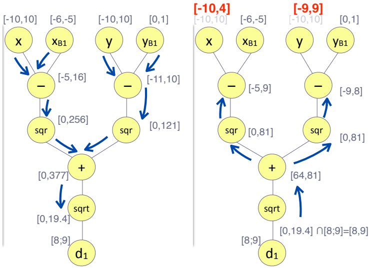
  Where,

- C is a Contractor which is an operator that can be used to contract H
- Union, Intersection, Composition

Contractors

HC4 Contractor

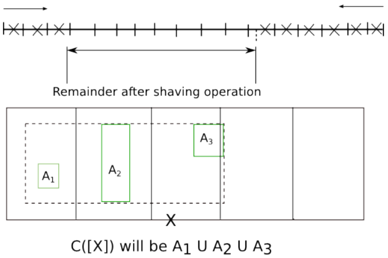
Based on Interval Constraint propagation



Example with :

$$d = \sqrt{(x - x_B)^2 + (y - y_B)^2}$$

Shaving



- 3B Consistency using HC4
- 4B Consistency using 3B
- NB Consistency ...

UAV Pose tracking

Evolution model

Taking in account the fact that a UAV is a dynamic system that has embedded sensors.The kinematic evolution model is as follow:

$$\dot{q} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} {}^{c}R_{w} & 0_{3\times3} \\ 0_{3\times3} & M \end{pmatrix} \begin{pmatrix} V \\ W \end{pmatrix}$$

With,

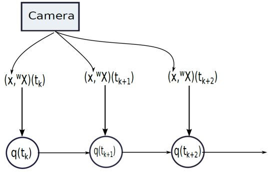
- $V = (u, v, w)^T$: linear body frame velocities
- $W = (p, q, r)^T$: angular velocities
- M is the matrix corresponding to the rotation matrix derivative in case of small angles variations

Pose prediction

- The robot measures its body frame velocities with an approximative estimate of the uncertainty using a gyro and an optical flow sensor.
- $\Rightarrow$ Bounded error velocities

$$[q]\,(t_{k+1}) = [q]\,(t_k) + \Delta_t \begin{bmatrix} {}^cR_w([q](t_k)) & 0_{3\times 3} \\ 0_{3\times 3} & M([q](t_k)) \end{bmatrix} \begin{pmatrix} [V] \\ [W] \end{pmatrix} (t_k)$$

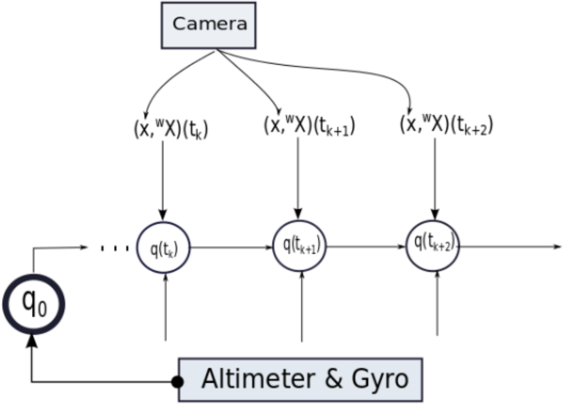
So, instead of doing a snapshot estimation, we use this interval evolution model to predict the pose at the next time-stamp.



Sensors data

- Given
  - Altitude (Z): a Barometric Altimeter
    No GPS $\Rightarrow$ No info about horizontal position (X,Y)
  - Attitude
    - good pitch and roll estimates (from inertial measurement unit)
    - larger uncertainty on yaw (due to magnetic perturbations)
- How do we make use of them ?
  - To initialize the pose domain
  - In case of empty domains,use proprioceptive data instead of camera

Method : Predictor/Corrector estimator



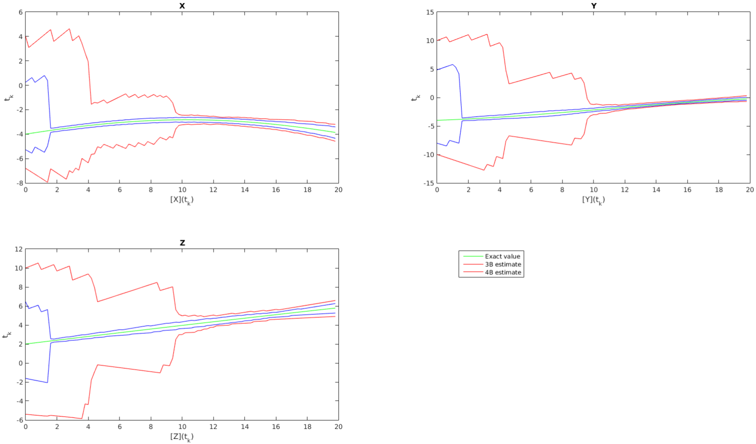
Simulation results

Simulation Environment

- Simulate the evolution of a UAV
  - Complete pose generation
  - From an arbitrary set of N Virtual World point
  - Reconstruct from reprojection the corresponding image point
- Uncertainty
  - image points $\approx 1px$
  - World points $\approx 1m$

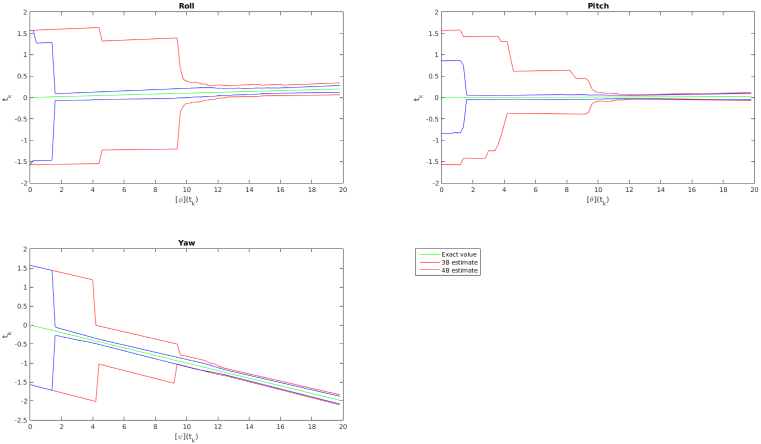
Simulation results : 6Dof camera only

Estimated position



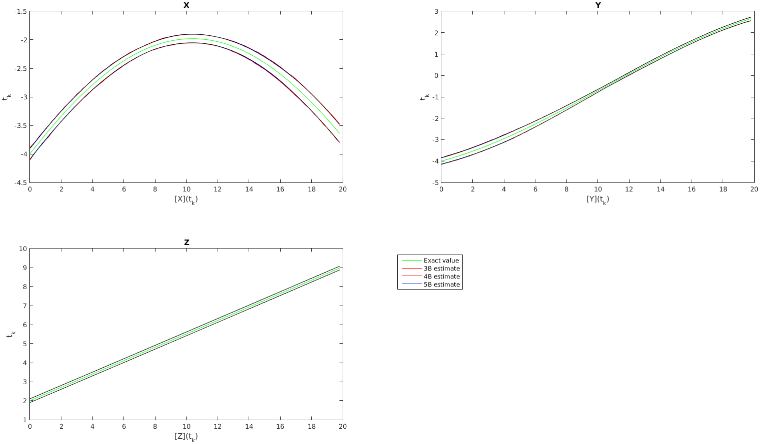
Simulation results

Estimated attitude



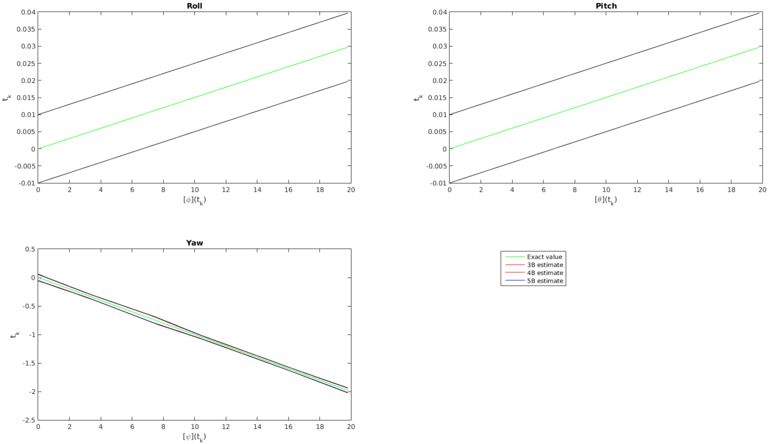
Simulation results Camera + Barometer + IMU

Estimated position



Simulation results Camera + Barometer + IMU

Estimated attitude



Simulation results : Mean Computation time/epoch

Camera only

	Mean(s)	Max(s)
3B	0,26	0,59
4B	6,37	14,40
5B	137,88	223,34

Camera + Barometer + IMU

	Mean(s)	Max(s)
3B	0,1584	0,25s
4B	0,28788s	0,388
5B	52,7664	0,7384

Conclusion and Research directions

- Conclusion
  - Camera pose estimation using 3B contractor compatible with real time application
  - Data fusion (Camera + Barometer + IMU) provides tighter bounds and improves computation time
- Future work
  - Real data experiment (ongoing)
  - Precondition the system to be align with the axis for better contractions
  - Guaranteed numerical integration
  - Outlier identification and rejection
  - Multi-robots cooperative localization

Equations

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & sin(\phi)tan(\theta) & cos(\phi)tan(\theta) \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi)sec(\theta) & cos(\phi)sec(\theta) \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

Main References

[1] E. Marchand, H. Uchiyama and F. , Pose estimation for augmented reality: a hands-on survey, *IEEE Trans. on Visualization and Computer Graphics*, To appear, 2016.

[2] B. Telle, M. -J. Aldon and N. Ramdani, Guaranteed 3D visual sensing based on interval analysis, *In proceedings of Intelligent Robots and Systems(IROS 2003)*, 2003.

[3] V. Drevelle and P. Bonnifait, Localization Confidence Domains via Set Inversion on Short-Term Trajectory, *IEEE Trans. on Robotics* 29(5):1244-1256,2013..

Thanks for listening !