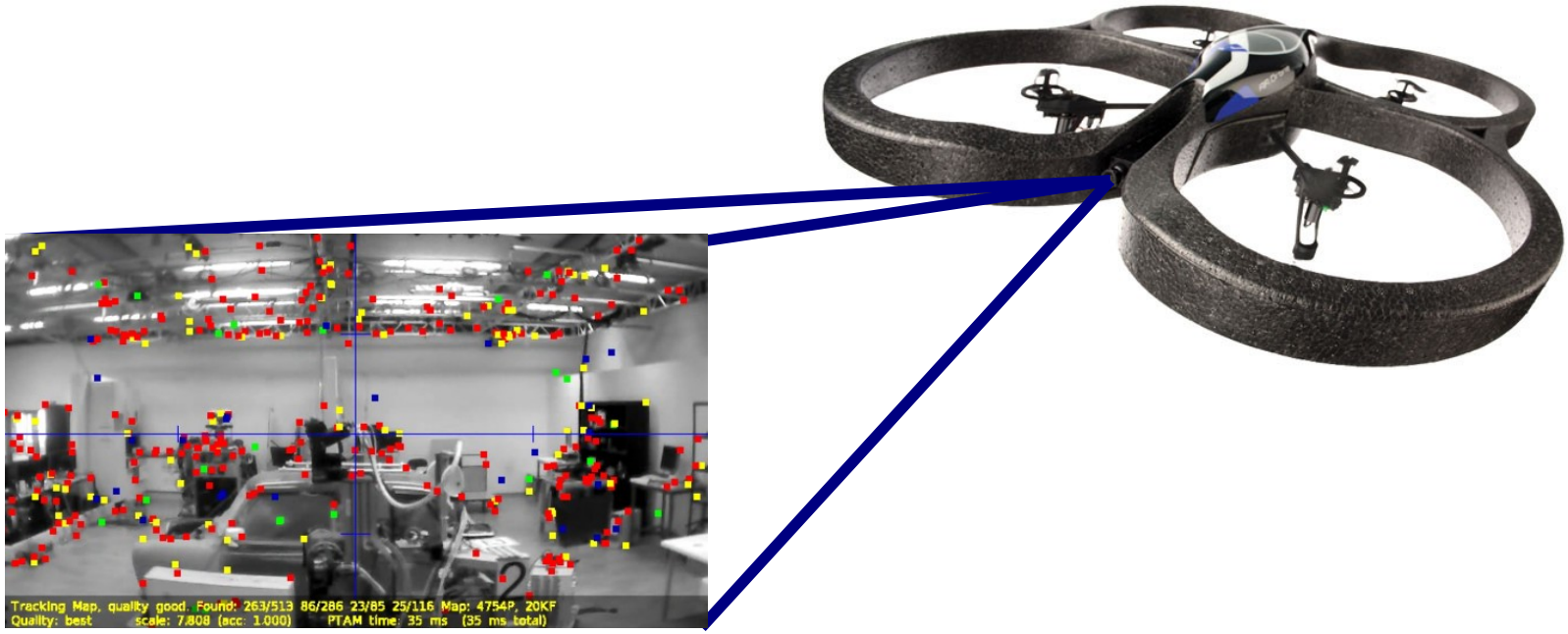
Autonomous Navigation of Small-Scale Quadrocopters

Jakob Engel

VisNav invited talk, 25.06.2013



How to make the AR.Drone fly more stable?

1. time delays
2. marker → PTAM

1. Time Delays

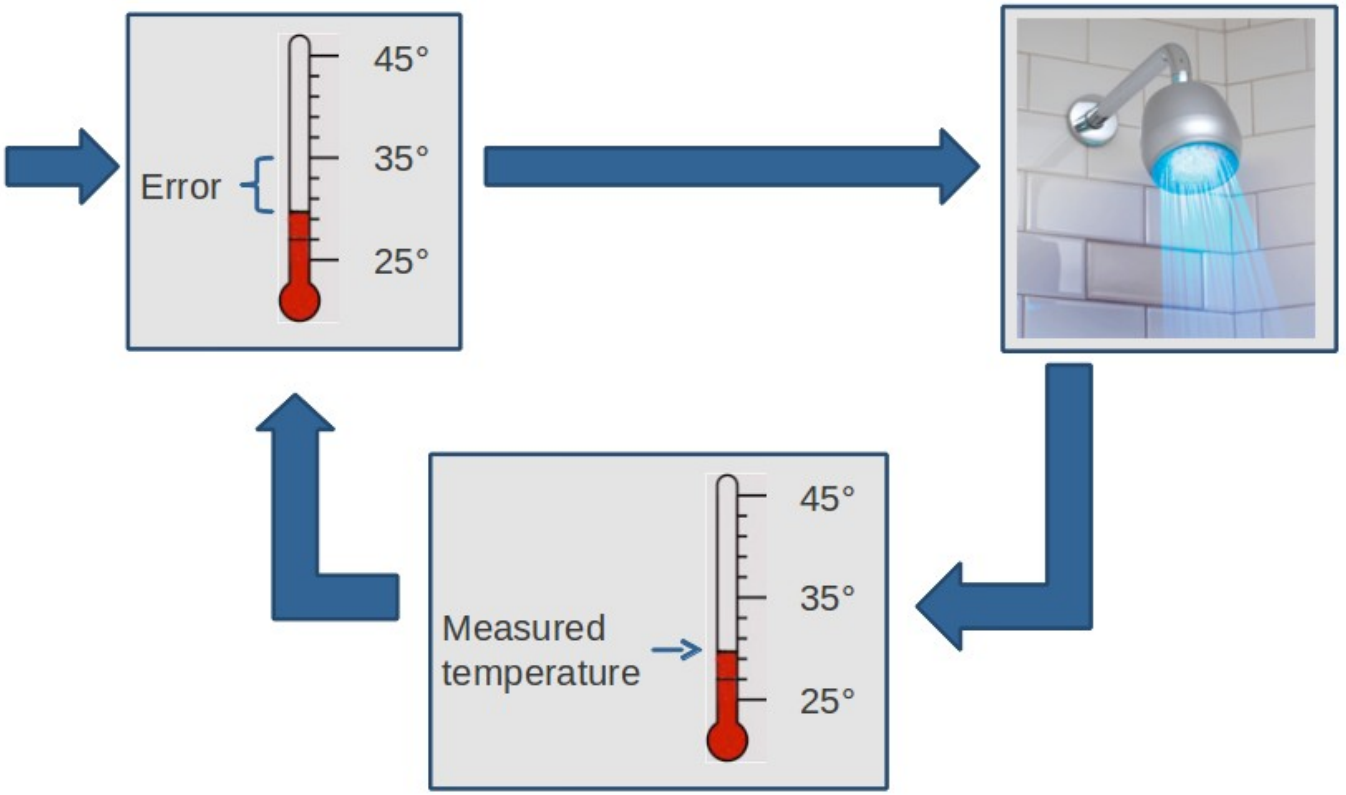
capture frame → send to PC → compute → send control

takes 150ms - 250ms

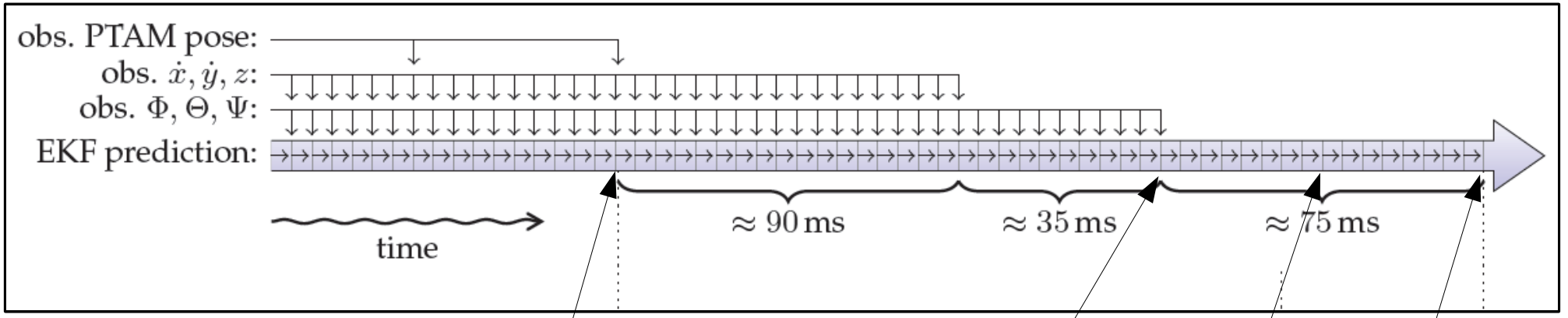
1. Time Delays

capture frame → send to PC → compute → send control

takes 150ms - 250ms



Solution: Explicitly model delays



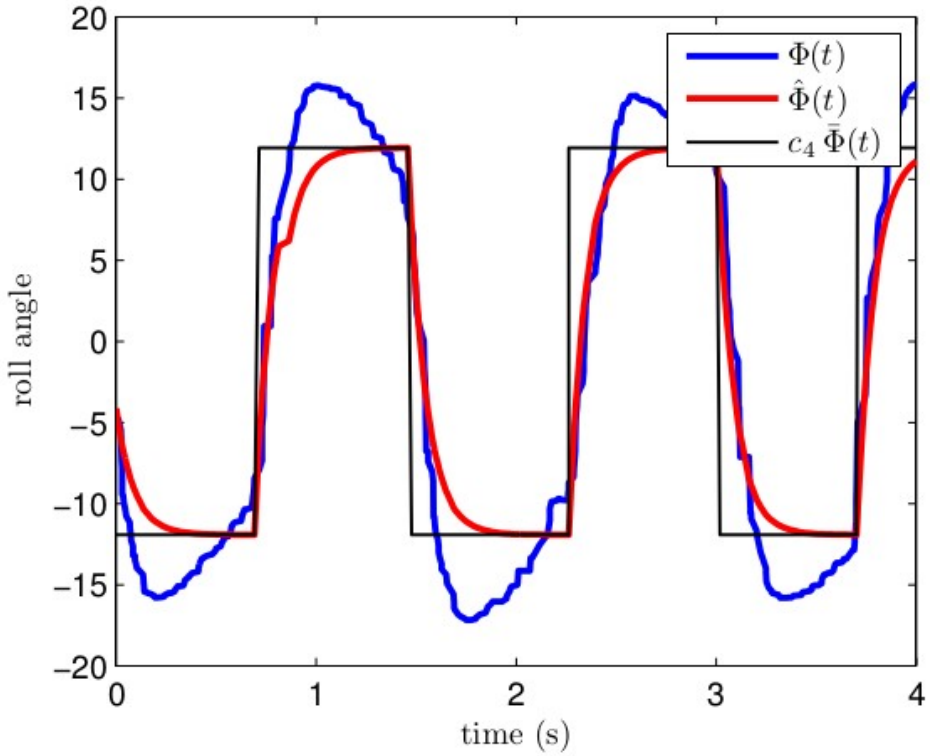
last visual observation

last IMU observation

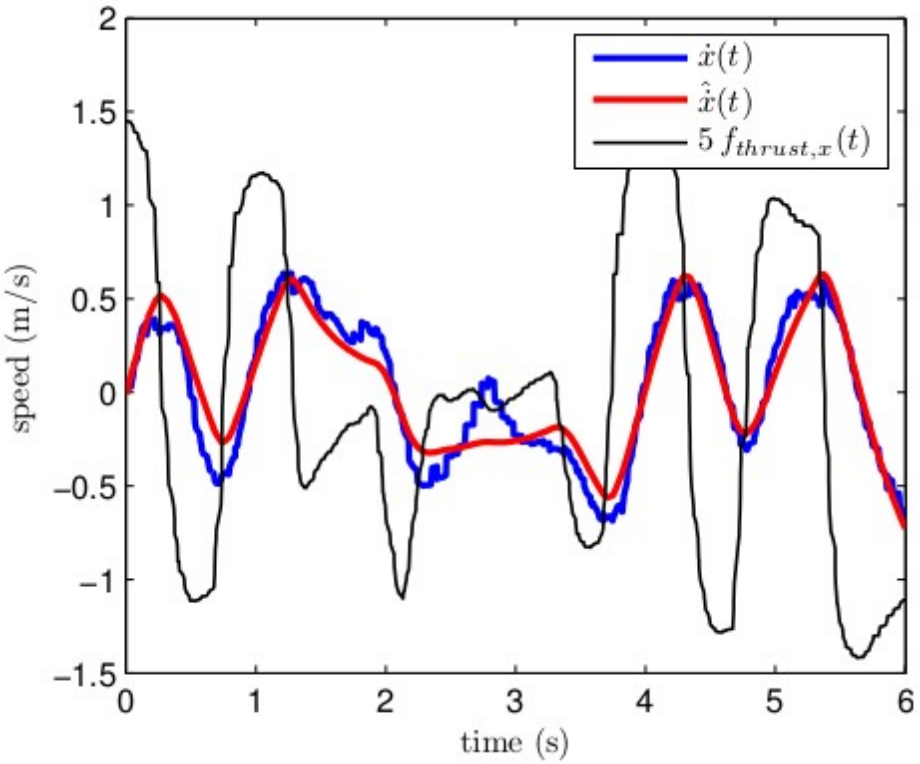
now

drone will receive control

Solution: Model control & dynamics



control → attitude



attitude → velocity

→ enables fully “blind” prediction

Full EKF

$$\begin{pmatrix} x_{t+1} \\ y_{t+1} \\ z_{t+1} \\ \dot{x}_{t+1} \\ \dot{y}_{t+1} \\ \dot{z}_{t+1} \\ \Phi_{t+1} \\ \Theta_{t+1} \\ \Psi_{t+1} \\ \dot{\Psi}_{t+1} \end{pmatrix} \leftarrow \begin{pmatrix} x_t \\ y_t \\ z_t \\ \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \\ \Phi_t \\ \Theta_t \\ \Psi_t \\ \dot{\Psi}_t \end{pmatrix} + \delta_t \begin{pmatrix} \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \\ \ddot{x}(\mathbf{x}_t) \\ \ddot{y}(\mathbf{x}_t) \\ \ddot{z}(\mathbf{x}_t, \mathbf{u}_t) \\ \dot{\Phi}(\mathbf{x}_t, \mathbf{u}_t) \\ \dot{\Theta}(\mathbf{x}_t, \mathbf{u}_t) \\ \dot{\Psi}_t \\ \ddot{\Psi}(\mathbf{x}_t, \mathbf{u}_t) \end{pmatrix}$$

$$\dot{\Phi}(\mathbf{x}_t, \mathbf{u}_t) = c_3 \bar{\Phi}_t - c_4 \Phi_t$$
$$\dot{\Theta}(\mathbf{x}_t, \mathbf{u}_t) = c_3 \bar{\Theta}_t - c_4 \Theta_t$$
$$\ddot{\Psi}(\mathbf{x}_t, \mathbf{u}_t) = c_5 \bar{\dot{\Psi}}_t - c_6 \dot{\Psi}_t$$
$$\ddot{z}(\mathbf{x}_t, \mathbf{u}_t) = c_7 \bar{\dot{z}}_t - c_8 \dot{z}_t$$
$$\ddot{x}(\mathbf{x}_t) = c_1 (\cos\Psi_t \sin\Phi_t \cos\Theta_t - \sin\Psi_t \sin\Theta_t) - c_2 \dot{x}_t$$
$$\ddot{y}(\mathbf{x}_t) = c_1 (-\sin\Psi_t \sin\Phi_t \cos\Theta_t - \cos\Psi_t \sin\Theta_t) - c_2 \dot{y}_t$$

EKF Prediction

$$h_{\text{PTAM}}(\mathbf{x}) := (x, y, z, \Phi, \Theta, \Psi)^T \in \mathbb{R}^6$$
$$\mathbf{z}_{\text{PTAM}} := \log(\mathbf{E}_{\mathcal{DC}} \mathbf{E}_{\mathcal{C}}) \in \mathbb{R}^6$$

PTAM Observation

$$h_{\text{IMU}}(\mathbf{x}) := \begin{pmatrix} \cos(\Psi)\dot{x} - \sin(\Psi)\dot{y} \\ \sin(\Psi)\dot{x} + \cos(\Psi)\dot{y} \\ z \\ \Phi \\ \Theta \\ \Psi \end{pmatrix} \in \mathbb{R}^6$$

$$\mathbf{z}_{\text{IMU}} := \begin{pmatrix} \dot{x}_d \\ \dot{y}_d \\ z(t-\delta_t) + h(t) - h(t-\delta_t) \\ \hat{\Phi} \\ \hat{\Theta} \\ \Psi(t-\delta_t) + \hat{\Psi}(t) - \hat{\Psi}(t-\delta_t) \end{pmatrix} \in \mathbb{R}^6$$

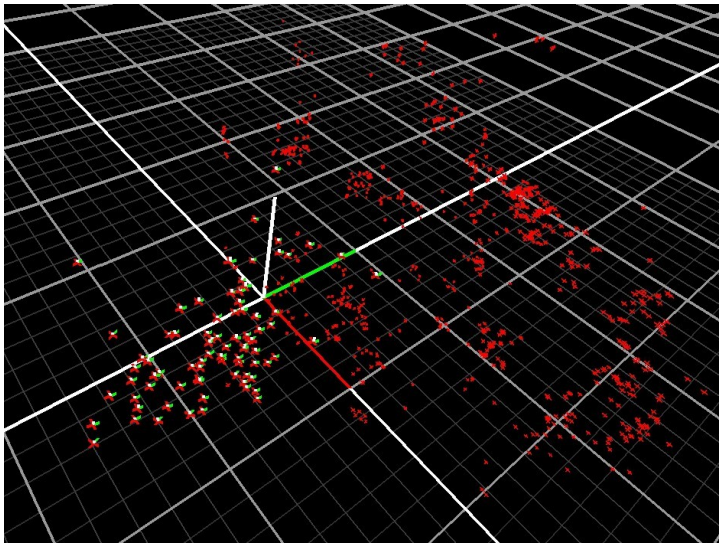
IMU, altimeter, velocity Observation

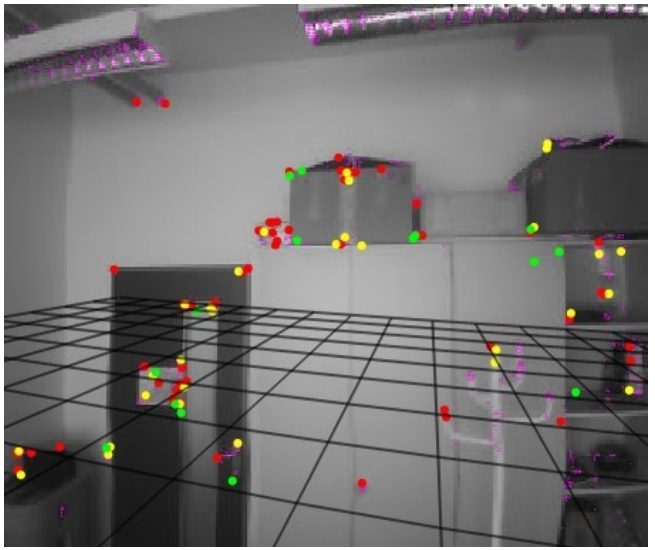
2. PTAM

Parallel Tracking and Mapping [Murray '07]:

→ keyframe based, monocular SLAM system

→ open-source





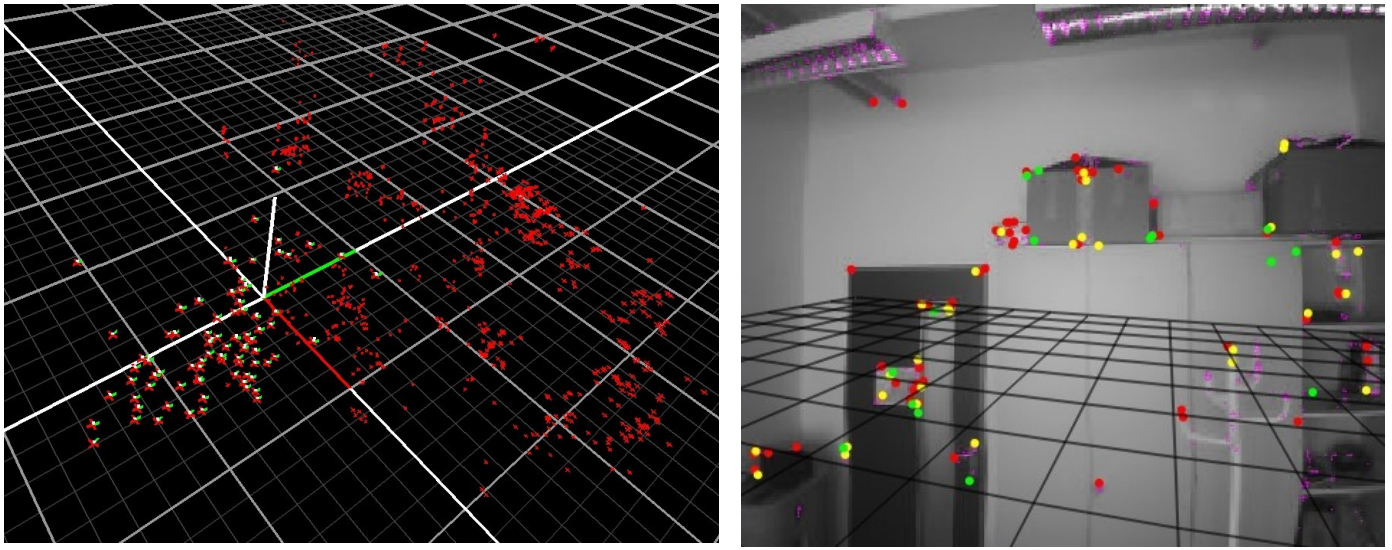
Problems: Unreliable, no scale

2. PTAM

Parallel Tracking and Mapping [Murray '07]:

→ keyframe based, monocular SLAM system

→ open-source

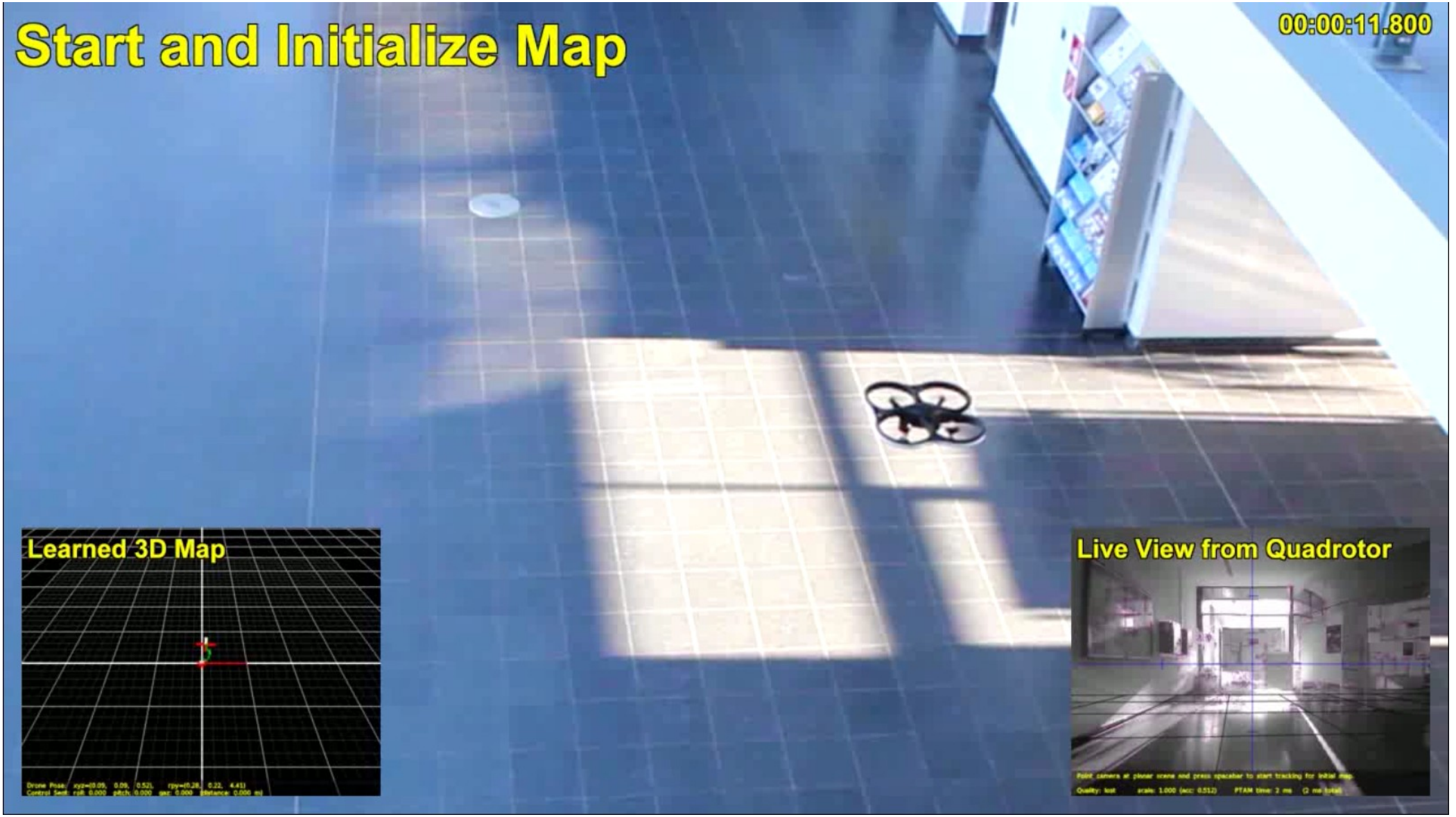


Problems: Unreliable, no scale

→ **enhance reliability** by incorporating IMU data

→ **add scale-estimation** from altimeter

2. PTAM


Start and Initialize Map
00:00:11.800
Learned 3D Map
Live View from Quadrotor

2. PTAM

tum_ardrone ROS package

tum_ardrone GUI

Send Commands

```
autoInit 500 800 4000 0.5

setReference $POSE$
setInitialReachDist 0.2
setStayWithinDist 0.3
setStayTime 3
lockScaleFP

goto 0 0 1 0
goto 0 0 0 0

goto -1 -1 0 0
goto -1 1 0 0
goto 1 1 0 0
goto 1 -1 0 0
goto -1 -1 1 0
goto -1 1 1 0
goto 1 1 1 0
goto 1 -1 1 0
goto 0 0 1 0
goto 0 0 0 0
```

Load File: jax.txt

Clear and Send | Clear | Send | Reset

Land | Takeoff | Emergency | Toggle Cam | FlatTrim

Node Communication Status

Drone Navdata: 199 Hz
Drone Control: 5 Hz
Pose Estimates: 30 Hz
Joy Input: 0 Hz
Pings (RTT): 21 (500B), 25 (20kB)
Motors: 0.000000 0.000000 0.000000

Autopilot Status:

Idle (Queue: 0)
Current: NULL
Next: NULL
Target: (0.00, 0.00, 0.00), 0.0
Error: (0.00, 0.00, 0.00), 0.0 (|.| 0.00)
Cont.: r 0.00, p 0.00, g 0.00, y 0.00

Stateestimation Status:

PTAM: Idle
Map: -
Scale: 1.000 (1 in, 0 out), acc: 0.51
ScaleFixpoint: FIX
Drone Status: Unknown (100 Battery)

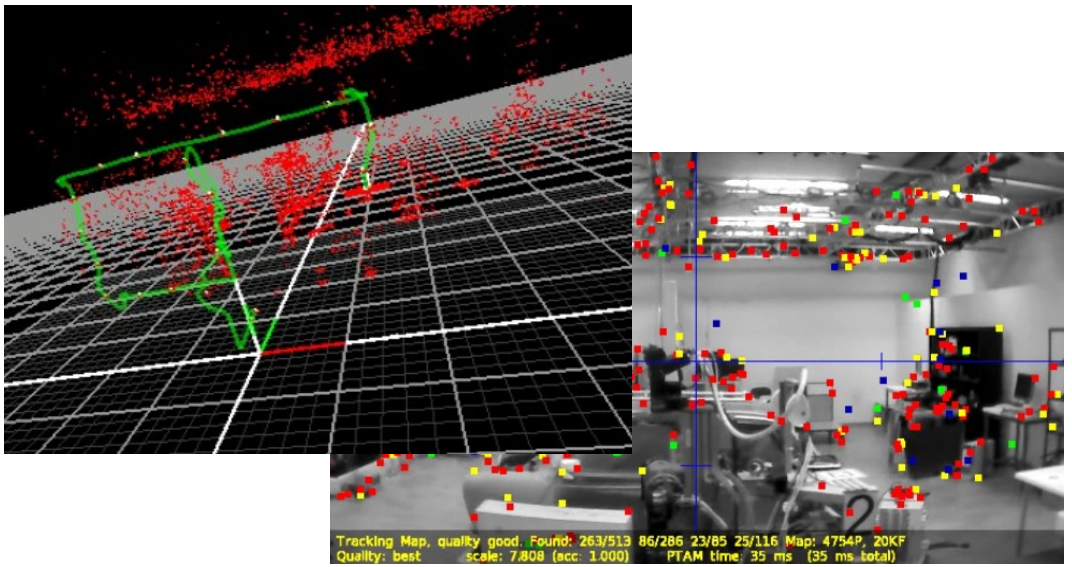
Control Source:

○ Keyboard ○ Joystick
○ Autopilot ◉ None
☑ Use Onboard Hovering
☑ Ping Drone (every 1s)

Messages

```
PTAM has been reset.
PTAM initialization started (took first KF)
PTAM has been reset.
PTAM initialization started (took first KF)
PTAM has been reset.
PTAM initialization started (took first KF)
PTAM initialization started (took first KF)
PTAM has been reset.
PTAM initialization started (took first KF)
PTAM has been reset.
```

Node 1: GUI + backup control



Node 2: State-estimation (SLAM, EKF)

```
engelj@lapcremers38:~$ rosrun tum_ardrone drone_autopilot
[ INFO] [1369780328.885242358]: Started TUM ArDrone Autopilot Node.
set minPublishFreq to 110ms
```

Node 3: Autopilot (PID Controller)

Open Source @ www.ros.org/wiki/tum_ardrone

Future Work

- **Increase Range:**
  Augment PTAM with eg. FABMAP and/or g2o

- **Initialization:**
  Faster & more robust using gyro readings;
  Automatic re-initialization

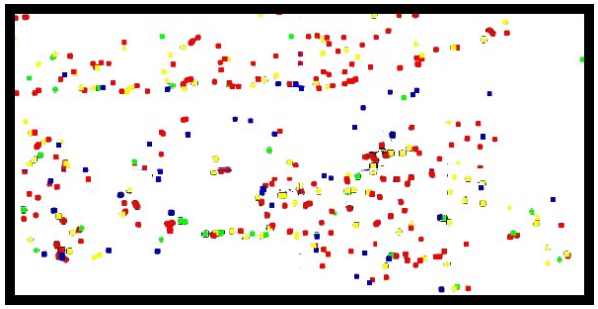
- **Performance (→onboard):**
  e.g. only 3DoF coarse tracking + gyro.

Future Work

- **Add feature**
  e.g. person following, gesture recognition, …
- **Obstacle Avoidance**
  e.g. using optical flow.
- …

Preview

Monocular SLAM without keypoints



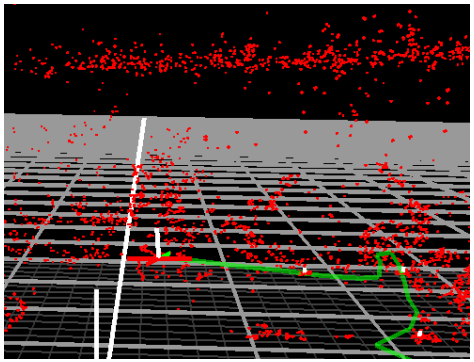


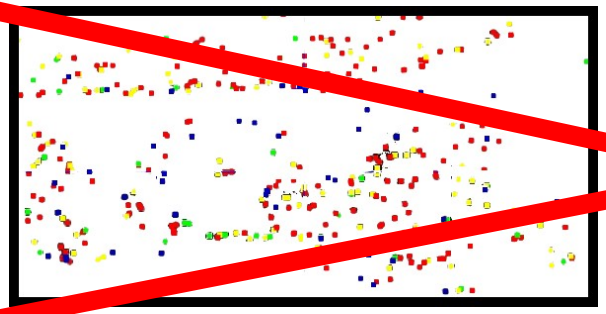
Image (pixels)

Features (e.g. point-positions)

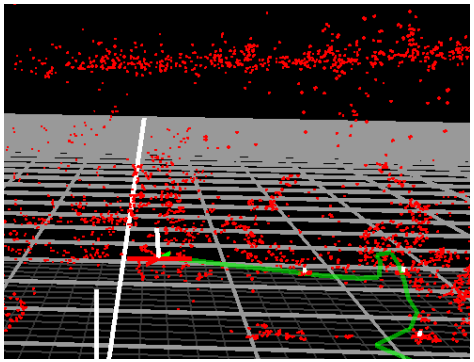
Mapping, Tracking

Monocular SLAM without keypoints

Image
(pixels)



Features
(e.g. point-positions)



Mapping,
Tracking

Preview

Thanks!

Questions?