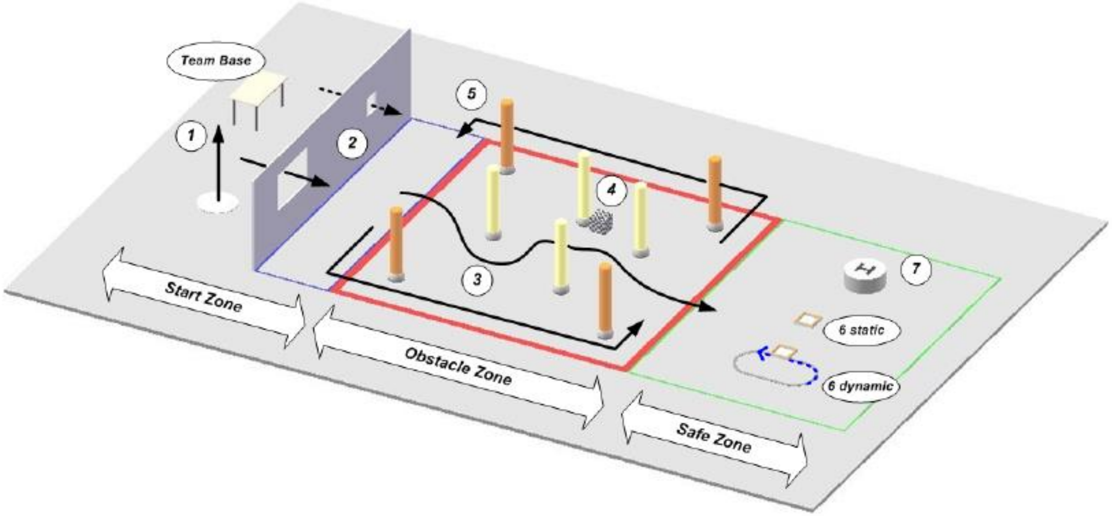
Autonomous Perception and Navigation for Micro Aerial Vehicle

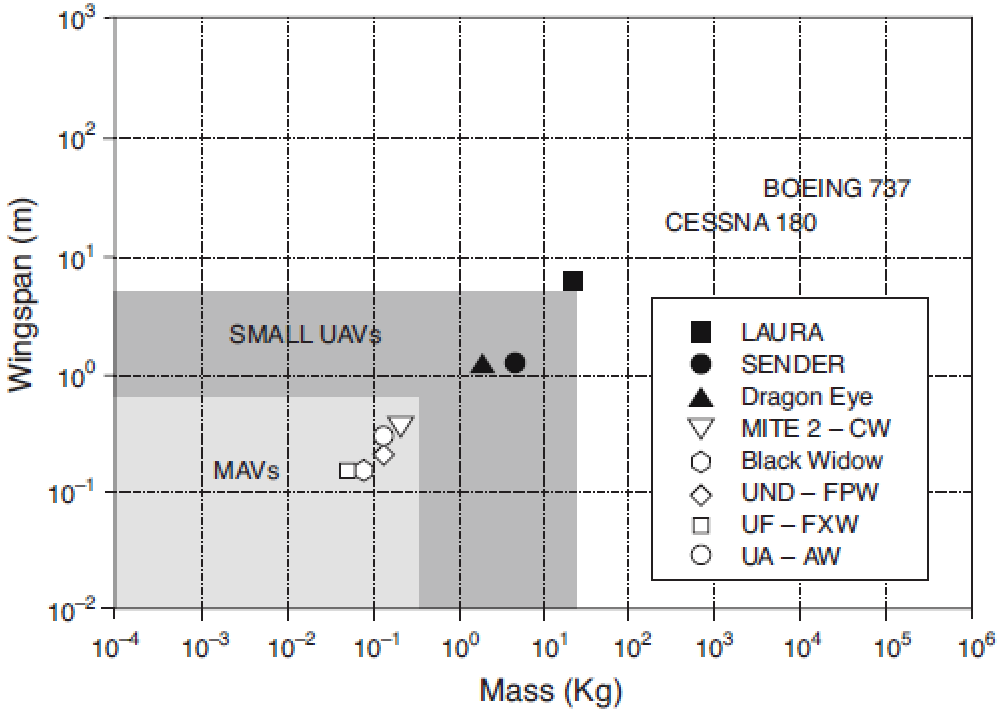


Arnoud Visser

KIvI Niria Drone Day, Utrecht, November 5, 2013

Universiteit van Amsterdam
Intelligent Robotics Laboratory

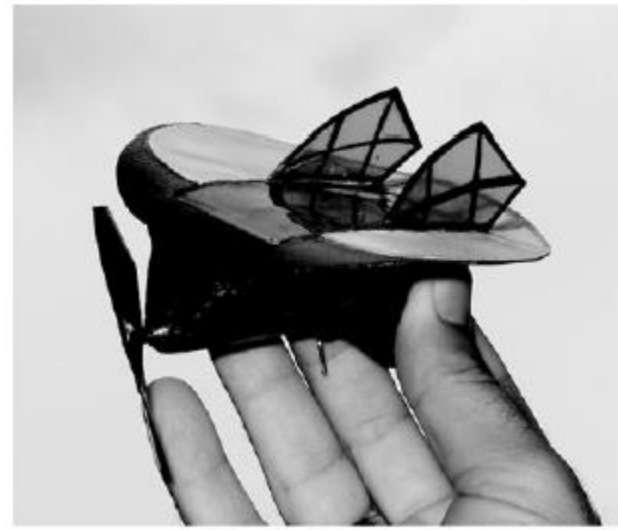
Context



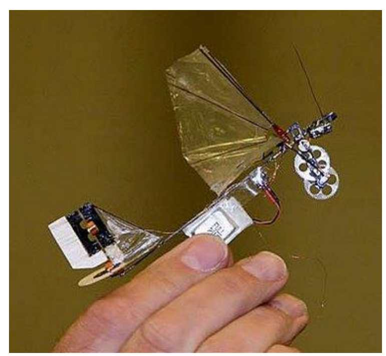
Courtesy [1] T. J. Mueller, "On the birth of micro air vehicles," International Journal of Micro Air Vehicles, vol. 1, no. 1, pp. 1–12, 2009.

Smaller UAVs

Black Widow, Courtesy AeroVironment Inc.



MAV of the University of Florida

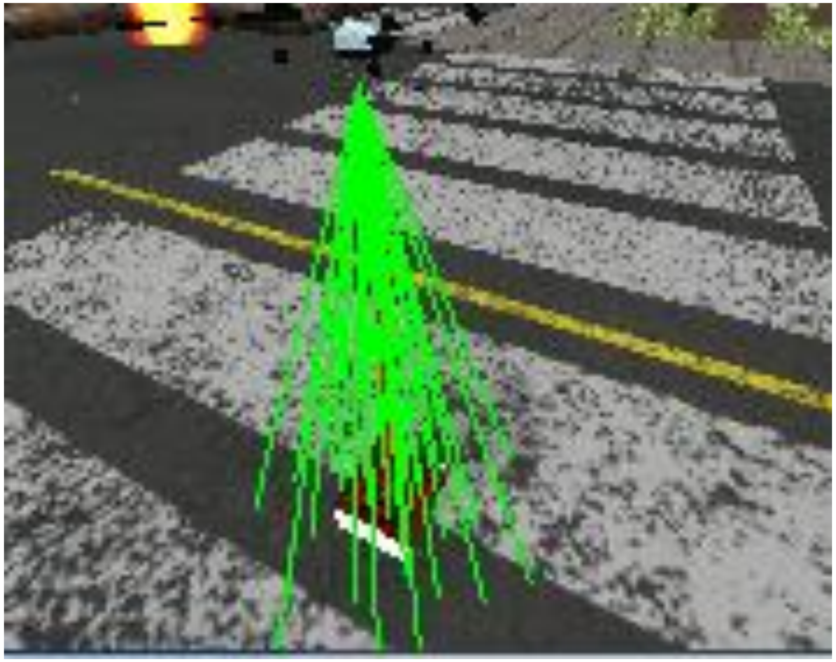


[Delfly Micro, TU Delft]

Experience with UAVs

The versatile scouting capabilities of small air vehicles make them very useful for surveillance, inspection and search & rescue

Hard to provide autonomy



The limited sensor suite and the fast movements make it quite a challenge to fully automate the navigation for such platform

Autonomy increasingly important

Year	Score
2007[3]	$\sum_{K \in tasks} \alpha_K \times T_K (2 - \frac{L}{L_{max}})^3$ with $\alpha_K = (1, 2, 6)$ and $L_{max} = 500mm$
2008	$\sum_{K \in tasks} \alpha_K \times T_K (2 - \frac{L}{L_{max}})^3$ with $\alpha_K = (1, 2, 6)$ and $L_{max} = 1000mm$
2009[4]	$\sum_{K \in tasks} (\alpha_K - e_K) \times T_K (2 - \frac{L}{L_{max}})^3$ with $\alpha_K = (1, 2, 6, 12)$ and $L_{max} = 700 - 800mm$
2010	$\sum_{K \in tasks} \alpha_K \times T_K (2 - \frac{L}{L_{max}})^2$ with $\alpha_K = (1, 2, 6)$ and $L_{max} = 1000mm$
2011	$\sum_{K \in tasks} (\alpha_K - e_K) \times T_K (2 - \frac{L}{L_{max}})^3$ with $\alpha_K = (1, 6, 12)$ and $L_{max} = 700 - 800mm$
2012 [5]	$\sum_{K \in tasks} (\alpha_K - e_K) \times T_K \times D \times (2 - \frac{L}{L_{max}})^2$ with $\alpha_K = (1, 6, 12)$ and $L_{max} = 1000mm$
2013 [6]	$P \times \sum_{v \in vehicles} (\alpha_v - e_v) \times T_v \times \frac{1}{L}$ with $\alpha_K = (1, 4, 6, 12)$ and L in m

TABLE II. THE DEVELOPMENT OF THE SCORE-FUNCTION DURING RECENT COMPETITIONS.

Autonomy factor α_K

Level of control	α_K
Video based control: control of the MAV is manual (from complete manual control to attitude stabilized control)	1
Assisted flight control: the navigation is not completely autonomous but the low level control is augmented by additional controls (such as collision avoidance or hovering based on laser scanner or optical flow)	4
Autonomous flight control: the navigation is completely autonomous but the operator is controlling the mission and the payload, processing perception, and making decisions	6
Autonomous target detection: the navigation and decision making is not autonomous but the detection and processing of the targets is automatic	6
Autonomous mission control: not only the navigation but also the detection and decision making is autonomous, without assistance of the operator	12

TABLE I. THE AUTONOMY FACTOR α_K AND THE CORRESPONDING LEVELS OF CONTROL (COURTESY[1]).

[1] http://www.imav2013.org

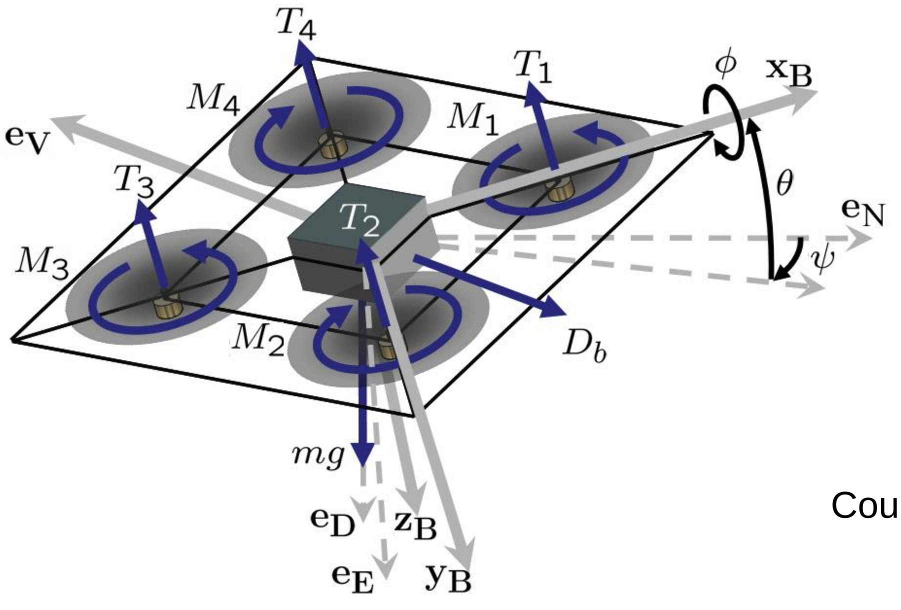
Motion model:
Validation of the simulation model

Carsten van Weelden and Nick Dijkshoorn

intelligent autonomous systems

Universiteit van Amsterdam
Informatica Instituut

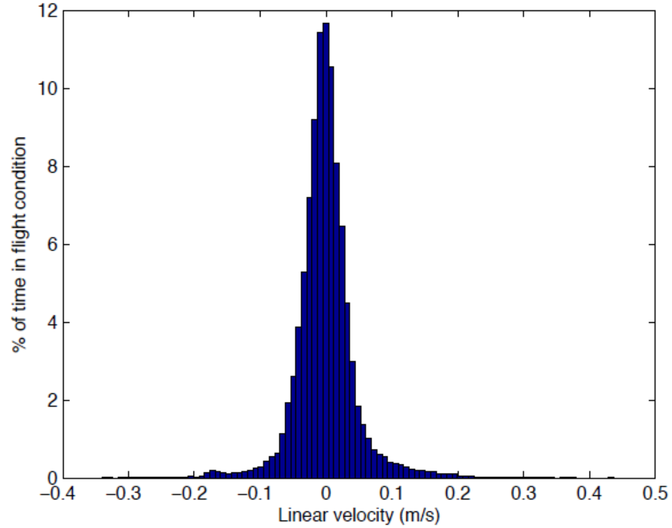
AR.Drone is stabilized quadrotor

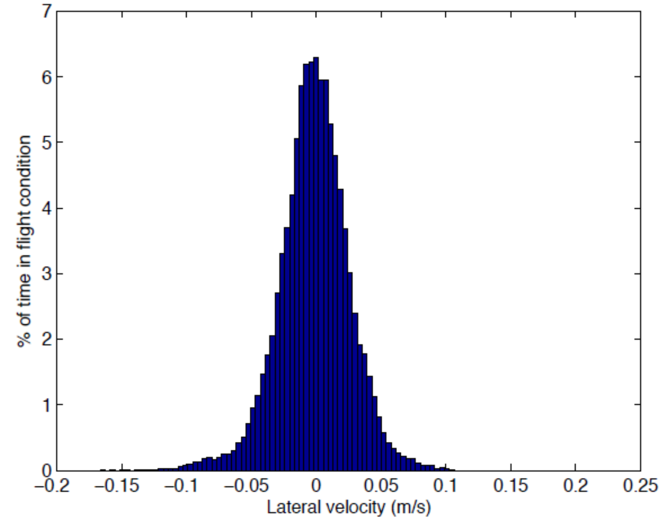


Courtesy Hoffman et al.

The AR.Drone is mainly controlled with his angle of attack α (θ in x_B)

Hovering

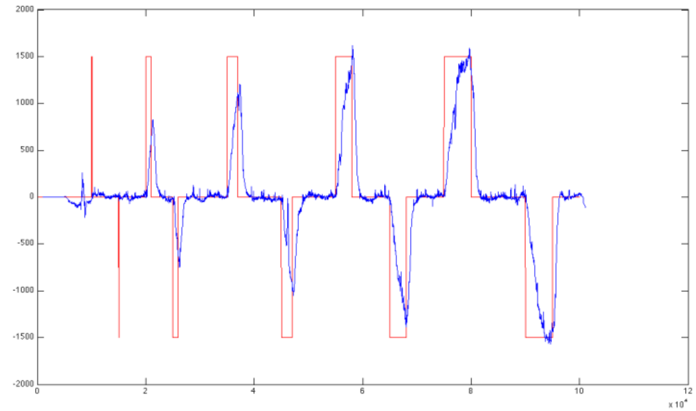




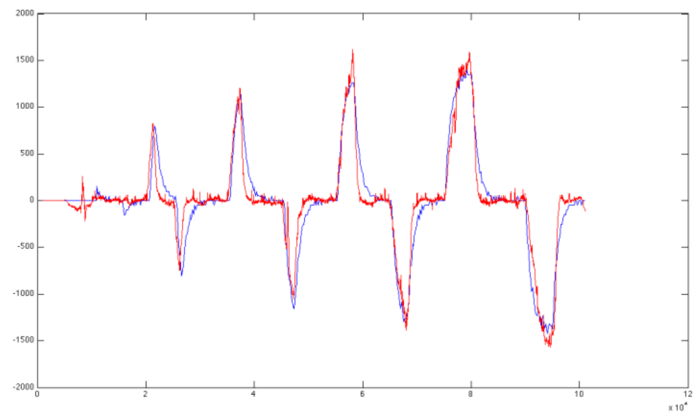
The AR.Drone is stabilized by optical flow of bottom camera. The variance in position is *7.07±0.12 cm* and in velocity is 0.0*422 m/s.*

Horizontal movement

The AR.Drone is given forward and backward steppulses with a fixed amplitude.



The 5 second pulses are used to estimate the speed at this angle.



$$v = 0.2967 \cdot s \cdot \theta_{max}$$

Pulses with amplitude (*s=0.15*).

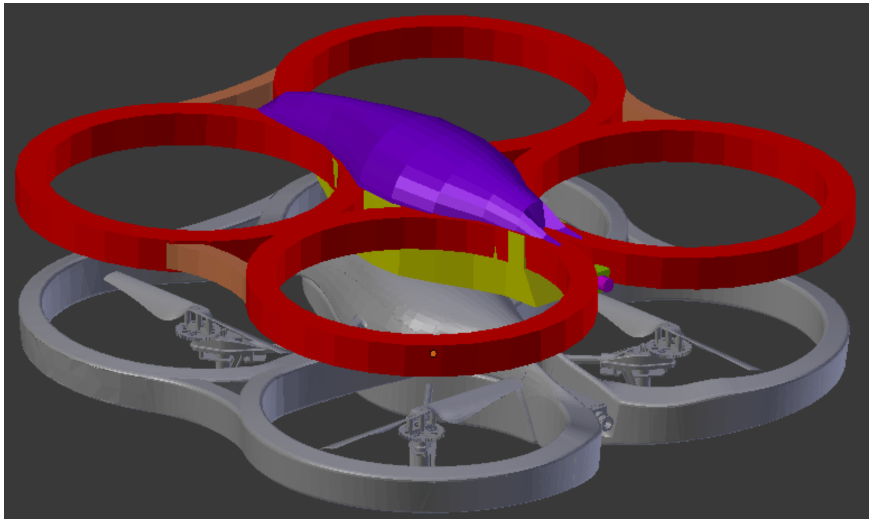
Horizontal movement

	Control signal s				
	0.05	0.10	0.15	0.20	0.25
$\bar{v}$ (m/s)	0.4044	0.6284	1.4427	1.7587	2.2094
σ_v (m/s)	0.096	0.226	0.070	0.126	0.165
θ (deg)	1.4654	2.9025	4.1227	5.7457	7.4496
σ_θ (deg)	0.455	0.593	0.482	0.552	0.921

Experiments with different amplitudes of the signal *s* revealed the following relationship between angle of attack *α* and the speed *v.*

$$v = 0.2967 \cdot s \cdot \theta_{max}$$

Simulation model



Parrot provided a very detailed 3D-model. Our team created a simplified model with 3142 vertices.

Overall, the dynamic behavior closely resembles the dynamics of the real system.


Unreal Tournament 2004

Sensor model:

Optical Flow based Obstacle Avoidance

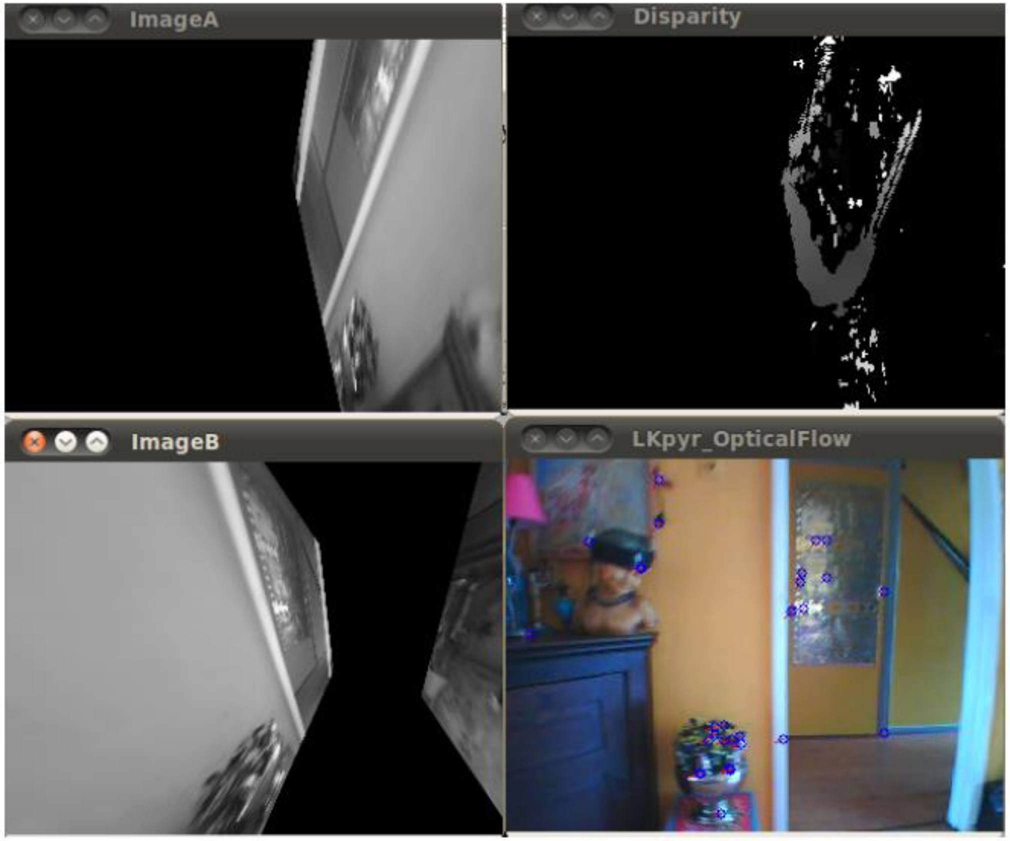
Robrecht Jurriaans

Robrecht Jurriaans, Flow based Obstacle Avoidance for Real World Autonomous Aerial Navigation Tasks, Bachelor thesis, Universiteit van Amsterdam, August 2011.

Optical Flow algorithm

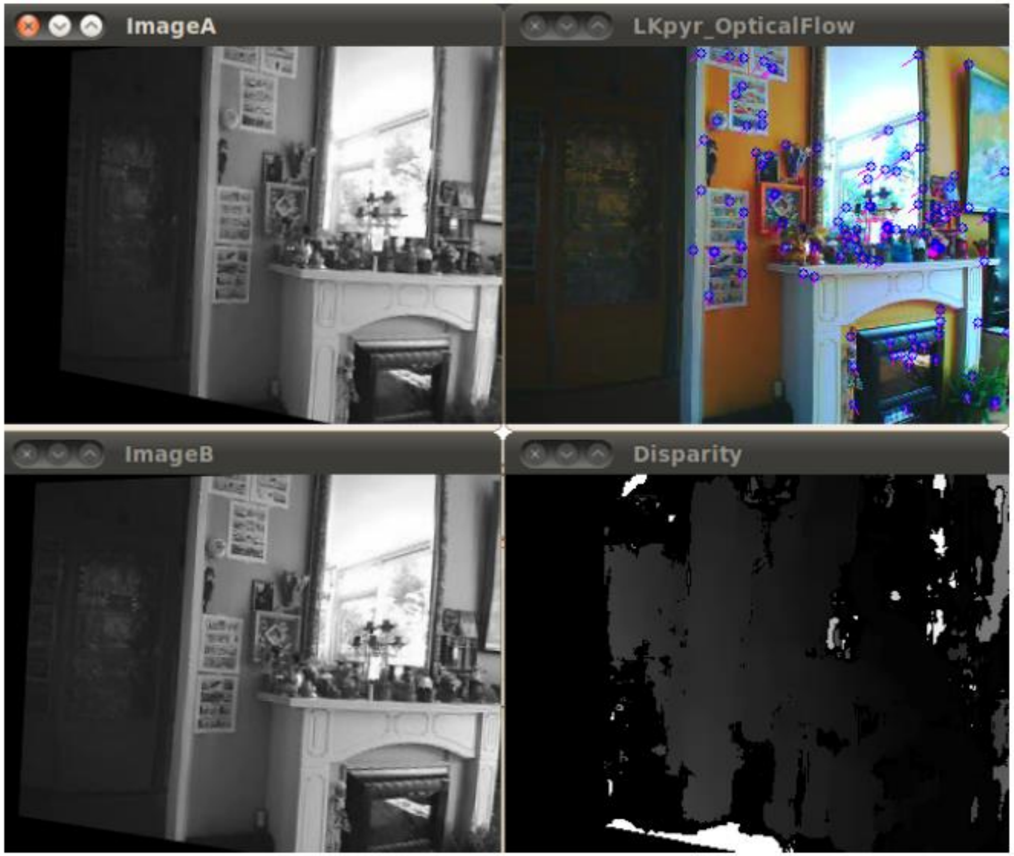
- Pyramidal Lucas-Kanade implementation $v = (A^T A)^{-1} A^T b$
- Tracking of Shi-Tomasi features
- RANSAC to classify inliers / outliers
- Hartley's algorithm to estimate the translation of the camera in world coordinates

Disparity map (low texture)



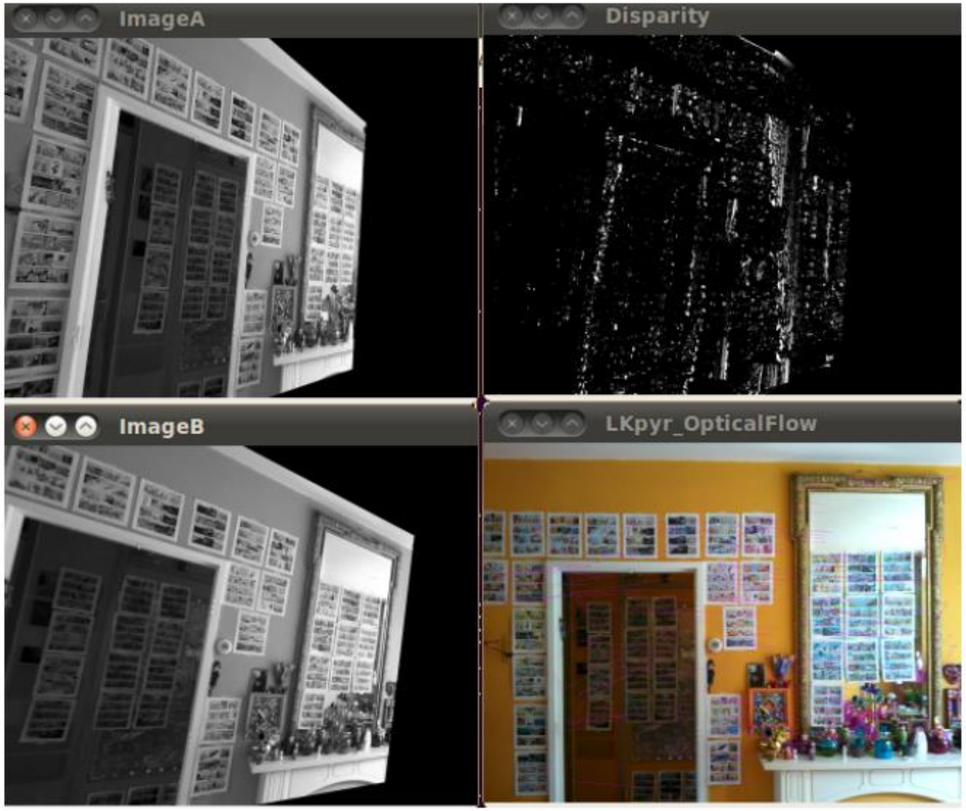
- Not enough features to build reliable disparity map

Disparity map (added texture)



- The disparity map shows more detail

Disparity map (higher resolution)



- Nearly enough features for a robust estimation of the translation in Hartley's algorithm

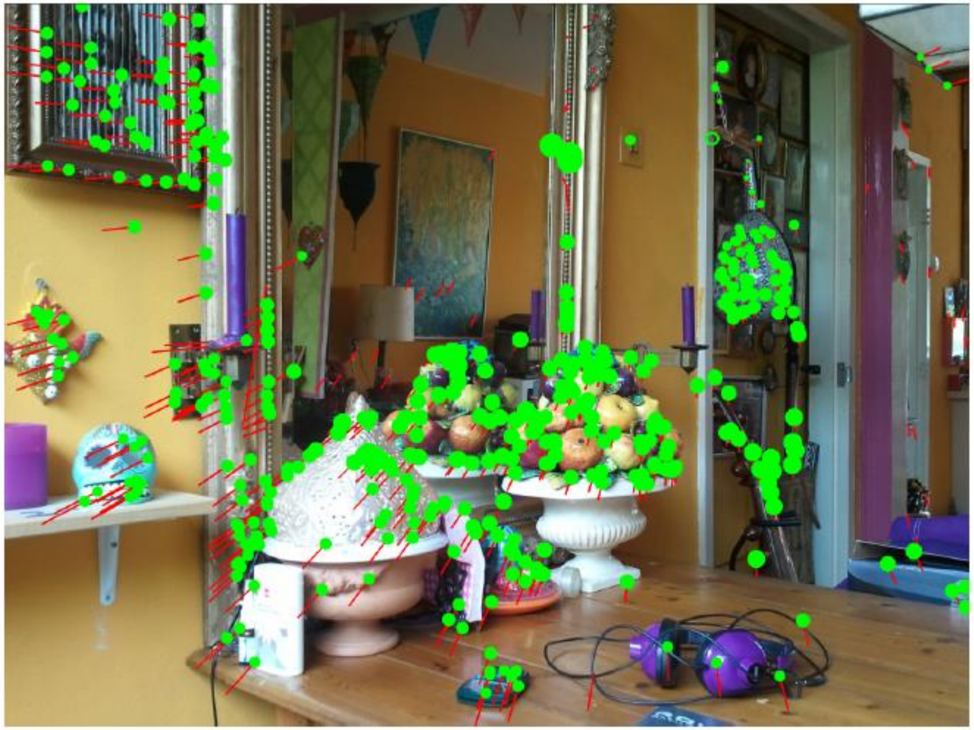
Features needed

TABLE 1: *The amount of point pairs for stereo vision*

Technique	Points
Stereo Vision, minimum necessary	2 views of 3x3 chessboard
Stereo Vision, advice for high quality	10 views of 7x8 chessboard
Optical Flow - Low texture count	150 corner count
Optical Flow - High texture count	260 corner count
Optical Flow - High resolution	380 corner count

- Typically, the Shi-Tomasi algorithm can pair 75% of the points in both images
- Typically, RANSAC removes 25-50% of the point pairs

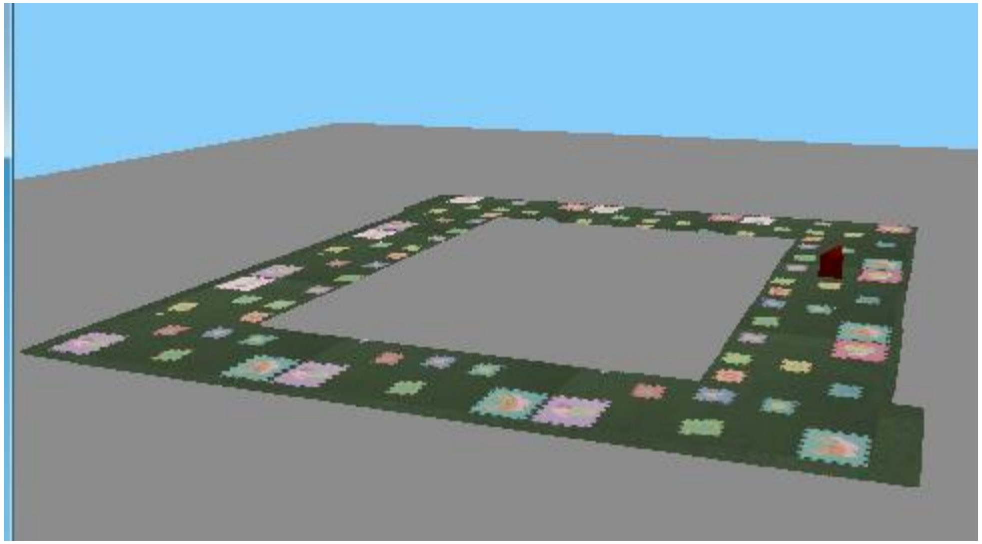
Depth Map



- Hartley’s algorithm cannot provide scale. To estimate depth time to contact has to be estimated from the optical flow.
- Disparity can be used to create a repulsive force when an obstacle is approached

Localization and Mapping challenge:

Visual SLAM combined with sonar and inertia measurements

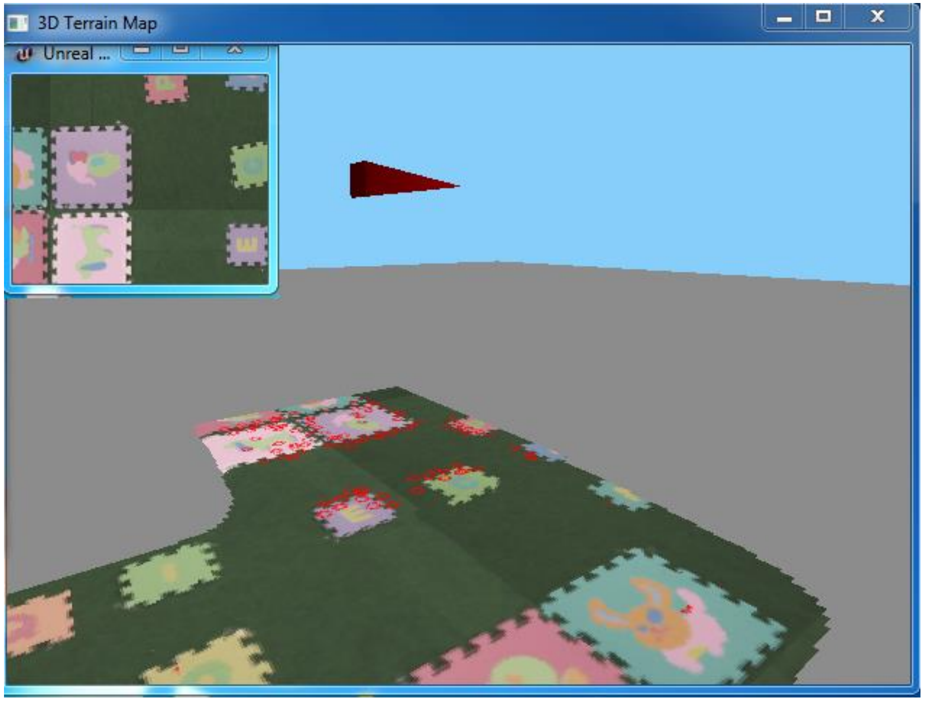


Nick Dijkshoorn

intelligent autonomous systems

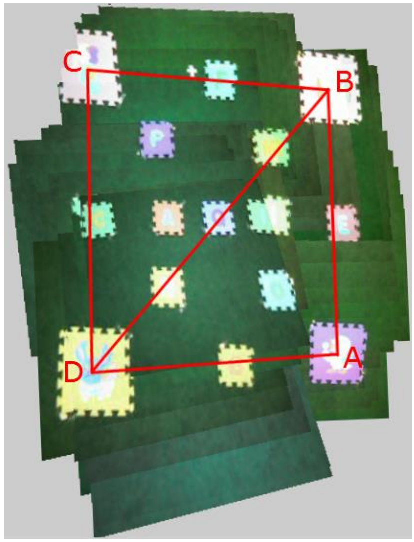
Universiteit van Amsterdam
Informatica Instituut

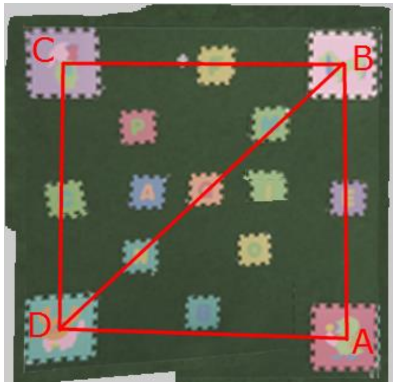
Map Stitching algorithm



- Tracking of SURF features
- RANSAC to classify inliers / outliers
- Back-projection with least-square optimization to estimate the perspective transformation (replaced by an estimate of the camera's transformation in OpenCV's SolvePNP)

Map Stitching results





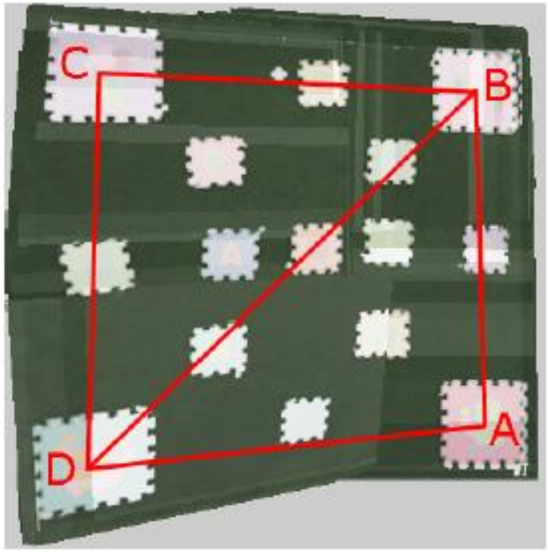
Care has been taken to reproduce the real circumstances:

- decreased saturation,
- increased brightness,
- downsampled resolution.

landmarks	A-B	B-C	C-D	D-A	B-D
AR.Drone					
mean error (m)	0.385	0.146	0.608	0.156	0.445
error (%)	29.6	11.2	46.8	12.0	24.1
USARSim simulator					
mean error (m)	0.019	0.047	0.026	0.075	0.028
error (%)	1.46	3.62	2.00	5.77	2.15

Map Stitching results

In addition, white balance variations are added. Now the average feature distance increases from *22.1px* to *32.9px* (real AR,Drone *32.7px)*

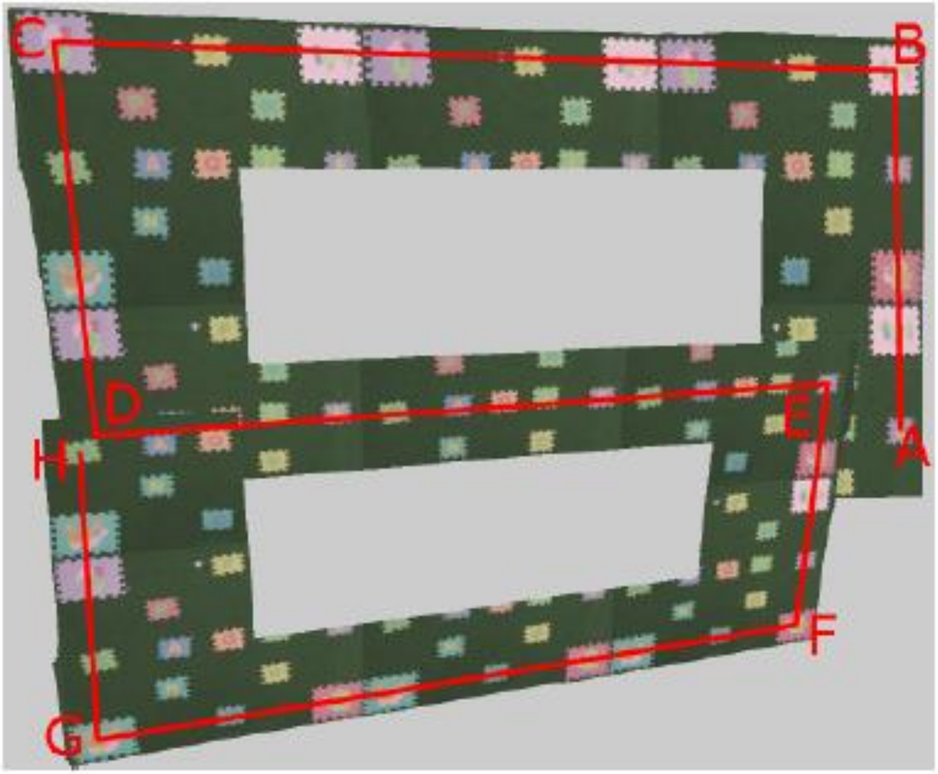


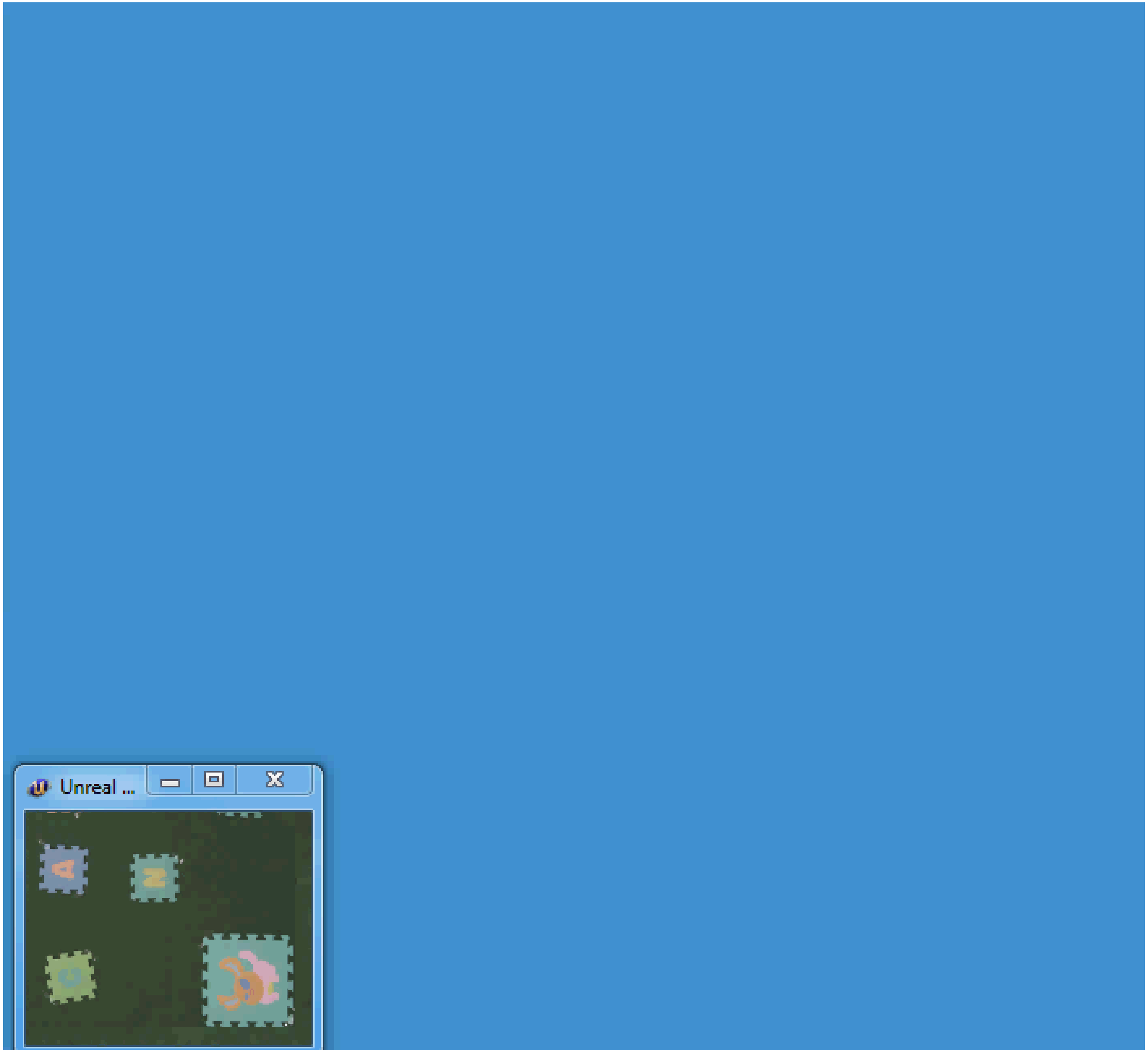
landmarks	A-B	B-C	C-D	D-A	B-D
USARSim simulator (white balance variations)					
mean error (m)	0.031	0.181	0.215	0.254	0.190
error (%)	2.21	12.93	15.36	18.14	10.27

Scaling up results

When the map is scaled up 3x, the lack of global optimization becomes visible.

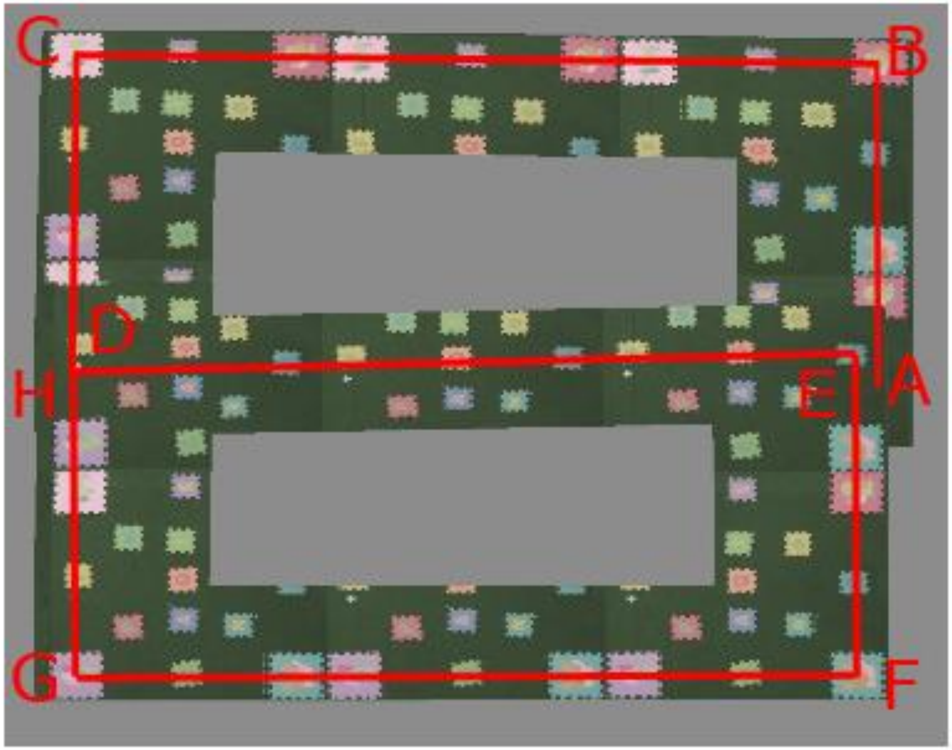
landmarks	A-B	B-C	C-D	D-E	E-F
mean error (m)	0.220	0.87	0.579	0.220	0.523
error (%)	10.48	19.33	27.57	4.89	24.90
landmarks	F-G	G-H	H-A	B-G	
mean error (m)	0.011	0.244	0.788	0.14	
error (%)	0.24	11.62	17.51	2.20	




Unreal ...

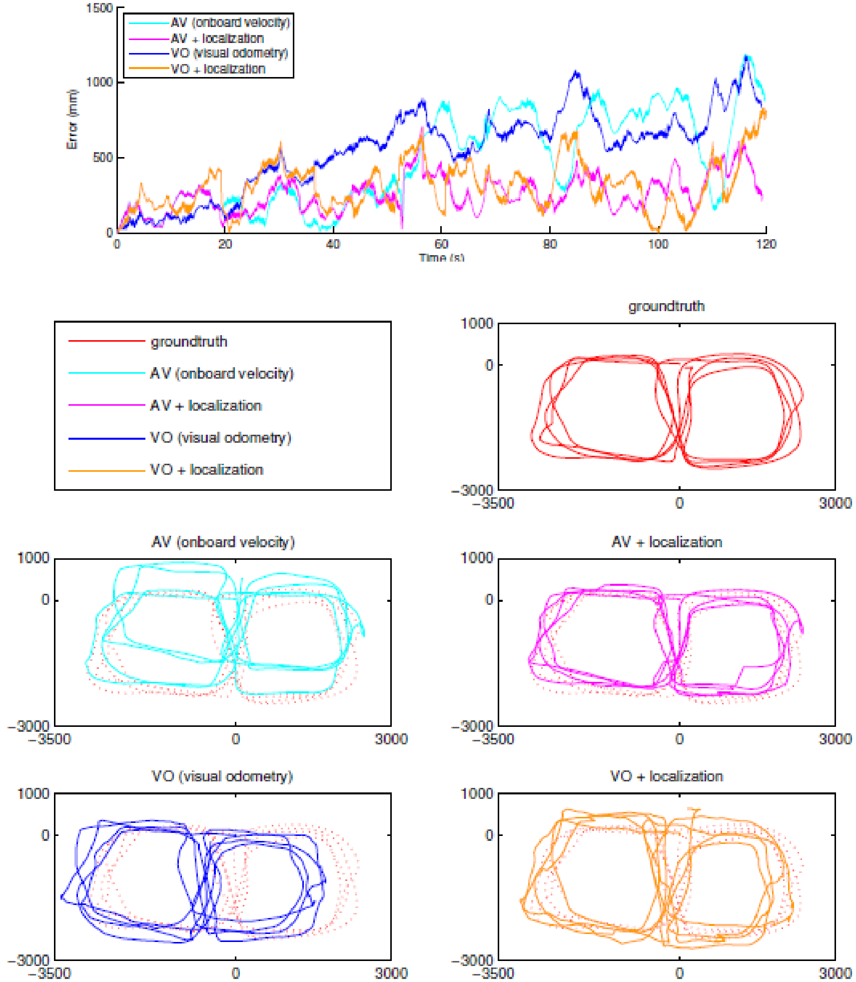
Additional information results

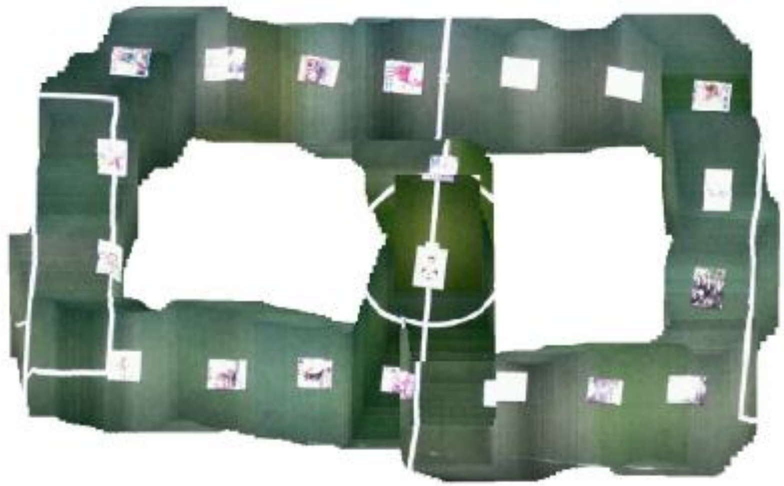
Including information from inertia sensors (using a extended Kalman filter) solves part of the problem.



landmarks	A-B	B-C	C-D	D-E	E-F
mean error (m)	0.029	0.689	0.049	0.565	0.013
error (%)	1.38	15.31	2.33	12.56	0.62
landmarks	F-G	G-H	H-A	B-G	
mean error (m)	0.596	0.080	0.720	0.243	
error (%)	13.24	3.81	16.0	3.83	

Comparison of methods

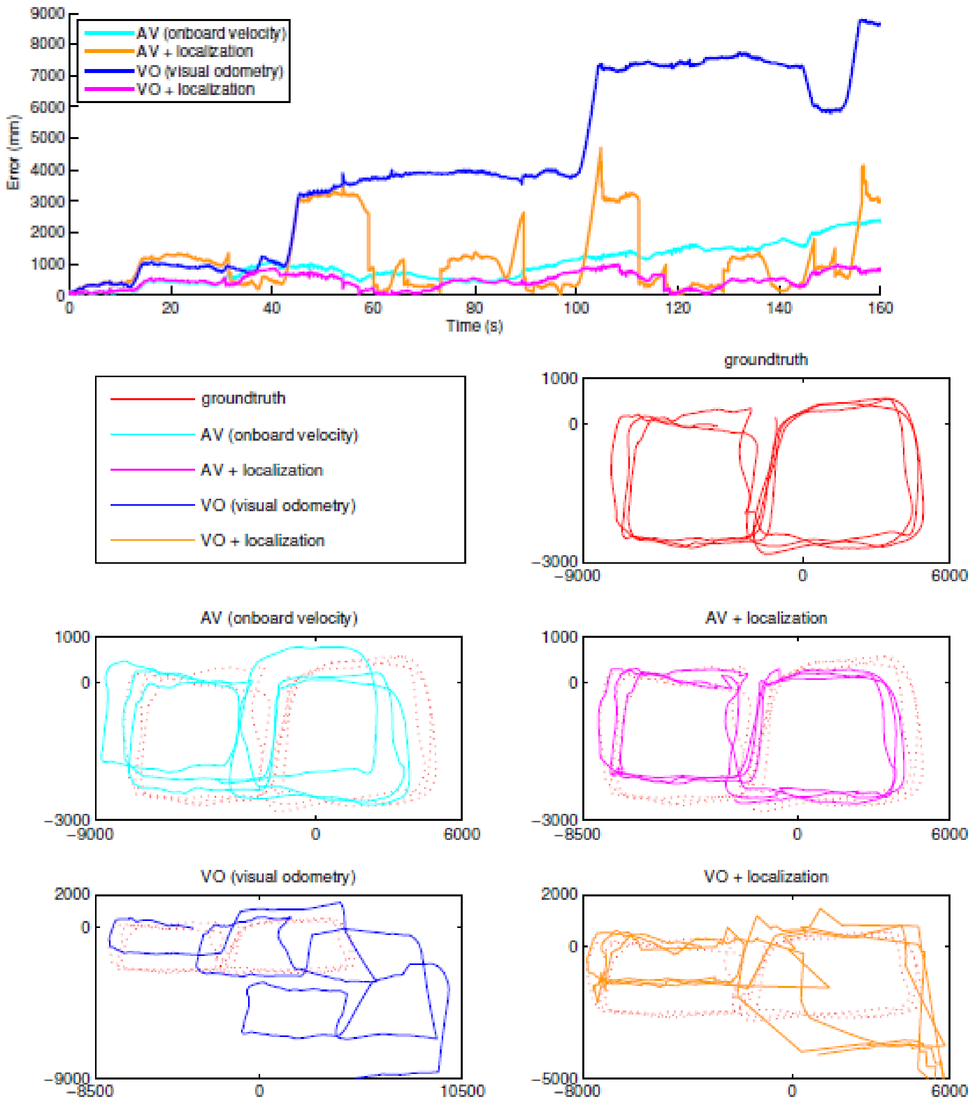


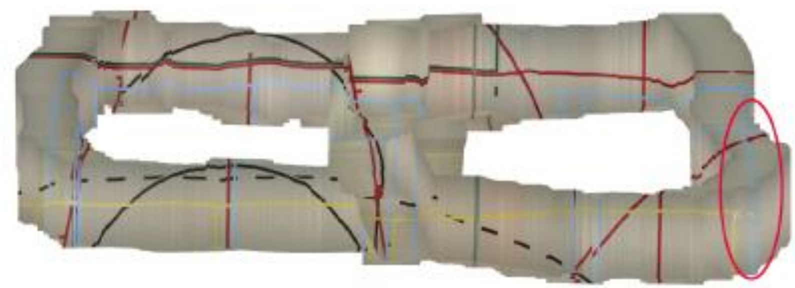


Map used for localization

- Localization results improve a factor 2 with localization.

Comparison of methods





Map used for localization

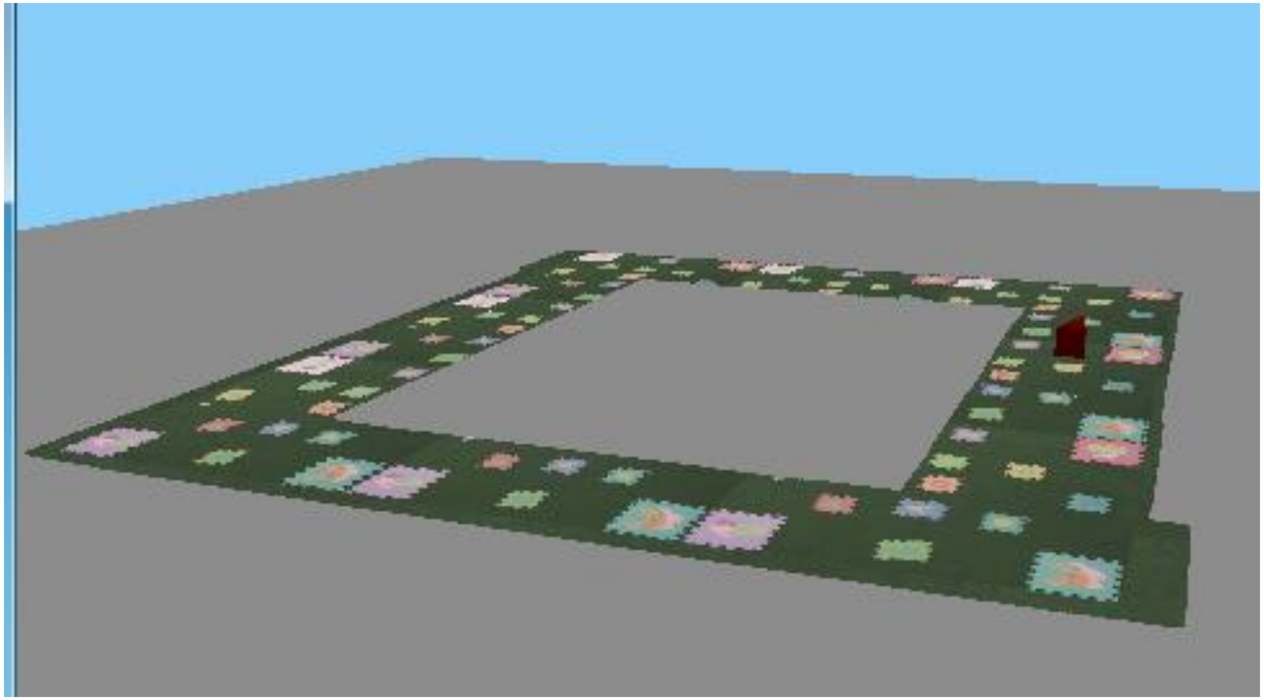
- Localization make visual odometry more robust.

Restrictions on variables

Method	Mean absolute error (mm)	Mean relative error (percentage of trajectory length)
visual odometry (VO)	552	0.828%
visual odometry (VO) - inliers based	734	1.101%
visual odometry (VO) - Euclidean transform	967	1.449%
visual odometry (VO) - affine transform	3784	5.672%
visual odometry (VO) - perspective transform	8923	13.375%

- Reducing the degrees of freedom results in more robustness against noise.

Resumé

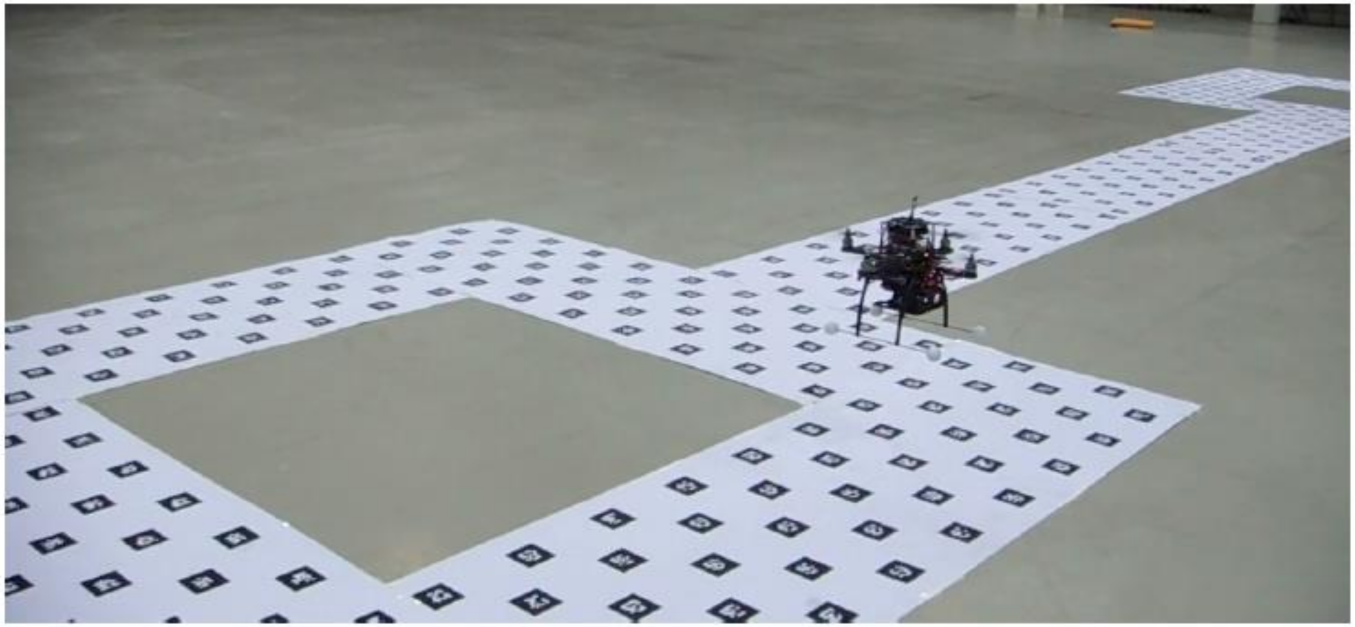


A visual map can be created with the low resolution bottom camera, which could be used for localization.

[Two rounds of autonomous flight by UvA team at IMAV 2011]

Comparison

- Model based approach

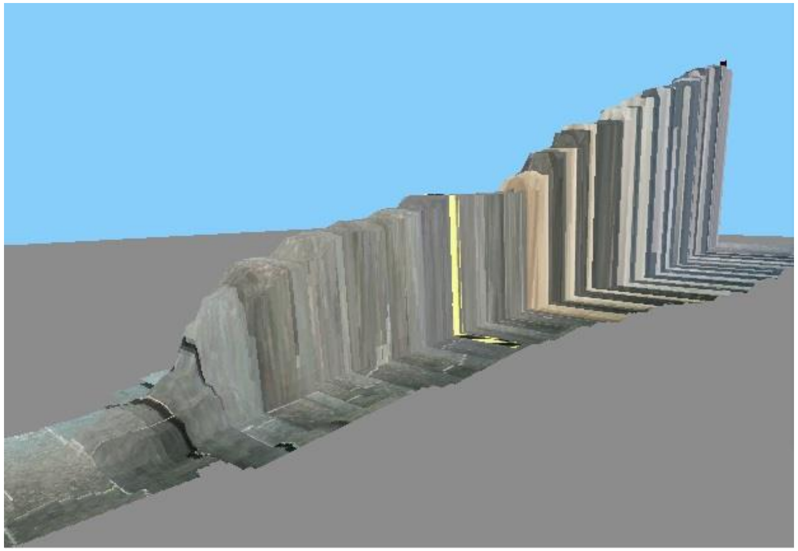


PixHawk localization based on markers [9].

[Two rounds of autonomous flight by PixHawk team at IMAV 2010]

[9] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "Pixhawk: A system for autonomous flight using onboard computer vision," in IEEE international conference on Robotics and automation (ICRA), 2011, pp. 2992–2997.

Localization and Mapping challenge:
Elevation map based on sonar

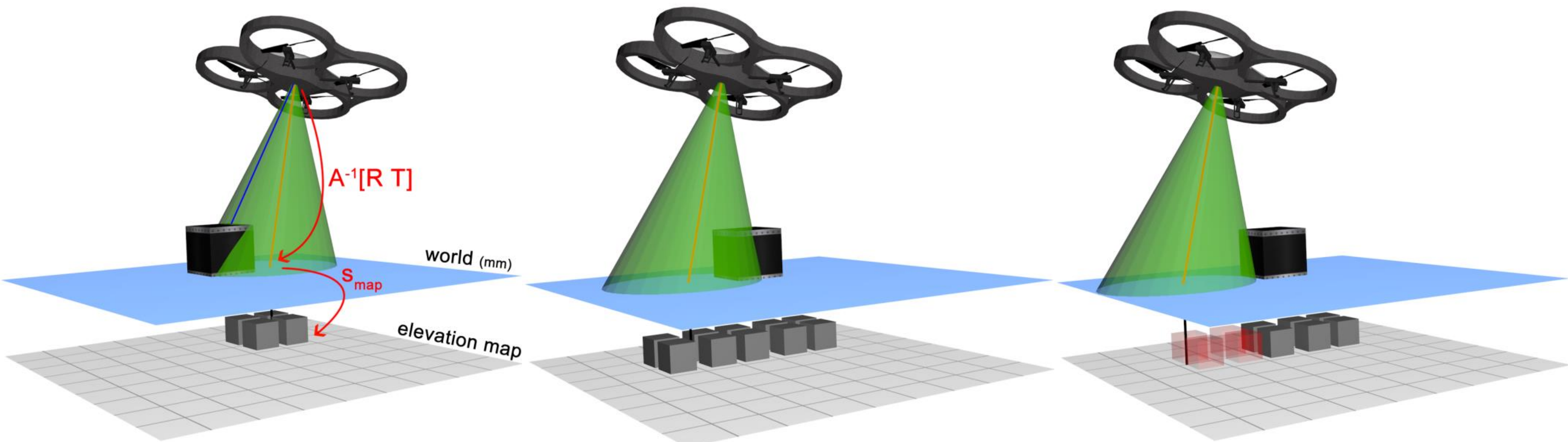


Nick Dijkshoorn

intelligent autonomous systems

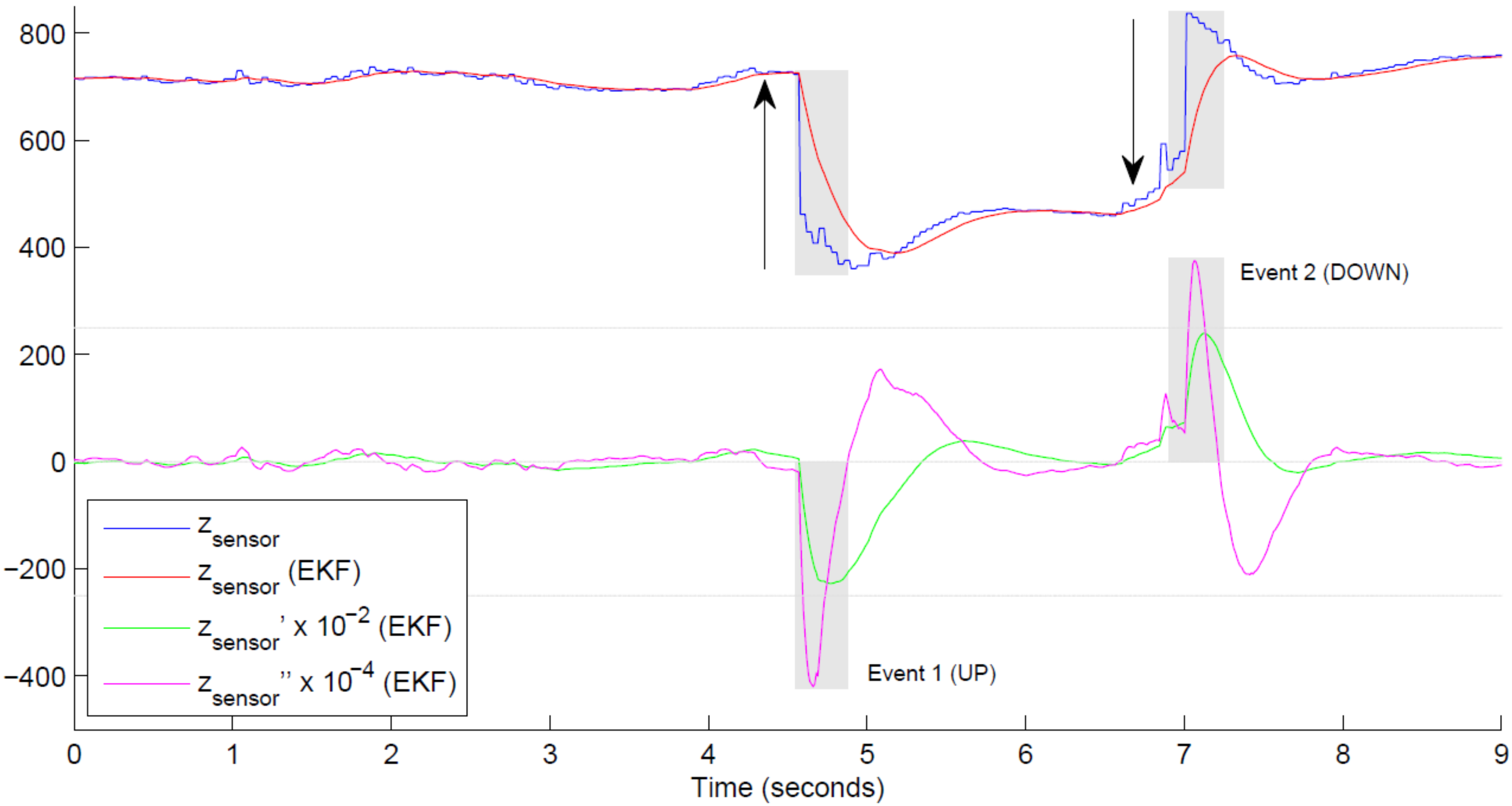
Universiteit van Amsterdam
Informatica Instituut

Update and refinement

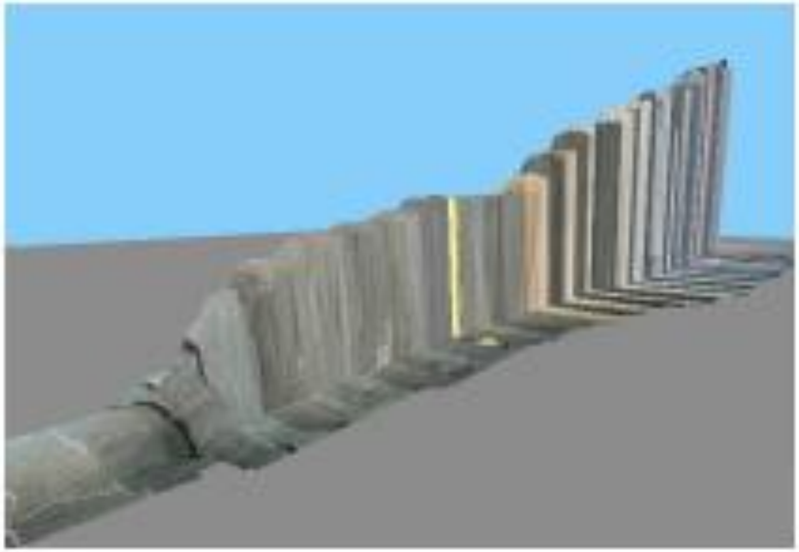


With the transformation **$A^{-1}[R\ T]$** **known, the height can be stored in a grid, which means that the flat floor assumption can hold.**

Automatic Height Estimation

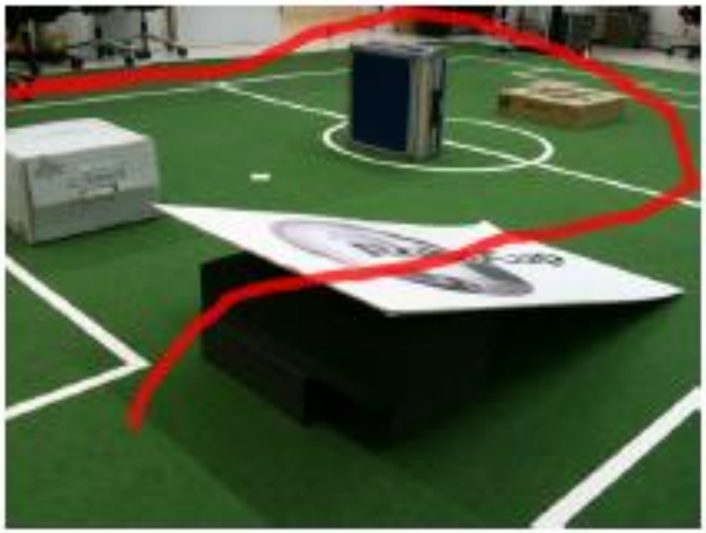


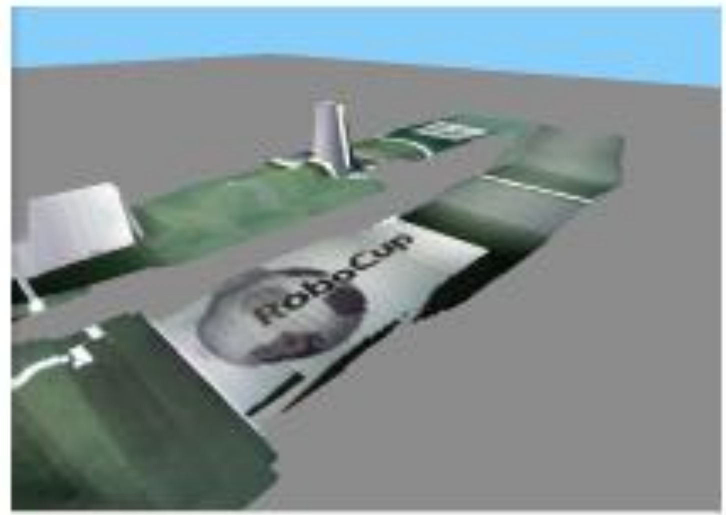
Resulting Map



- The height estimation underestimates the height of the staircase in our building.

Resulting Map



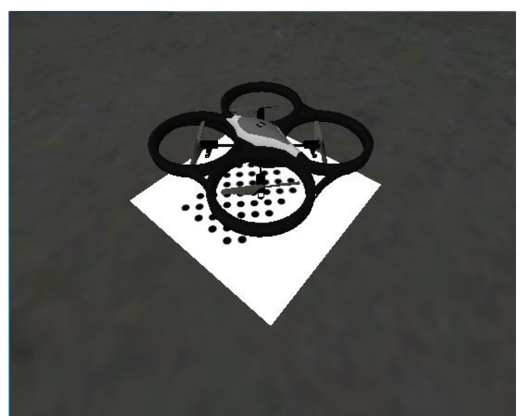


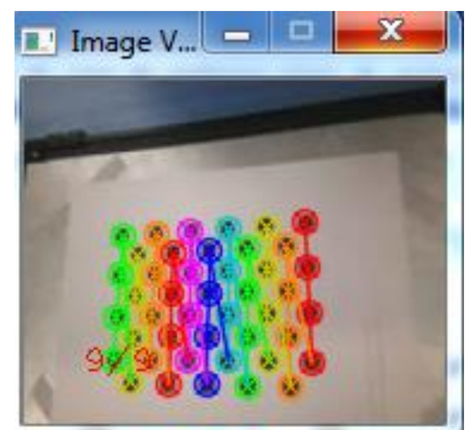
- Sudden elevations are well estimated, gradual elevations are cut off.

Resumé

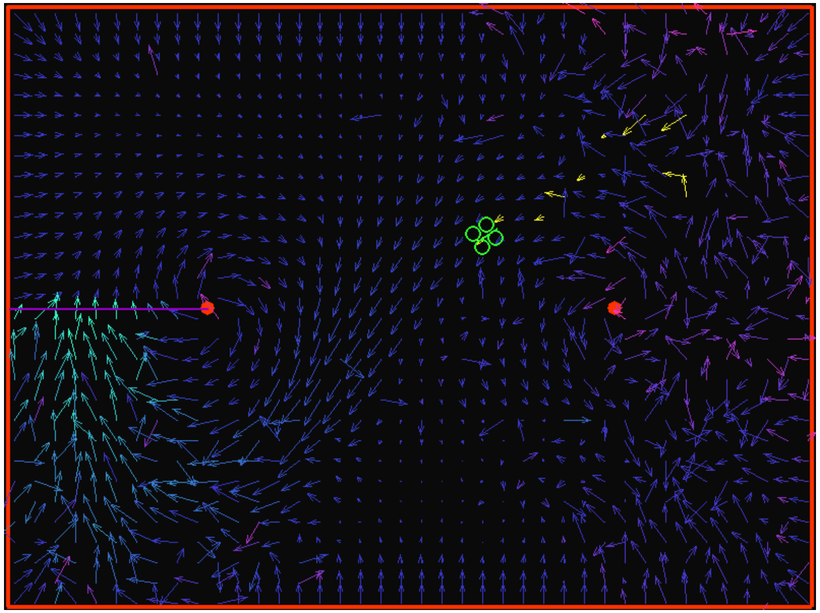
- Height estimates can only be made by a careful sensor fusion of external and internal measurements (sonar and inertia).

Universiteit van Amsterdam
UvA Rescue





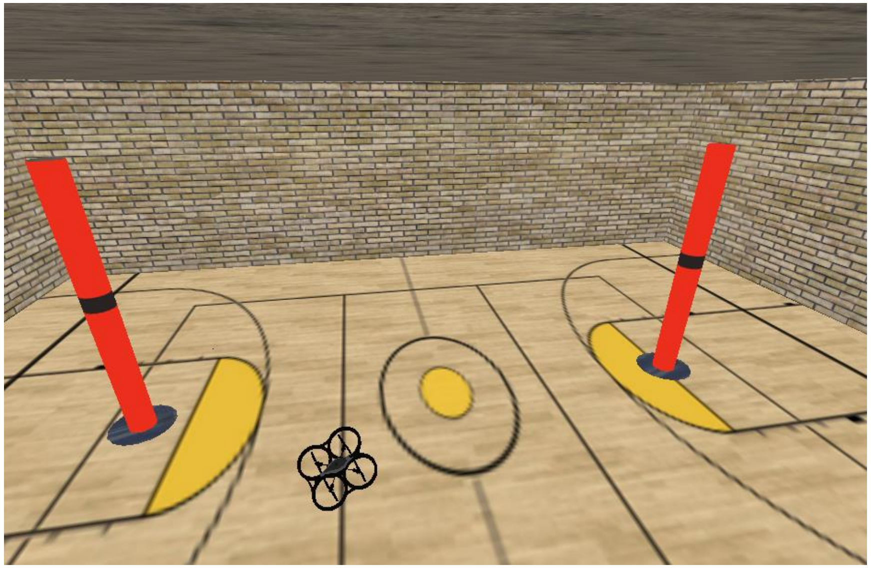
<u>Navigation challenge:</u>
Optimizing Artificial Force Fields

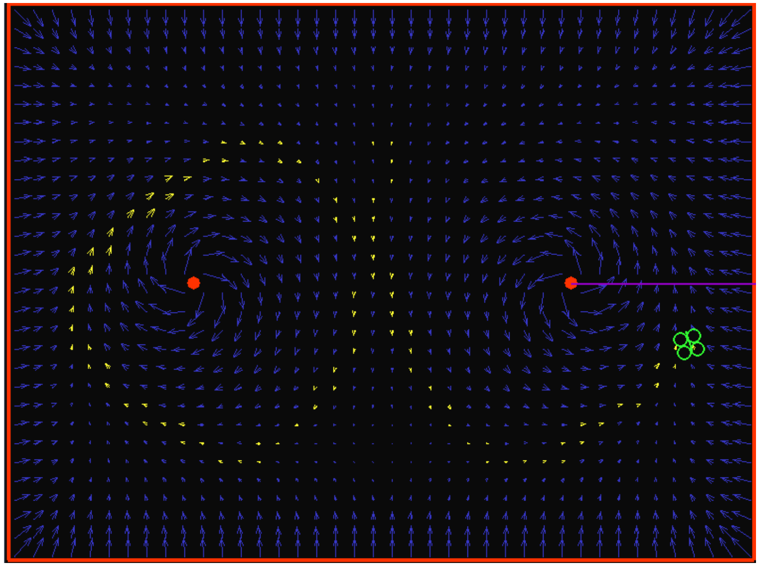


Martijn van der Veen

Martijn van der Veen, Optimizing Artificial Force Fields for Autonomous Drones in the Pylon Challenge using Reinforcement Learning, Bachelor thesis, Universiteit van Amsterdam, July 2011.

Initial Force Field

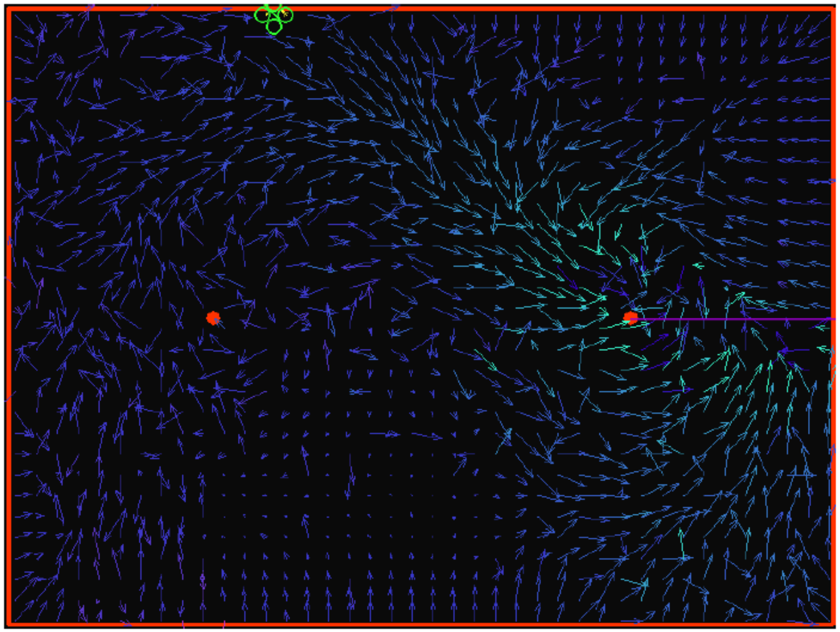


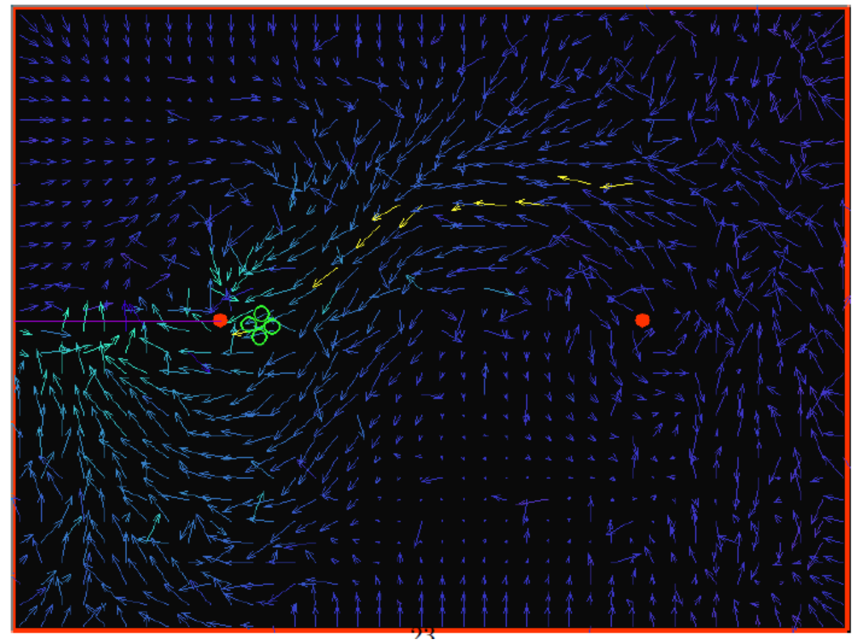


An initial field can be build up with a-priori knowledge

$$V(\vec{F}_{p-1}) \leftarrow (1-\alpha)V(\vec{F}_{p-1}) + \alpha(r(\vec{F}_p) + \gamma V(\vec{F}_p))$$

Optimizing Force Field



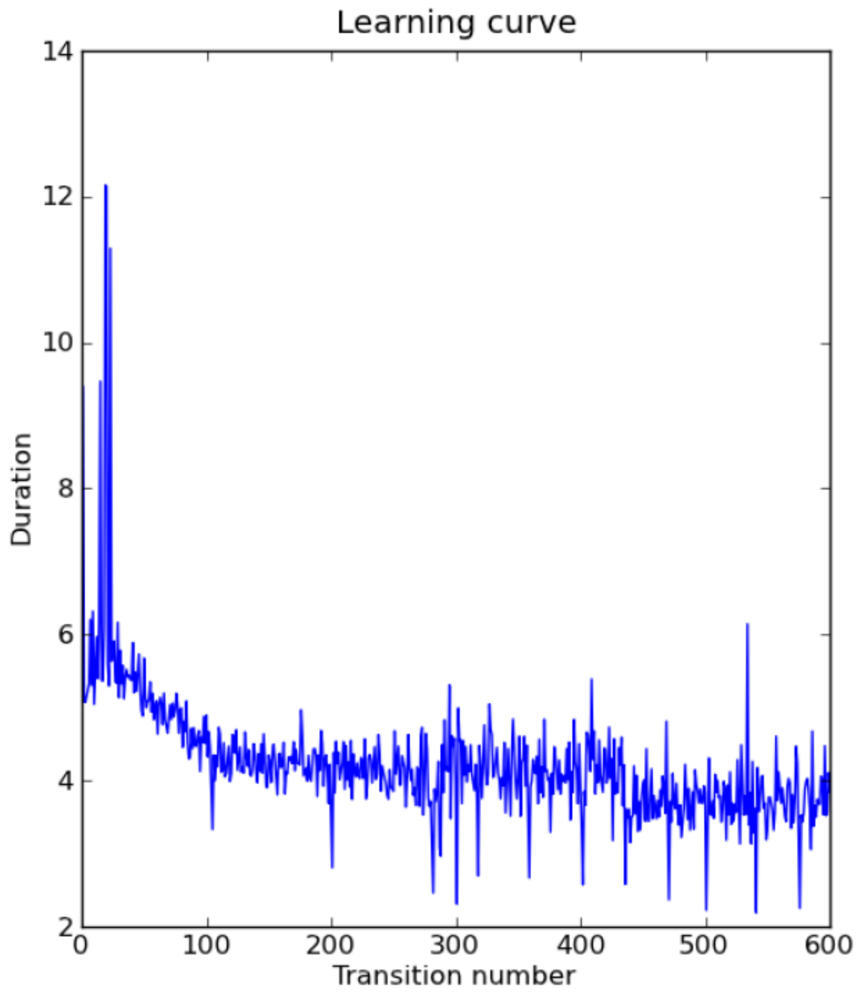


The force field can be optimized by value iteration of positive and negative experiences:

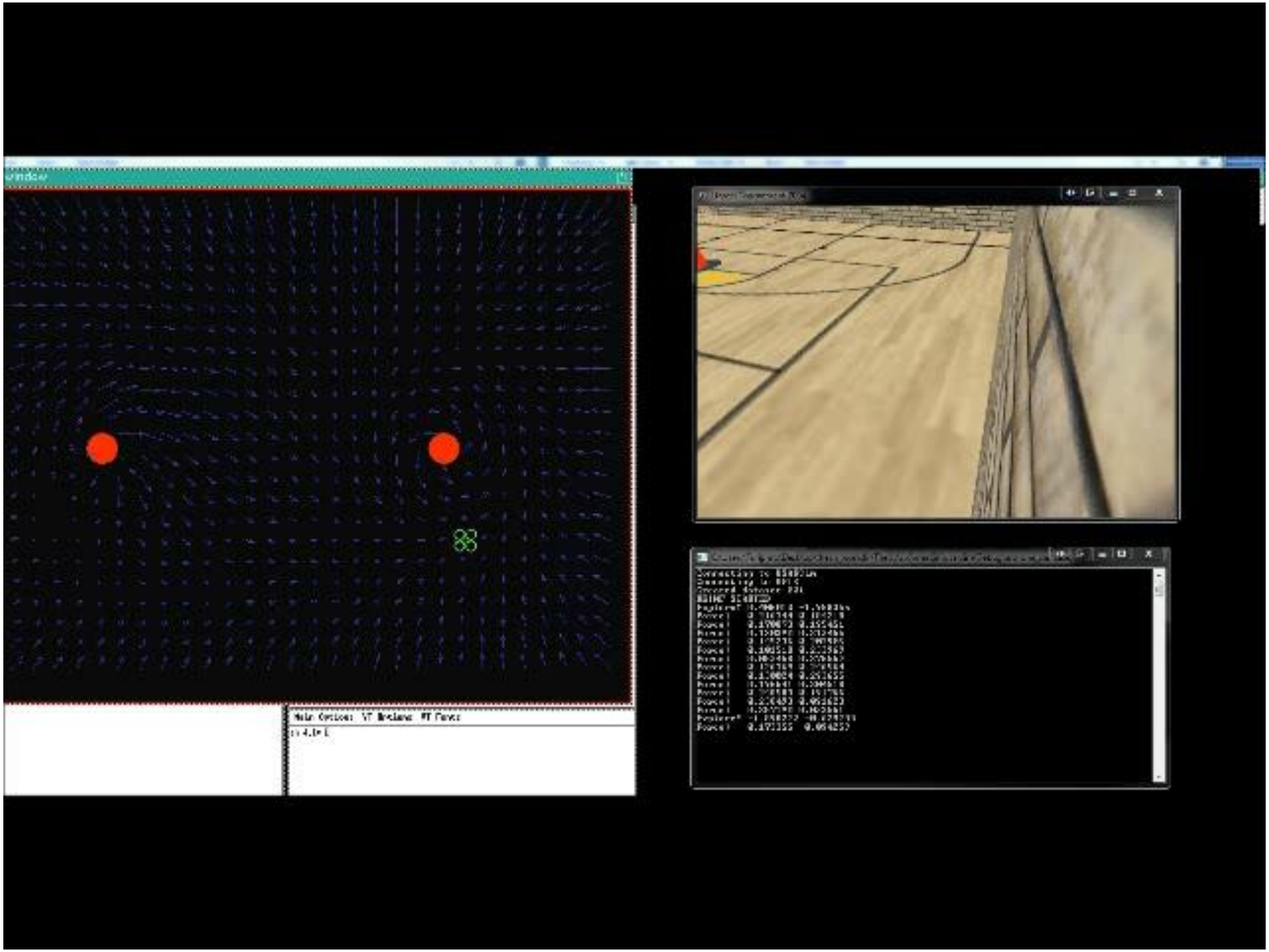
$$V(\vec{F}_{p-1}) \leftarrow (1-\alpha)V(\vec{F}_{p-1}) + \alpha(r(\vec{F}_p) + \gamma V(\vec{F}_p))$$

Improvement

The initial force field is already quite good, but can be further optimized.



Result



Conclusion

- Autonomous navigation with micro UAV is possible under favorable circumstances.
- Lack of features and the characteristics of the onboard camera make it a real challenge.
- The International Micro Air Vehicle flight competition becomes a benchmark for autonomous cooperating robot teams.

Universiteit van Amsterdam
Maneki-Neko Team

Universiteit van Amsterdam
Maneki-Neko Team