

Persistent self-supervised learning principle: from stereo to monocular vision for obstacle avoidance

by Kevin van Hecke, Guido de Croon, Laurens van der Maaten,
Daniel Hennes, and Dario Izzo

presented by Mike Kalaitzakis

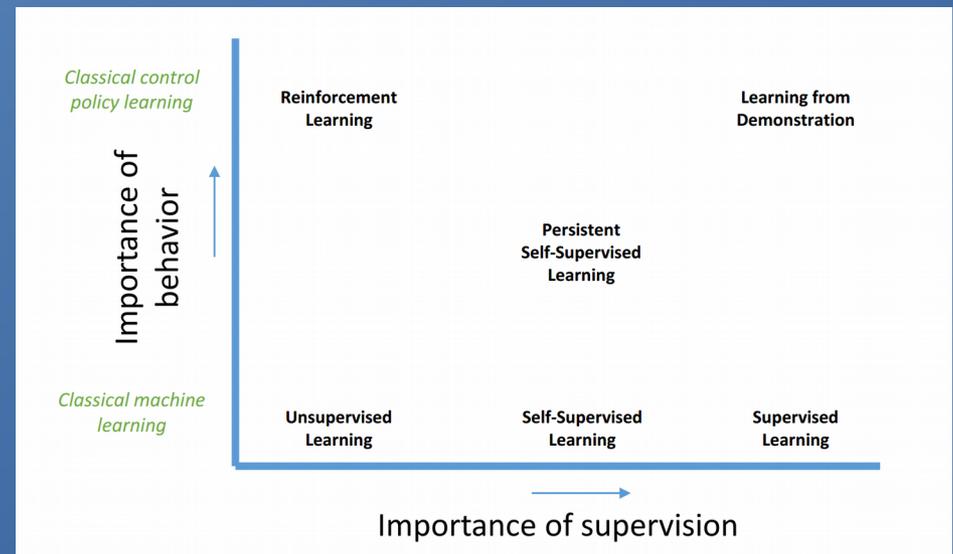
What is Persistent Self-Supervised Learning?

- What is Self-Supervised Learning
- What makes this persistent
- Differences and Similarities between other ML methods
- What are the advantages



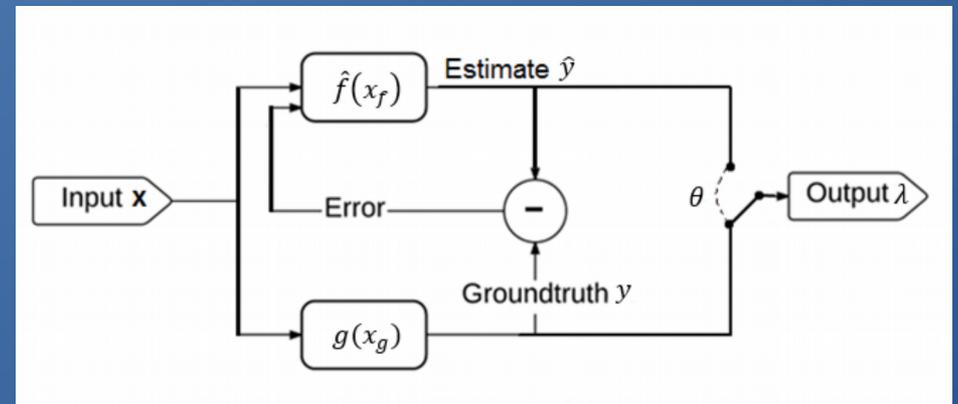
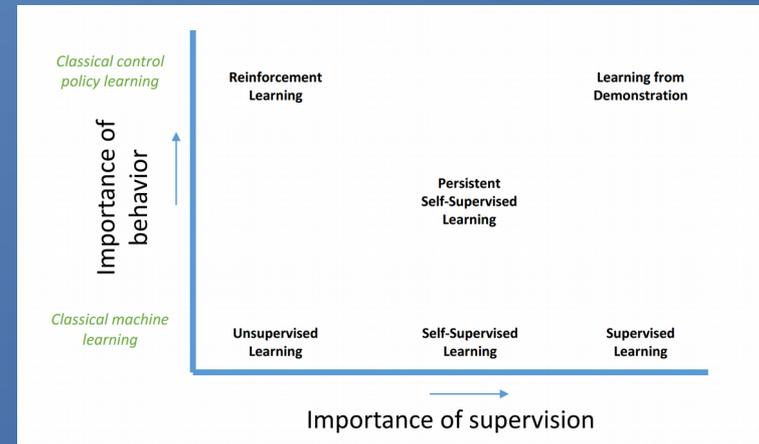
What is Persistent Self-Supervised Learning?

- **Self-Supervised Learning** is a mechanism that uses a trusted sensor cue for training to recognize a complementary sensor cue



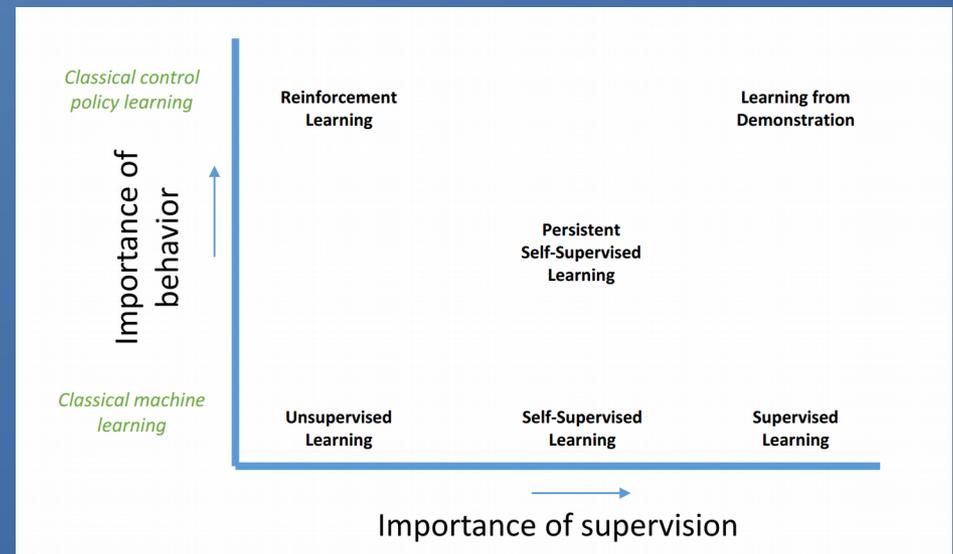
What is Persistent Self-Supervised Learning?

- **Persistent Self-Supervised Learning** is a Self-Supervised Learning method where the goal is to be able to replace the original trusted sensor cue if necessary



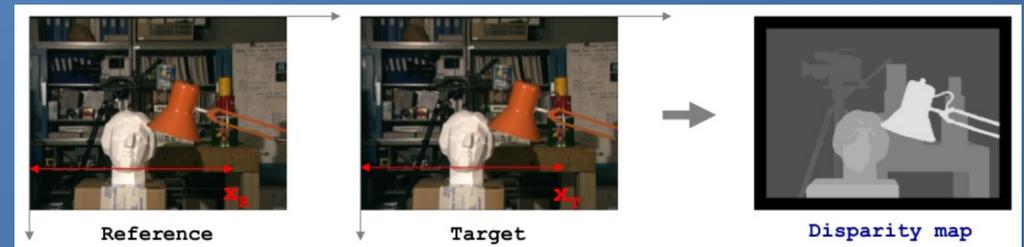
What is Persistent Self-Supervised Learning?

- Shares some similarities with Unsupervised Learning (UL) and Learning from Demonstration (LfD)
- Does not need labeled data like UL
- Does not need an external “teacher” as LfD
- Does not need a reward function and many trial-and-error sessions as Reinforcement Learning
- Can autonomously decide which sensor cue to use or if re-training is needed

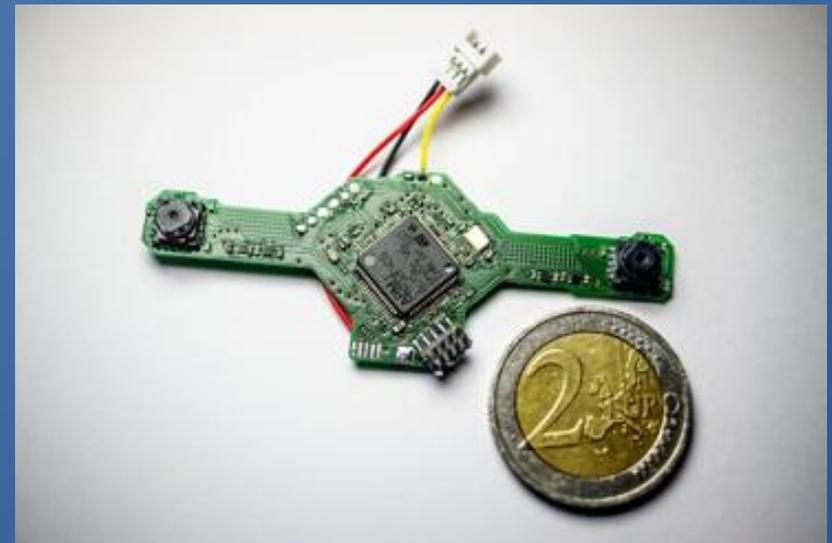


Going from Stereo Vision to Monocular Vision

- The disparity is the difference between the same image feature in the two images of a calibrated stereo pair
- It is used to calculate the distance of an object from the cameras
- A 4-gram custom made stereo camera is used (128x96 pixel, 10fps)
- The PSSSL tries to estimate the average disparity and not the whole map



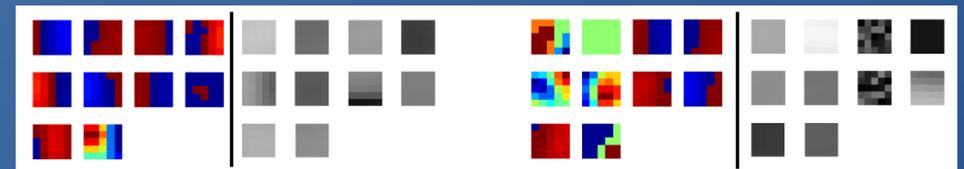
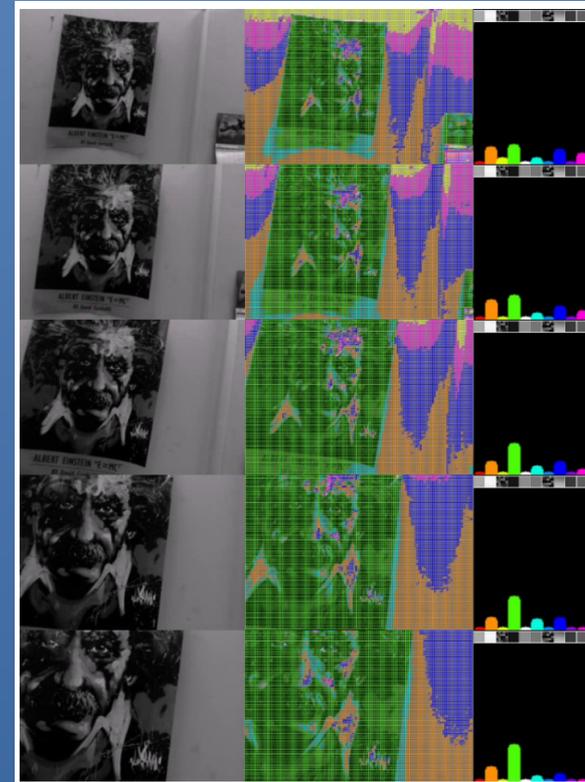
Stefano Mattocchia, University of Bologna



TU Delft, MAV-Lab

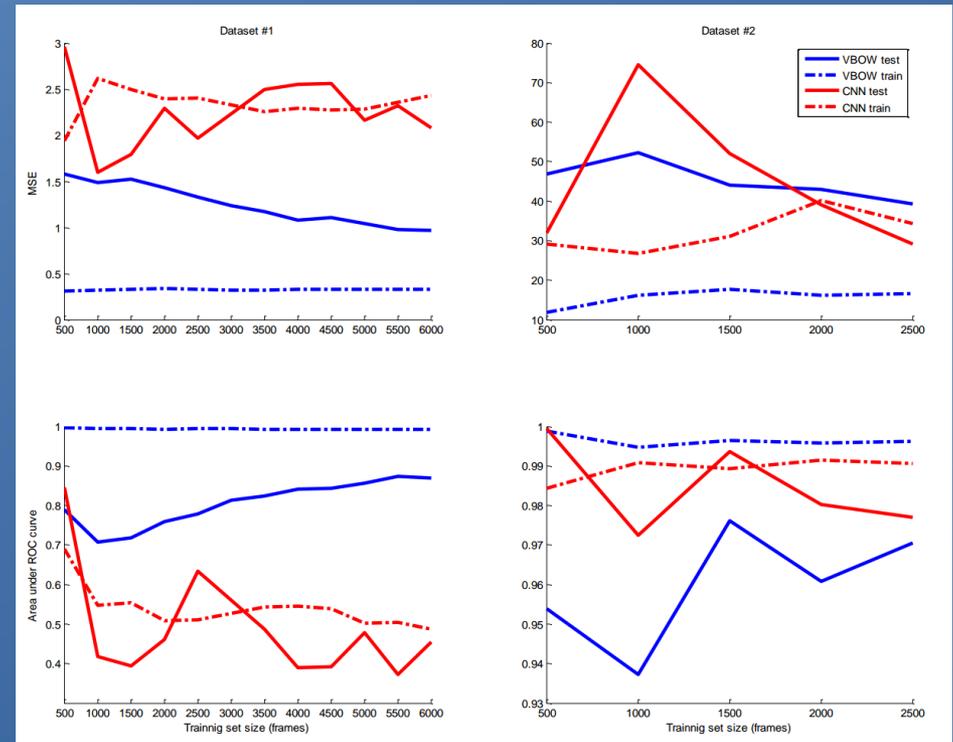
Visual Bag of Words (VBoW) and Textons

- In VboW a complex image is split in small image patches
- A dictionary is created by clustering the patches called textons
- When processing an image each patch is compared to the dictionary creating a texton occurrence histogram
- Both intensity and gradient textons are used



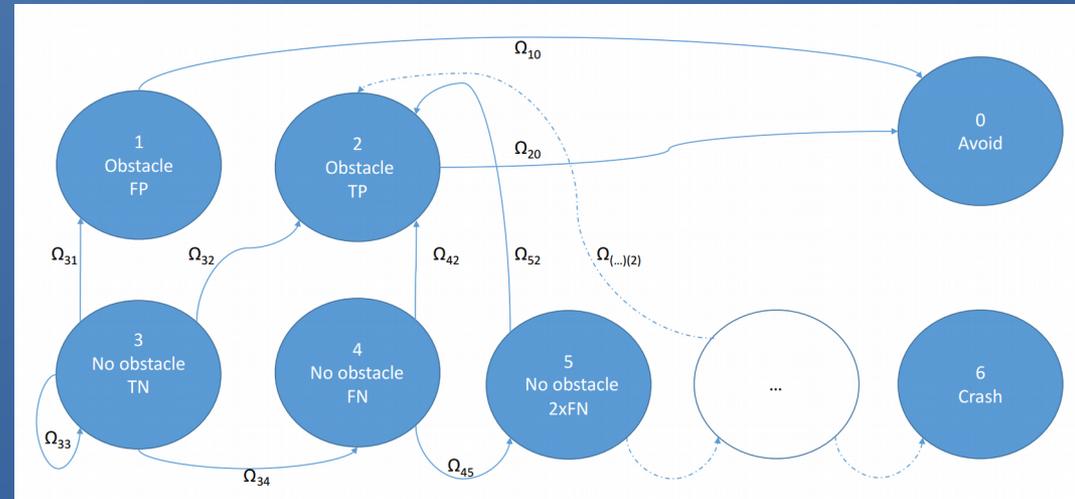
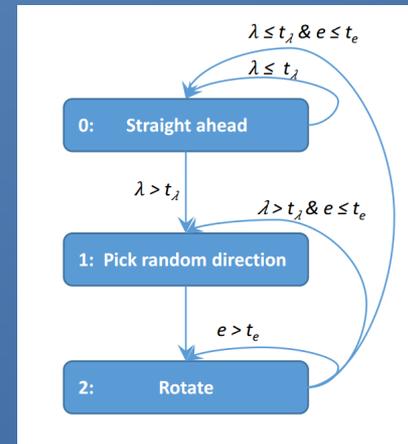
Why Visual Bag of Words?

- Short Answer:
Computationally efficient and limited computational power platform (Parrot AR Drone 2, 1GHz ARM cortex A8, 128MB RAM)
- Compared to the best available CNN the results were comparable and depended on the dataset

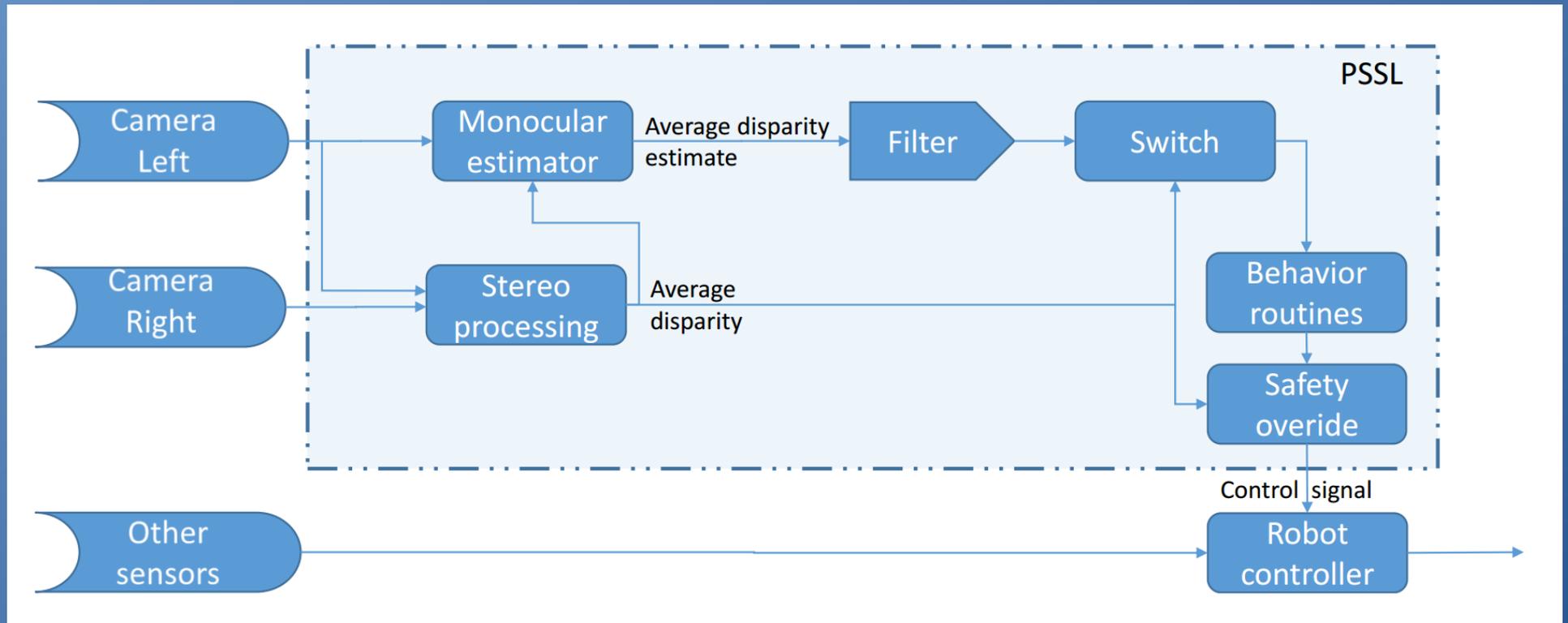


System Overview

- Binary classification
- Determine the threshold (t_λ) from the ROC curve
- The threshold is chosen to minimize the probability of collision
- Since the data cannot be assumed identically and independently distributed a Markov process was used to model the system
- **PSSL limitation factor:** By design the system will encounter very few occasions where obstacles are detected

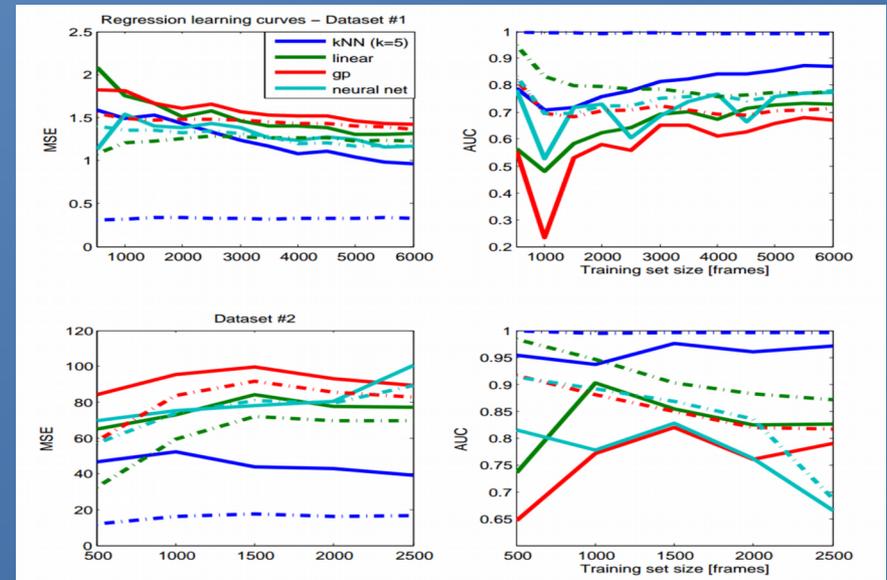


System Overview

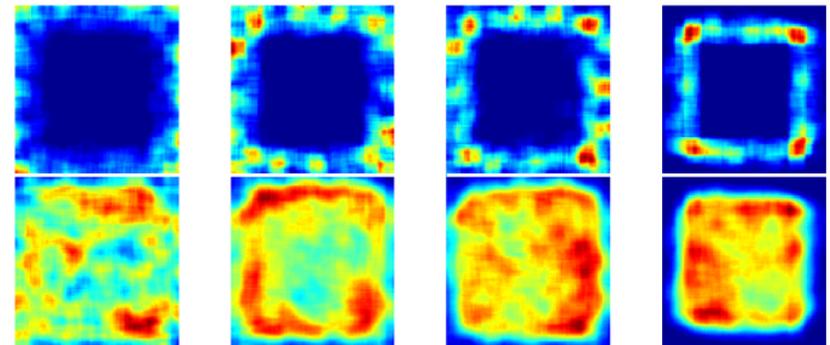


Simulation Results

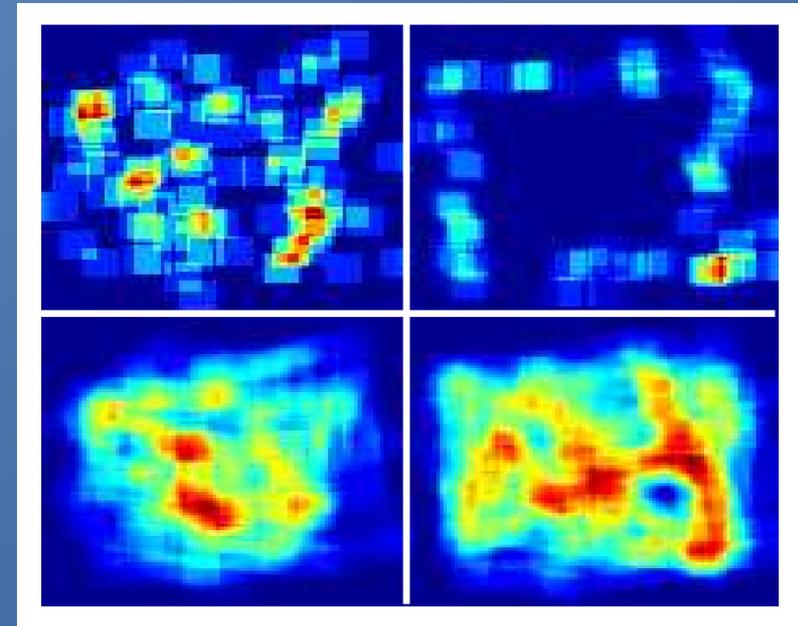
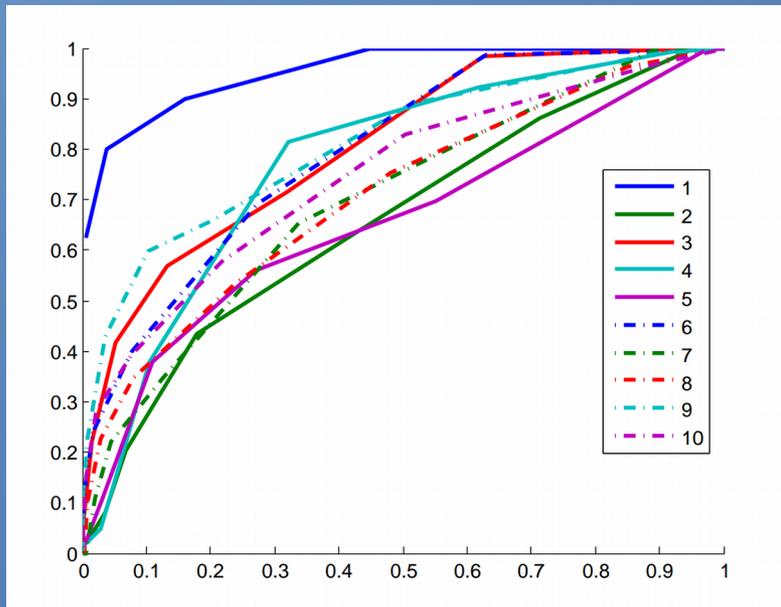
- SmartUAV simulation environment was used to choose regression method and learning scheme
- kNN, Linear regression, Gaussian process and Neural Networks were tested
- “Cold Turkey”, DAgger and “Training wheels” approaches to learning were tested



Method	Overrides	Turns
Pure stereo	N/A	45.6 ($\sigma = 3.0$)
1. Cold turkey	25.1 ($\sigma = 8.2$)	42.8 ($\sigma = 3.7$)
2. DAgger	10.7 ($\sigma = 5.3$)	41.4 ($\sigma = 3.2$)
3. Training wheels	4.3 ($\sigma = 2.6$)	40.4 ($\sigma = 2.6$)



Experiment Results



Description	Room 1					Room 2					Avg.
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	
Stereo flight time m:ss	6:48	7:53	2:13	3:30	4:45	4:39	4:56	5:12	4:58	5:01	4:59
Mono flight time m:ss	3:44	8:17	6:45	7:25	4:54	10:07	4:46	9:51	5:23	5:12	6:39
Mean Square Error	0,7	1,96	1,12	0,95	0,83	0,95	0,87	1,32	1,16	1,06	1,09
False Positive Rate	0,16	0,18	0,13	0,11	0,11	0,08	0,13	0,08	0,1	0,08	0,11
True Positive Rate	0,9	0,44	0,57	0,38	0,38	0,4	0,35	0,35	0,6	0,39	0,47
Stereo approaches	29	31	8	14	19	22	22	19	20	21	20,5
Mono approaches	10	21	20	25	14	33	15	28	18	15	19,9
Auto-overrides	0	6	2	2	1	5	2	7	3	2	3
Overrides ratio	0	0.72	0.3	0.27	0.2	0.49	0.42	0.71	0.56	0.38	0.41

