

SCOUT – SCANNING OUTSTANDINGLY

University of Dortmund

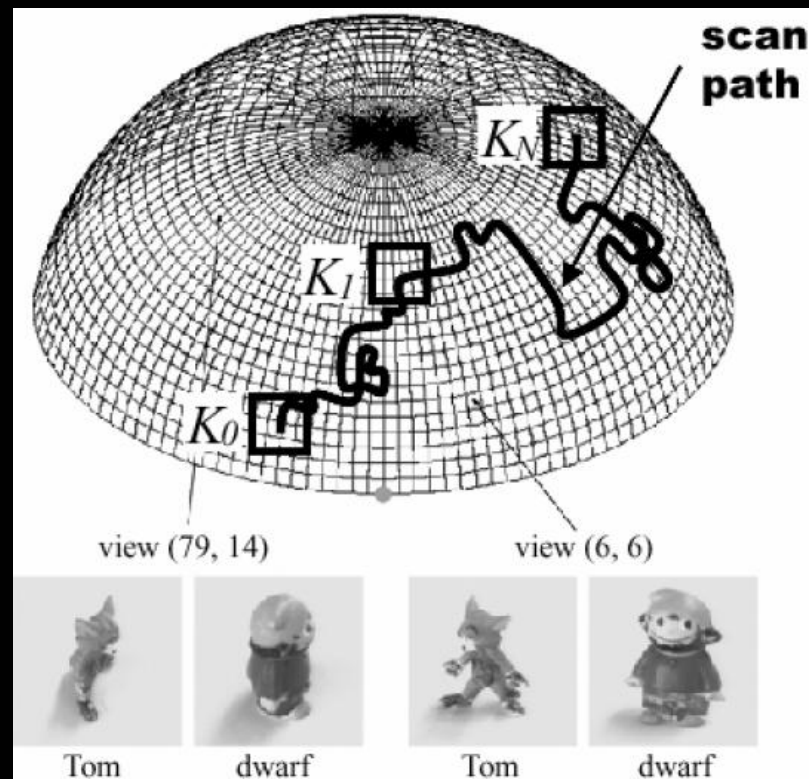
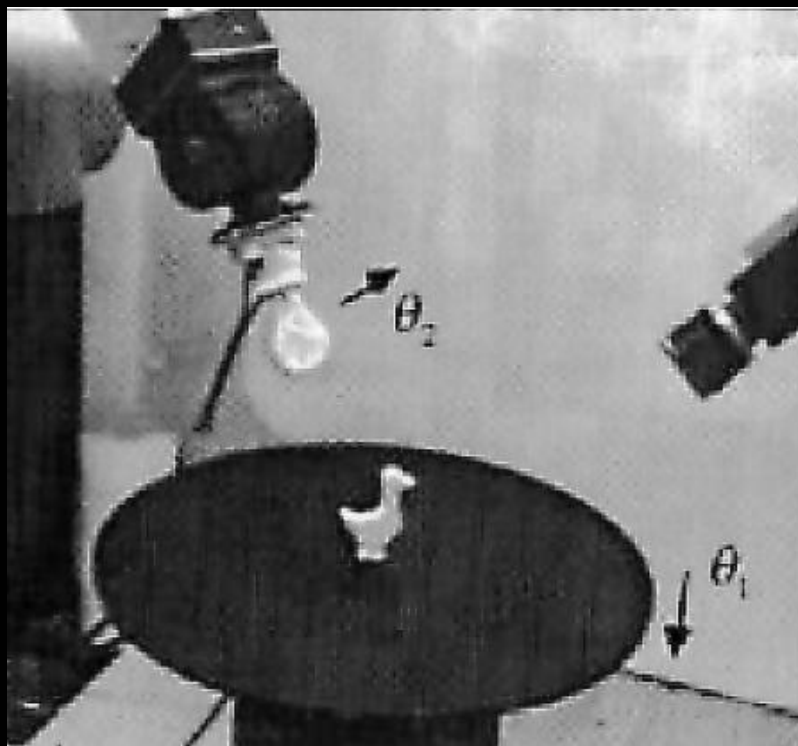
- Gabriele Peters
- Thomas Leopold
- Claus-Peter Alberts
- Markus Briese
- Sebastian Entian
- Christian Gabriel
- Zhiqiang Gao
- Alexander Klandt
- Jan Schultze
- Jeremias Spiegel
- Jürgen Thyen
- Martina Vaupel
- Peter Voß
- Qing Zhu



THE TASK (1-2)

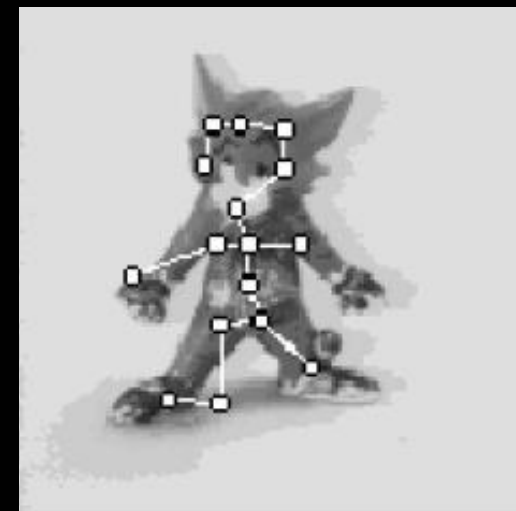
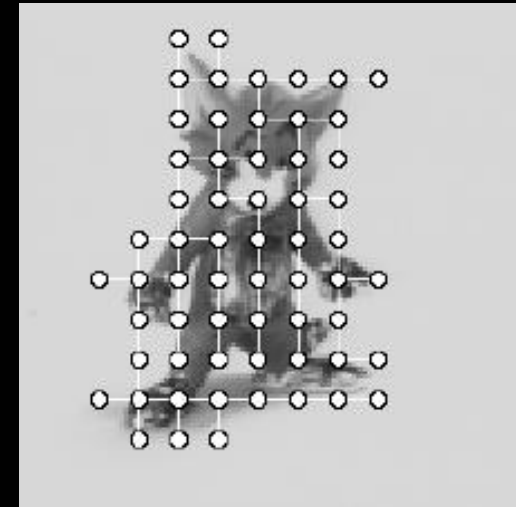
- 2d visual object representation
- up to now: Data acquisition and processing are separated.
- problem: Too many or too few data for the intended purpose are acquired.
- our goal:
 - acquisition of as many data as necessary, but as few as possible
 - in other words: Find a short scan path which acquires all necessary data to reconstruct non-visited views.

THE TASK (2-2)



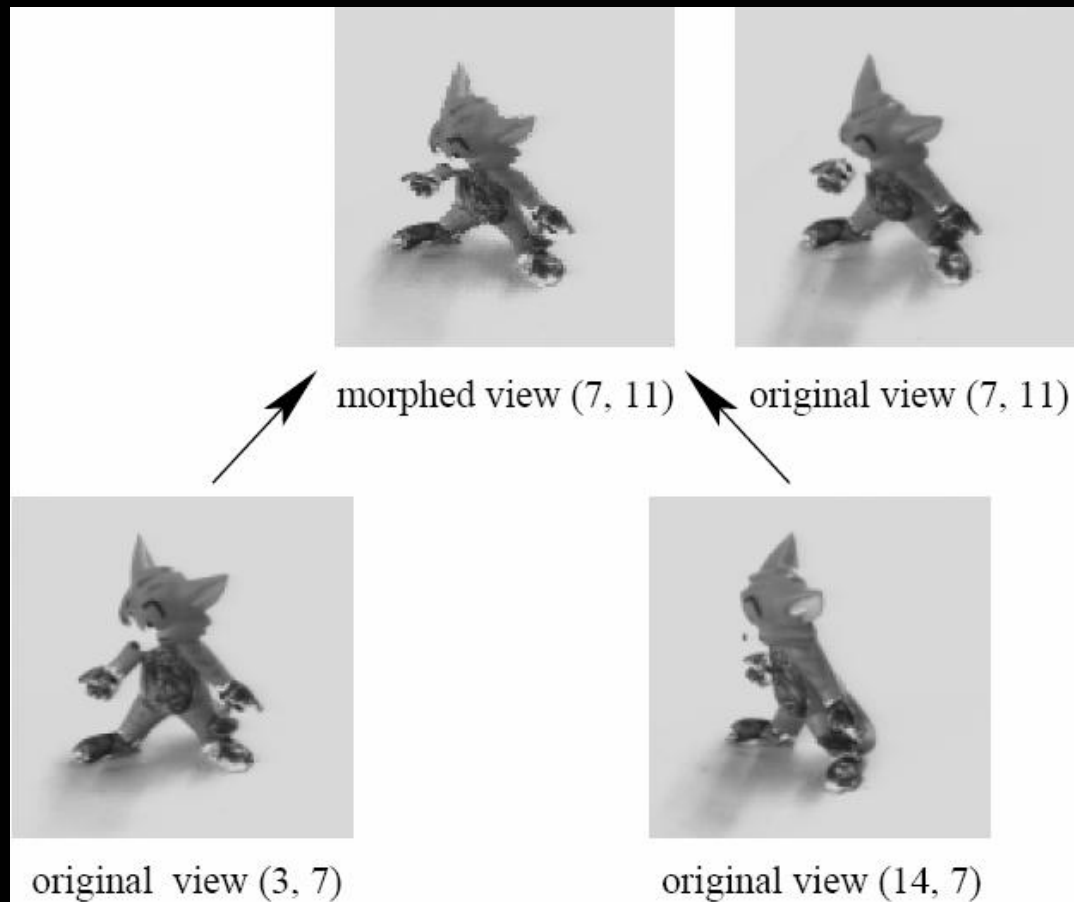
THE COMPONENTS OF OUR SYSTEM (1-2)

- data base (100 * 25 images/ views)
- view representation:
 - regular grid graph
 - each node is labeled with a feature vector (Gabor wavelet transform)
- tracking:
 - finds corresponding points in two neighboring views
 - uses feature vectors



THE COMPONENTS OF OUR SYSTEM (2-2)

- morphing:
 - reconstructs non-visited views
 - uses tracking
- core: RL-module:
 - finds scan path for object representation



REINFORCEMENT LEARNING: BASICS (1-2)

- agent and environment
- agent acts in this environment:
 - environment is in state s
 - agent chooses and performs an action a
 - environment changes its state to state s'
 - agent is rewarded for performing a in s
 - this (called *step*) repeats until the environment has reached a terminal state
- a sequence of such steps is called an *episode*

REINFORCEMENT LEARNING: BASICS (2-2)

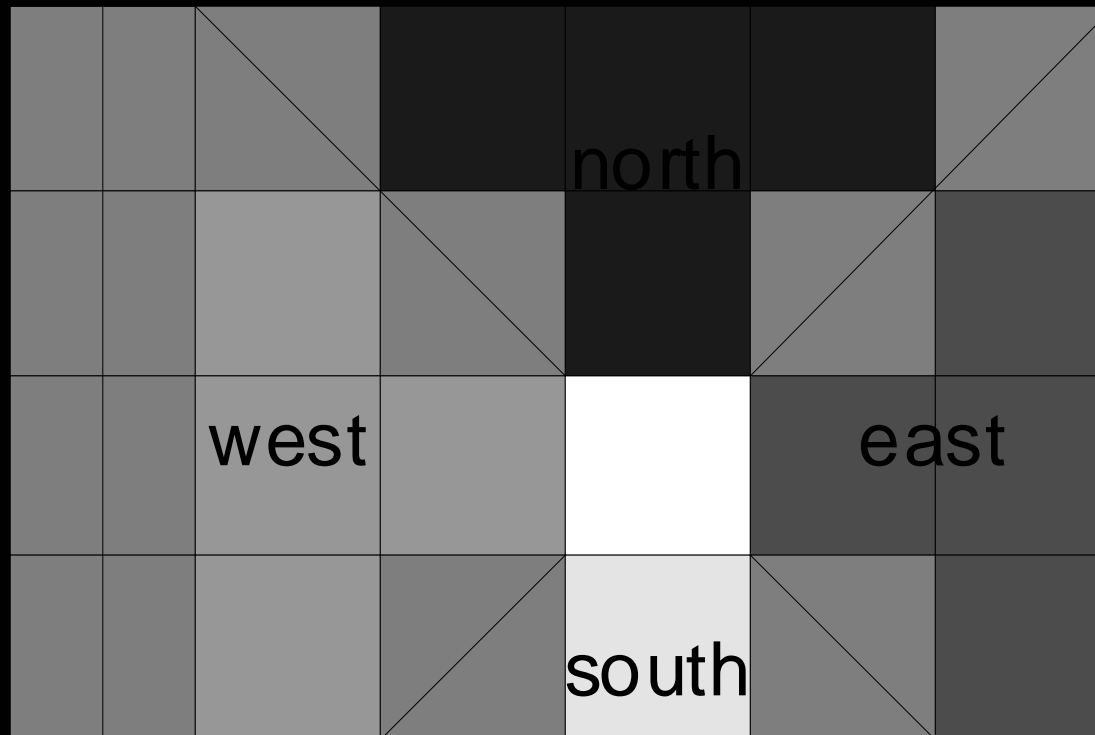
- goal of the agent: Maximize the sum of all rewards acquired during one episode.
- learns and improves by running many episodes
- advantage of RL:
 - Just define what the problem is (i. e., when it is solved) and not how to solve it.
 - Let the agent learn how to solve it: No human creativity is needed to find an algorithm that solves the problem.

OUR GOAL

- Find a shortest scan path which enables the reconstruction of any view with a quality of, e. g., 95 %.
- using Q-Learning (special kind of RL)

OUR AGENT: EXPLORER 1

- actions: go north, east, south or west
- state contains:
 - current position (x-, y-coordinates)
 - 4 'unexploration' values (1 for every direction)



OUR AGENT: EXPLORER 1

- actions: go north, east, south or west
- state contains:
 - current position (x-, y-coordinates)
 - 4 'unexploration' values (1 for every direction)

1	1	1	1	1	1
←					
1	1	1	1	1	
1					

vs.

		▲	1	2	1
			1	2	1
			1	2	1
			1	2	1

OUR AGENT: EXPLORER 1

- actions: go north, east, south or west
- state contains:
 - current position (x-, y-coordinates)
 - 4 'unexploration' values (1 for every direction)
- reward:
 - Explorer 1 predicts the view of the new position based on the previously perceived views
 - compares it to the actually perceived view
 - the larger the difference, the higher the reward
 - reward between -1 (bad) and 0 (good)

RESULTS

- 16 possible values for 'unexploration'
- $\alpha = 1/3$; $\varepsilon = 1/3$ and slowly annealing
- convergence: after $\approx 1,000,000$ episodes
- random path (32 steps):
- learned path (32 steps):



RESULTS

- 16 possible values for 'unexploration'
- $\alpha = 1/3$; $\varepsilon = 1/3$ and slowly annealing
- convergence: after $\approx 1,000,000$ episodes
- random path (32 steps):
 - average total reward: -24.6
 - total error: 9.2 % (calculated from 25 random views)
- learned path (32 steps):

RESULTS

- 16 possible values for 'unexploration'
- $\alpha = 1/3$; $\varepsilon = 1/3$ and slowly annealing
- convergence: after $\approx 1,000,000$ episodes
- random path (32 steps):
 - average total reward: -24.6
 - total error: 9.2 % (calculated from 25 random views)
- learned path (32 steps):



RESULTS

- 16 possible values for 'unexploration'
- $\alpha = 1/3$; $\varepsilon = 1/3$ and slowly annealing
- convergence: after $\approx 1,000,000$ episodes
- random path (32 steps):
 - average total reward: -24.6
 - total error: 9.2 % (calculated from 25 random views)
- learned path (32 steps):
 - total reward: -13.6
 - total error: 5.7 % (calculated from 25 random views)