DRAFT PAPER 30.08.97, 19:10

A Taxonomy of Spatial Knowledge for Navigation and its Application to the Bremen Autonomous Wheelchair*

Bernd Krieg-Brückner[1], Thomas Röfer[1], Hans-Otto Carmesin[2], Rolf Müller[1]

[1]Bremer Institut für Sichere Systeme, TZI, FB3; [2]FB1
Universität Bremen, Postfach 330440, D - 28334 Bremen

Abstract. A taxonomy is described that relates different navigational behaviours in a hierarchical and compositional way. Elementary navigation tactics are combined to tactical navigation in routes; landmarks in space are contrasted to routemarks in networks of passages. Survey knowledge comes in at the level of strategic navigation. The Bremen Autonomous Wheelchair is then presented as a vehicle for experimentation in robotics, both to model biologically plausible navigational behaviours and to develop efficient navigational mechanisms for a technical application. The implementation on the autonomous system is based on the use of basic behaviours and the identification of routemarks. The actual recognition of artificial routemarks is described and early results of the current work on the identification of natural 3D-marks are presented.

1 Introduction

Perception, Spatial Knowledge and Navigational Behaviour. Spatial knowledge, its representation and acquisition are intimately related with perceptual information on the one hand, and intended navigational behaviour on the other. In fact, there is a ternary relation between sensory performance, behavioural performance and spatial knowledge, cf. the “Spatial Cognition Triangle” in fig. 1. When analysing a specific animal or human behaviour, sensory and behavioural performance are given and we are trying to model the mechanisms for spatial cognition. When designing an autonomous robot, the task may be to develop navigation techniques for given sensory equipment and desired behavioural performance; alternatively, we may try to develop the best sensor to achieve optimal behaviour, or we may ask, for given techniques and desired behaviour, what minimal sensory equipment might do.

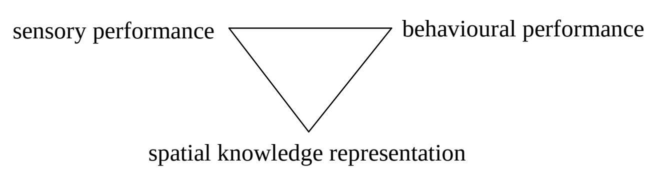


Fig. 1. The Spatial Cognition Triangle

* This work has been supported by the Deutsche Forschungsgemeinschaft in the priority programme Spatial Cognition.

Objectives of the Taxonomy. The taxonomy presented in this contribution relates navigational behaviours, i.e. observable behaviour associated to conjectured spatial knowledge representations, where particular sensory equipment is given. In the context of the interdisciplinary DFG priority programme on Spatial Cognition, it is intended to serve several purposes: a common framework for

1. theoretical modelling and reasoning unifying concepts and terminology of biology, psychology, artificial intelligence and robotics,
2. empirical investigations, e.g. about navigation performance and conjectured mental representations in animals or humans, and
3. experimentation with robots.

Two recent workshops at Göttingen (Werner et al. 1997) and Berlin (Schmid et al. 1997) have demonstrated that the essential navigational issues are the same, although animal, human and robot navigation differ substantially in many respects. Psychologists and biologists are concerned with understanding the mechanisms that enable humans and other animals to navigate, while the eventual objective in robotics research is to develop navigational skills for technical applications. Empirical evaluation focuses on efficiency and robustness of implemented mechanisms whereas empirical data is used in biological and psychological research to infer the underlying processes and mental representations.

Thus the synthetic and the analytic approaches complement one another: biological systems can be used as a guide for developing robots to perform similar tasks; moreover, synthetic approaches can be used to model biologically plausible navigational behaviours, isolating specific aspects and test hypotheses generated in biological and psychological research. On the other hand, robotics research may generate questions to be empirically investigated in the complex environment of biological systems. Synthetic approaches make the technical problems, representations and algorithms explicit that are associated with particular navigational behaviours; thus psychological theories can be restricted to computationally and biologically plausible models.

The Bremen Autonomous Wheelchair. After presenting the taxonomy, it is applied to robot navigation. We present the Bremen Autonomous Wheelchair as a vehicle for experimentation, both to model biologically plausible navigational behaviours and to develop efficient navigational mechanisms for a technical application.

2 A Taxonomy of Spatial Knowledge for Navigation

2.1 Overview

Spatial knowledge about an environment can be separated into location, route and survey knowledge. A *location* is often characterised by a view of the surroundings from this position (Schweizer and Janzen 1996, Franz et al. 1997a, 1997b, Schölkopf and Mallot 1995), also sometimes called a “snapshot” (Cartwright and Collett 1983). These views are described as consisting of a constellation of landmarks (Collet and Kelber 1988) or cues (Poucet 1993) that identify the location. A *route* is often identified with a sequence of locations or views (Schweizer and Janzen 1996). *Survey* *knowledge* is obtained as an abstraction and integration of specific routes. The mental representation of survey knowledge is referred to as “cognitive map” (Sholl 1987, Poucet 1993); e.g. in insect research, there has been a long discussion if insects such as honeybees have a cognitive map (Gould 1986) or not (Wehner and Menzel 1990, Dyer 1991).

The taxonomy keeps essentially the same levels in a hierarchy of navigational be-

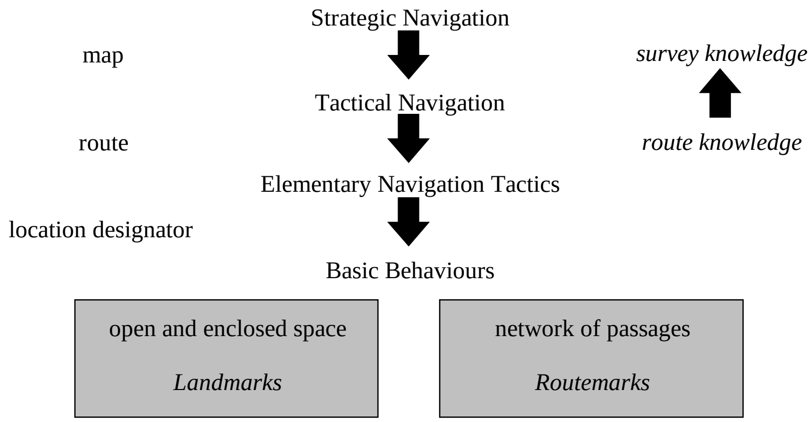


Fig. 2. Hierarchy of navigation behaviours

Basic Behaviours	**Elementary Tactics**	**Tactical Navigation**
approaching target	*searching*	*route navigation*
• reaching target	• spiralling	• homogeneous
• approaching target	• meandering	• heterogeneous
• stopping	• quarter searching	*explorative navigation*
basic behaviour in space	*vectorial navigation in space*	• path finding • threading
• course following	• directional navigation	*combined tactical navigation*
• docking at target	• dead reckoning	
	positional navigation in space	• positional path finding • traversing
	• landmark navigation	
	• celestial navigation	**Strategic Navigation**
basic behaviour in network of passages	*branching in networks of passages*	*alternative route navigation*
• passage following	• binary branching	• making a detour
• bearing {left | right}	• rectangular branching	• making a shortcut
• turning into a passage	• directional branching	*premeditated navigation*
	• n-way branching	• route map navigation
	• designated branching	• map navigation

Tab. 1. Overview of navigation behaviours

haviours (cf. fig. 2): starting from the top, *strategic navigation*, employing survey knowledge, is mapped to *tactical navigation* with routes; *elementary tactics* use *basic behaviours*. The classification leads to a fine-grain distinction of navigational behaviours (cf. tab. 1); it should be emphasised that any actual agent—such as an animal—is likely to use a combination in order to increase robustness.

Landmarks and routemarks are distinguished; the actual objects in view may be the same, but their tactical use is different. *Landmarks* are used to determine the position of an agent in enclosed or open space. From one distant landmark, e.g. a church squire, an agent can determine its rotation but it needs at least three landmarks to exactly determine its position in 2D-space. *Routemarks* are used to determine an agent's position along a route. As routes are only one-dimensional, a single routemark might sufficiently characterise the location of the agent.

2.2 Basic Behaviours

At the level of basic behaviours and elementary navigation tactics (cf. tab. 2), two settings should be distinguished: *space,* either open or enclosed, and *networks of passages*.

Title	***Percept***	***Representation***	***Action***
approaching target			
reaching target	remaining distance	distance	when remaining distance near Zero, trigger new behaviour
heading for target	view (of target)	designated view	when view corresponds to designated view, trigger new behaviour
stopping			stop [at target]
basic behaviour in (enclosed or open) space			
course following	course; orientation	course (direction)	adjust orientation to course, steer clear of obstacles
docking at target	position, orientation relative to target	designated target position, orientation	manoeuvre into target position and orientation
basic behaviour in network of passages (tunnels, corridors, roads, trails)			
passage following	walls, obstacles		follow passage centred between walls, avoiding obstacles
wall following {left | right}	wall, corners, obstacles	{left | right}	follow {left | right} wall (around corners), avoiding obstacles
turning into a passage	junction with n branches	branch designator	turn into designated passage

Tab. 2. Basic navigation behaviours

For example, basic behaviours in a passage, e.g. a tunnel, a corridor, a road, a trail, may be *wall-following*, *passage-following*, i.e. centred between walls while avoiding obstacles (Nehmzow 1995), *turning* into a designated passage at a junction, etc. Basic behaviours are atomic tactics, the simplest to be investigated—or realised in a robot. They are used in more complex elementary tactics.

2.3 Elementary Navigation Tactics

Elementary navigation tactics in space (cf. tab. 3) include *directional navigation* guided by a compass (Wiltschko 1995, Collet and Baron 1994), *dead reckoning* (Gallistel 1990, Müller and Wehner 1988, Mittelstaedt and Mittelstaedt 1982) using a homing vector accumulated from self-movement. The major knowledge representation for such navigation tactics in space is a *target vector* (direction and distance) that may have been derived from a map, i.e. survey knowledge, learned from a previous experience or communicated from other agents, e.g. for honeybees via the bee lan-

searching			
Title	***Percept***	***Representation***	***Action***
spiralling	locality in space	designated view	spiral outwards, heading for target
meandering	locality in space	designated view	meander, heading for target
vectorial navigation			
directional navigation	(compass) direction; orientation, elapsed time, speed	target vector	compute course, distance to target; follow course, approach target
dead reckoning	homing vector; self-movement	target vector	compute course, distance to target; follow course, approach target
positional navigation			
landmark navigation	view of landmarks, orientation	target position in view of landmarks	triangulate vector to target, navigate vectorially
celestial navigation	view of moving celestial bodies, orientation	target position in view or co-ordinate system	triangulate vector to target, navigate vectorially

Tab. 3. Elementary navigation tactics in space

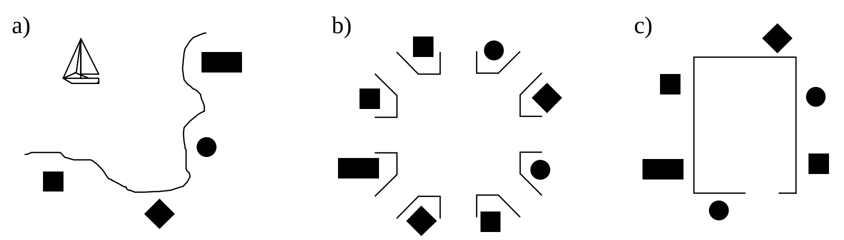


Fig. 3. Landmarks in open and enclosed space: a) Ocean. b) Town square. c) Room.

guage (v. Frisch 1967), transposed from a vertical 2D-notation in the hive to a horizontal situation in space. During actual navigation, the course is then computed relative to a compass direction, e.g. a magnetic compass, the polarisation of light (Rossel and Wehner 1986) in a hazy sky, or the direction of one particular landmark such as the sun.

Alternatively, the representation may be a particular configuration of several *global landmarks*, i.e. a *view*, used to determine a relative target position by triangulation; thus, a *view* from a particular location corresponds to a particular configuration of landmarks. Honeybees compare the original, learned view to the present view in order to determine the vector for “homing in” (cf. fig. 5), both from a defined direction (Cartwright and Collet 1982, 1983, 1987, Collet et al. 1986). In *landmark navigation*, these landmarks are fixed, e.g. prominent buildings, trees, lighthouses. In *celestial navigation* (Wehner 1983), landmarks are moving over the day, e.g. the sun, and over the year, e.g. the stars, and these movements have to be compensated for by

Title	***Percept***	***Representation***	***Action***
searching			
quarter searching	locality in rectangular grid of passages	designated view	search each block in vicinity by spiralling or meandering
branching			
binary branching	bifurcation	{left | right}	bear {left | right}
rectangular branching	intersection in rectangular grid	{left | straight | right}	follow passage approaching and across junction | bear {left | right}
directional branching	intersection in directional grid, direction	{N | E | S | W}	follow passage approaching and across junction | bear {left | right}
n-way branching	n-way junction	i	follow passage approaching junction; turn into i-th branch
designated branching	view of n-way junction	[view,] branch designator	follow passage approaching junction; turn into designated branch

Tab. 4. Elementary navigation tactics in networks of passages

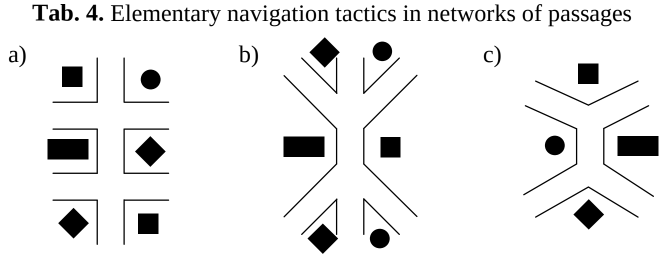


Fig. 4. Routemarks in networks of passages. a) Grid. b) Graph. c) “Hexatown” (Gillner and Mallot 1997).

an internal clock (circadian rhythm).

It is interesting to note that navigation tactics in space seem to be equally applicable to open or enclosed space, e.g. an ocean, a town square, a room (cf. fig. 3).

Elementary navigation tactics in passages (cf. tab. 4) can take advantage of the limited alternatives within a passage, e.g. *binary* or *n-way branching.* A junction constitutes a *decision point* for the target passage and the switch to a new behaviour. Characteristic *views*, depending on the direction from which the junction was approached, may be associated with the decision of selecting a particular target passage (Franz et al. 1997a, Schölkopf and Mallot 1995). Such a view may be characterised by *local landmarks* or *routemarks* (cf. fig. 4) such as prominent visual objects, odours, sounds or tactile percepts, as long as they are stable, fixed and persistent.

Task Dependency. Elementary navigation steps are highly task dependent. This can be modelled as a pair **(tactic, location designator)**, where the **location designator** captures the spatial knowledge needed to instantiate the general tactic and determines a tactical decision. Thus the tactic represents the procedural knowledge and the location designator the corresponding data. The actual representation of a location designator and its reference system depends on the particular tactic. For example, for navigation in space, a vector to the target relative to the present position and orientation, or a vector in a global co-ordinate system, could be used; for branching, a particular view or branch designator would suffice.

2.4 Tactical Navigation along Routes

Generalised Routes. So far, most authors regard a route as a sequence of locations; here, this corresponds to a *homogeneous route* **(tactic, ⟨location designator⟩)** where the same tactic is applied. As a generalisation, a *heterogeneous route* consists of a sequence of arbitrary navigation steps: **⟨(tactic, location designator)⟩**.

Tactical Navigation. Different tactics may be concatenated in *heterogeneous route navigation* (cf. fig. 9): consider e.g. the honeybee heading for a feeding area with a target vector communicated from a sister (v. Frisch 1967), then foraging by searching along a route that is learned and remembered for next time, thereafter returning to the vicinity of the hive with an accumulated homing vector by dead reckoning, followed by searching for the hive, and finally homing in by landmark navigation based on a

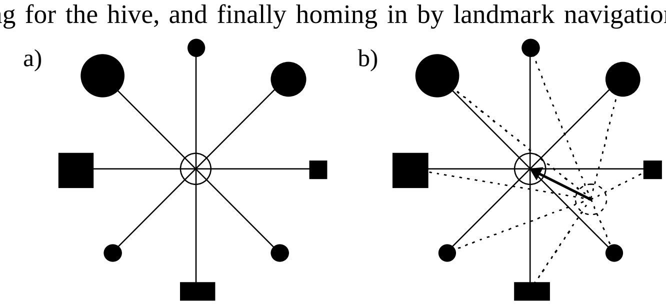


Fig. 5. Landmark navigation of honeybees. a) The learned landmark constellation. b) Calculating the homing direction from deviations.

view of a landmark constellation that had been acquired before leaving (Cartwright and Collet 1982, 1983, 1987, Wittmann 1995). Similarly, navigation in a *network of passages*, say roads, may be interspersed with landmark navigation across an enclosed *space* such as a large crowded city square.

Operations on Tactics. Some authors postulate an *inversion* operator on tactics in routes (Hermann et al. 97). This may be appropriate for humans, but is questionable for lower animals; it is definitely a problem for robots navigating with minimalistic tactics (cf. section 6.3). A better approach to allow backtracking is to overlay a kind of *threading* (cf. tab. 5) or computation of a *homing vector* (cf. fig. 5), as insects and other animals do, while moving forward. Moreover, different navigation tactics may be *overlaid* to achieve a tactical goal, e.g. vectorial navigation and explorative navigation such as path finding in a maze of passages. The list of combined tactics in tab. 5 is obviously incomplete and just states examples.

2.5 Location Abstraction and Strategic Navigation

Strategic navigation includes planning. The point of reference changes from a field perspective to that of an observer with *survey knowledge* (Herrmann et al. 1997). The

route navigation

Title	***Percept***	***Representation***	***Action***
homogeneous route navigation	{locations}	homogenous route: tactic, ⟨location designator⟩	apply tactic to location designators in sequence, till end of route; stop/dock at target
heterogeneous route navigation	{locations}	heterogeneous route: ⟨(tactic, location designator)⟩	(apply tactic to location designator) in sequence, till end of route
explorative navigation			
path finding	possibly obstructed space	target designator	navigate along passages towards designated target
threading	labyrinth of passages	target view	navigate along all passages, heading for target, constructing thread (inverse route), backtracking
combined tactical navigation			
positional path finding	possibly obstructed space	target designator	combine positional navigation (e.g. using GPS) and path finding
traversing	possibly obstructed space	target designator	combine vectorial/positional (e.g. landmark) navigation and explorative navigation around obstacles

Tab. 5. Tactical navigation

most basic form seems to be a combination of routes—in particular for navigation in passages—into a net or directed graph.

Location Abstraction and Routemarks. There are two ways in which routes may be combined by matching locations, thus regarding **location designator**s as equivalent: *source aliasing* and *target aliasing*. Either two target **location designator**s emanate from the same *source **location*** but denote different targets, or two target **location designator**s lead to the same *target **location*** but potentially emanate from different sources. Thus aliasing leads to a notion of (abstracted) **location** as a node in a *route graph*, in which the edges are labelled with respective navigation tactics.

Maps. A route graph can thus be generalised to a *map* with a set of **location abstraction**s as nodes and a set of **tactic abstraction**s as edges, i.e. **({tactic abstraction}, {location abstraction})**. Different tactical aspects of navigation may lead to different maps with different kinds of abstractions for tactic-oriented spatial knowledge contained in locations, e.g. by introducing topological or Cartesian relations (Barkowsky et al. 1997). This abstract information contained in a map (or an overlay of several maps) will then have to be sufficient for reconstructing routes and for planning shortcuts or detours around obstacles. A map need not be global, but may be used locally ("local chart") (Poucet 1993); the reference system may change according to the tactic applied.

Thus a simple (partial) route graph may do for *alternative route navigation* (cf. tab. 6) to plan for detours or shortcuts; possibly, additional quasi-survey knowledge about directions using a vectorial or positional navigation tactic (cf. tab. 3) is overlaid. For this representation, it may even be sufficient to stay in a field perspective. *Premeditated navigation*, on the other hand, requires a map as a first class object, trying

Title	***Percept***	***Representation***	***Action***
alternative route navigation			
making a detour	(temporarily) obstructed route	route graph	construct route to target of obstructed route via evasive location; navigate with alternative route
making a shortcut	"longer" route	route graph	construct "shorter/direct" route to target of "longer" route; navigate with alternative route
premeditated navigation			
route map navigation	route marked on map, {locations}	route on map	construct (concrete) route from abstract route; navigate with route
map navigation	source, target marked on map, {locations}	source, target marked on map	construct route map by abstract navigation on map; navigate with route map

Tab. 6. Strategic navigation

to invert the previous process of abstraction before navigating in the field—or even while going along.

3 The Bremen Autonomous Wheelchair

A wheelchair is used as an experimental robot platform in Bremen. It has four wheels. The front axle drives the wheelchair while the back axle is used for steering. Therefore, the wheelchair moves like a car driving backwards. It is equipped with a Pentium 100 computer, twelve bumpers, six infrared sensors, 16 ultrasonic sensors and a camera. The infrared sensors can only detect whether there is an obstacle within a radius of approximately 15 cm; however, they cannot measure the distance. Two different kinds of ultrasonic sensors are fitted to the wheelchair: half of the sensors have an opening-angle of 80° while the other half only measure in a range of 7°. In addition, the wheelchair can measure the rotations of its front wheels. Thus, it is able to perform dead reckoning.

The sensors are assigned to four control subsystems. Three of these systems are illustrated in fig. 7:

Collision Detection. The wheelchair uses all bumpers, all infrared sensors and the wide-angle ultrasonic sensors to detect collisions with the environment. As long as an obstacle is perceived by the infrared and ultrasonic sensors, respectively, the wheelchair is able to stop before physical contact is made.

Steering Restriction. As the wheelchair is steering with its back wheels, its rear swings out very heavily. To prevent it from colliding with obstacles on the side during driving manoeuvres, the distance to the closest obstacle is measured and the steering angle is reduced as much as necessary to avoid a collision.

Fig. 6. The Bremen Autonomous Wheelchair.

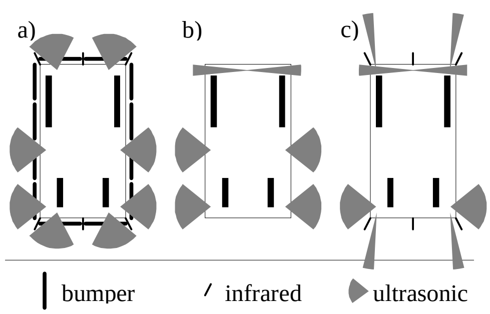


Fig. 7. The sensor control subsystems. a) Collision detection. b) Steering restriction. c) Navigation.

Navigation. Six of the narrow-angle ultrasonic sensors and the six infrared sensors are employed for navigation purposes. They have been chosen because their measurements do not only reflect a certain distance to an obstacle but also determine it in a definite direction. In contrast, the wide-angle opened ultrasonic sensors would not allow a precise localisation. At the moment, these sensors are supported by two wide-angle sensors to the side.

Landmark or Routemark Detection. The camera is used to scan the surroundings for landmarks or routemarks, respectively. It is mounted on a pan-tilt-head. Therefore, it can watch the environment independently from the current orientation of the wheelchair.

4 Navigation in Space using the Parti-Game Algorithm

The Parti-Game algorithm, originally developed by Moore and Atkeson (1995) was one of the first navigation algorithms implemented on the wheelchair. It is a navigation approach that improves simple, pre-determined survey knowledge by learning route knowledge and additional survey knowledge. It has been adapted to the Bremen Autonomous Wheelchair by Kollmann et al. (1997).

The Parti-Game algorithm has three prerequisites:

1. A mechanism that is able to detect collisions.
2. A self-localisation system that determines the agent's current state.
3. A navigation method that is able to move the agent from one state to another in an obstacle-free environment.

The Parti-Game algorithm allows an agent with these capabilities to also navigate between positions between which the *direct* connection is blocked by obstacles. It does this in two ways: on the one hand, it divides the space of all possible states into partitions of different size (cf. fig. 8). The granularity of the partitioning depends on the narrowness or spaciousness of a certain part of the state-space, i.e. the number of obstacles present in this region. As the partitioning can be used for navigating along any route, it is part of the survey knowledge. On the other hand, the Parti-Game algorithm determines whether it is possible to move from one partition to another one. This is only learned for the transitions between partitions that have already been visited. Therefore, this knowledge is route-dependent.

The Parti-Game algorithm is a very artificial navigation approach because most animals do not have a global self-localisation system. Nevertheless, birds are assumed

a) b)

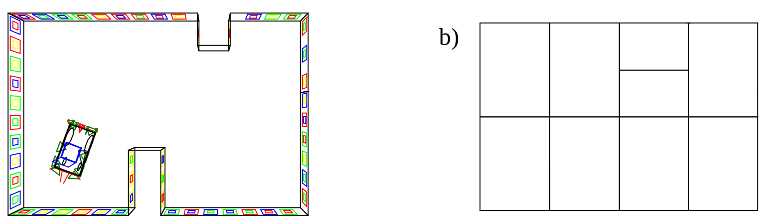


Fig. 8. The partitioning by the Parti-Game algorithm. a) Visualisation of a real world scene. b) The learned state-space.

to have a global “grid-map” that may be based on the earth’s magnetic field (Wiltschko 1997).

5 Navigation in Space by Image-Based Homing

The image-based navigation approach (Röfer 1995a) is inspired by the homing behaviour of honeybees (Cartwright and Collet 1982, 1983, 1987). It has been implemented using the simulation SimRobot (Siems et al. 1994, Röfer 1995b) and on the Bremen Autonomous Wheelchair utilising the panoramic sensor shown in fig. 9 (Röfer 1997a). To enable the method to be used in indoor navigation, the implementation goes beyond the bee’s homing behaviour in two aspects: it is able to determine the rotation as well as the direction to the target position, and it learns and autonomously reproduces complete routes. The peculiarity of this approach is that no associations between perceptions and actions are learned. Instead, the actions are directly calculated from a stored view and the current (cf. fig. 5). Locations are characterised by panoramic views and routes are represented as sequences of these views—as described in section 2.3.

The learned routes can be combined to a route graph. This enables the wheelchair to concatenate several routes to a longer one in order to drive between positions between which the direct route has not been trained. Nevertheless, the wheelchair is not able to construct new routes since it has no knowledge about the spatial relationship of the learned routes. Hence, this navigation approach does not deal with survey knowledge.

6 Route Navigation in Passages

An architecture with several layers has been chosen for the control system of the present Bremen Autonomous Wheelchair. In contrast to the aforementioned navigation approaches that have been inspired by biological findings or the reinforcement learning theory, the current approach follows the taxonomy presented in section 2. It consists of the levels “basic behaviours”, “route knowledge” and—in future work—“survey knowledge”.

6.1 Basic Behaviours

Several basic behaviours, e.g., wall-following and turning-into-door, form the basis of the presented navigation method. They enable the wheelchair to move in corridors,

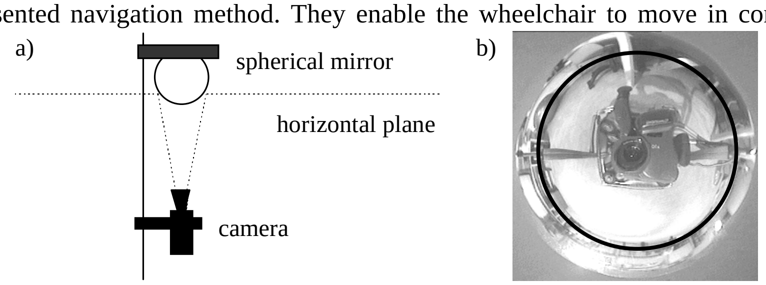


Fig. 9. Apparatus for taking panoramic images. a) Scheme. b) Image on the sphere. The circle indicates the mapping region of the horizontal plane.

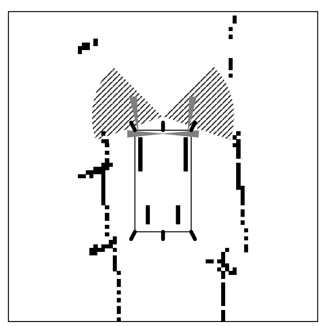
and to enter and exit rooms. They are fairly robust against changes in the environment because they hardly ever assume fixed compositions of the surroundings. Their implementation uses the sensors of the navigation subsystem (see section 3). Ultrasonic sensors have several weaknesses (Jörg et al. 1993). The signal that has been sent out by a sensor may not return to the same sensor if it hits a smooth surface diagonally (reflection) or is caught by another sensor (cross-talk).

Obstacle Map. As the wheelchair's sensors with the small opening-angle do not seem to produce cross-talks at all in their current arrangement but often miss smooth objects, it is not possible to implement the basic behaviours by a straight sensor-motor-linkage. Instead, the measured distances are recorded in an obstacle map (cf. fig. 10). This map plots the local environment around the wheelchair and represents an area of $4 \times 4\ m^2$. When the wheelchair drives on, the measurements in the map are shifted in the same way as the environment passes along the moving wheelchair. Everything that is scrolled out of the map is forgotten, as are measurements that are older than 30 seconds. This allows the wheelchair to cope with dynamic obstacles. Thus the local obstacle map corresponds to a short term memory.

The information in the obstacle map is abstracted by two *virtual sensors* in the map; their measurements are used for navigation. The virtual sensors work like ultrasonic sensors: they determine the distance to the closest obstacle in their measuring range. In contrast to real sensors, the virtual ones only measure distances to objects already represented in the obstacle map. Thus, they can exploit the sensory data that has been collected in the past 30 seconds; they may even detect an obstacle that the real sensors currently overlook. The two virtual sensors scan the map from a position that, in reality, is 10 cm in front of the wheelchair. One is oriented towards the left side; the other towards the right side (cf. fig. 10).

Six basic behaviours have been implemented:

Centre between Walls. If this behaviour is selected, the wheelchair tries to measure the same distance with both virtual sensors. To achieve this, it always steers in the direction of the larger measurement. First, the difference between both sensor readings has to be calculated:



▪ obstacle sensors: infrared ultrasonic virtual

Fig. 10. The local obstacle map.

$$\Delta = v_{left} - v_{right} \tag{1}$$

Then this difference is transformed into a *steering radius*. The steering radius is the radius of the arc that describes the wheelchair's trajectory. A negative radius describes a curve to the left, while a positive radius represents a curve to the right. As a special case, a steering radius of zero stands for a movement straight ahead:

$$r = \begin{cases} 0 & \text{if } \Delta = 0 \\ \operatorname{sgn}(\Delta)\dfrac{w}{\Delta^2} & \text{otherwise} \end{cases} \tag{2}$$

While performing this behaviour, the wheelchair is driving in the forward direction. As it is possible that an obstacle may have been overlooked, the collision detection subsystem perceives it and stops the wheelchair. In this case, the wheelchair performs the same behaviour in a backward direction for a distance of 50 cm and then returns to driving forwards. The infrared sensors will have detected the missed obstacle before the collision. As their measurements have been entered into the obstacle map, the wheelchair is aware of this obstacle during further actions.

To perform the same behaviour while driving backwards, the difference between the two virtual sensor readings simply has to be inverted.

Follow Left/Right Wall. Wall-following is realised by a slightly modified wall-centring behaviour. The only difference is that the measurements of the virtual sensor that is opposite to the wall are limited to a maximum distance of v_{max}. Therefore, the wheelchair assumes that there is another wall within a distance of v_{max} and centres itself between this virtual wall on one side and the real one on the other side. If the real corridor is narrower than this virtual one, there is no change to the wall-centring behaviour.

To follow a left wall, the difference between the virtual sensor readings has to be calculated as

$$\Delta_{left} = v_{left} - \min\left(v_{right}, v_{max}\right) \tag{3}$$

Turn into Left/Right Door. These two behaviours enable the wheelchair to turn into a door that is either in the left or the right wall. They are quite similar to the wall-following behaviours. The only difference is that on the side of the door, no virtual sensor is used. Instead, the measurement of the real ultrasonic sensor that is oriented toward this side is used. In this way, the hole between the door-jambs can be determined as soon as possible. If the wheelchair has turned more than 60°, it automatically switches to the wall-centring behaviour.

Follow a Direction while Driving Forward/Backward. This behaviour controls the wheelchair in a certain direction. As it is not equipped with a compass, the odometry is used as a reference system for the specification of the direction. The behaviour can be performed by driving forward or by driving backward. It is based—as all behaviours that have been presented so far—on the centring between walls. As a result, the wheelchair is still able to avoid obstacles while it is driving in a certain direction. Similar to the wall-following, a virtual corridor is constructed from the given direction; the wheelchair centres itself in this virtual corridor. If the real corridor is narrower than this virtual one, there is no difference to the normal wall-centring behaviour.

The virtual corridor is generated by calculating the difference α between the wheelchair's orientation and the desired direction. This difference is limited to a range of [-α_{max} ... α_{max}]. α_{max} determines how "aggressive" the wheelchair tries to follow the given direction. Then, the position in the direction of α is calculated that is on the same level as the virtual sensors. The values of the two virtual sensors are modified in a way as if the wheelchair would drive through a corridor with walls in a certain distance *s* to the left and the right of this position. In the case of driving forwards, Δ is calculated as

$$\Delta_{forward} = \min\left(v_{left}, s + o_{forward} \tan\alpha\right) - \min\left(v_{right}, s - o_{forward} \tan\alpha\right) \quad (4)$$

In this equation, $o_{forward}$ is the distance between the wheelchair's driving axle and the position of the virtual sensors.

Stop. As it is always the goal of the wheelchair to reach a certain position, it has to stop if it has arrived at the target.

6.2 Routemark Detection

The basic behaviours enable the wheelchair to *move* in an office environment. In order to also allow it to *navigate*, it must be able to localise itself in the environment. Therefore, it has to be capable of recognising reference points in the surroundings, e.g. to switch to the next behaviour. In the navigation approach that is presented in this article, these reference points are called routemarks because they are used to locate the wheelchair's position along a certain route. The long-term goal of the authors is to use some features of the environment's 3D-structure as routemarks. This will be described in section 0. At the moment, only artificial 2D-marks are employed, determined by an image processing algorithm. These marks consist of a black circle on a white background (cf. fig. 11). In the circle, there are up to four white, horizontal stripes interpreted as a scan-code. The recognition of the routemarks is performed in four steps:

1. First, the image is scanned for pixels that are darker than a certain threshold φ_{max}. If such a pixel p_i has been found, a continuous region of pixels that are all darker then 110% of p_i is determined, starting from the position of the found pixel.
2. In a second step, the width *w* and the height *h* of the region are calculated. If either the width is smaller than a predefined threshold w_{min} or the height is smaller than h_{min}, the extracted region is not used as a candidate for being a routemark.
3. Based on the values of *w* and *h*, 32 prototypical points of an ellipse with these dimensions are calculated. If the distance of at least one of these points to the region's border is larger then a predefined threshold d_{max}, the region is not assumed to be a valid mark.
4. Based on the region's height, the routemark's horizontal centre is analysed in a vertical direction to extract the embedded scan-code. Again, if there is not at

least one white stripe on the mark, it is ignored. If the selected area has passed all these hurdles, it is assumed to be a valid routemark.

Routemark Map. As the sizes of the artificial routemarks are known, the distances to them—as with routemarks detected by 3D-vision—can be determined. Therefore, the position of a mark relative to the wheelchair can be calculated from its distance and its bearing. This allows the creation of another map: the routemark map. This map is similar to the obstacle map that has already been presented. When the wheelchair moves, the positions of the marks in the map are adjusted on the basis of its dead reckoning system. As the obstacle map, the routemark map is local to the wheelchair, i.e. it only maps a region of a radius of 5 m around the vehicle. All routemarks that move out of this radius are forgotten. Through the routemark map, the wheelchair knows the positions of all marks in its vicinity, even if its camera does not currently see them. Hence, it represents a *view*, i.e. the current routemark constellation of a certain place.

6.3 Route Knowledge

In the second layer of the control hierarchy, the basic behaviours and the routemark recognition are combined. On this tactical level, the environment is represented as a set of routes. A route is a static way from a starting location to a target location. The wheelchair can drive along such a route by a concatenation of different basic behaviours while routemarks are used to trigger the starting and changing of these behaviours. They are the reference marks along the route. Therefore, a route is represented as a sequence of basic behaviours and the routemarks that trigger these behaviours. This sequence can be learned by the wheelchair, e.g. when a teacher controls the vehicle along the route by switching between the available behaviours. Meanwhile, the camera scans the surroundings for routemarks and inserts them into the routemark map. When the teacher alters the wheelchair's behaviour, it stores the view of the routemark map as the trigger for the new operation.

Multiple Routemarks. If the wheelchair drive along the learned route autonomously, it executes each recorded behaviour until it recognises a part of the routemark constellation that it has stored during the training to trigger the change to the next behaviour. As the wheelchair will never reach exactly the same positions during the training and the autonomous repetition, it cannot compare the stored routemark constellation with the current one directly. Instead, the intersection of the stored routemark constellation and the current state of the routemark map is determined. If at least one common routemark is found, it is checked whether the wheelchair has passed all these

Fig. 11. View of an artificial routemark. The square indicates recognition.

common routemarks. It has “passed” a routemark, if the current bearing of a mark is further back than the bearing of the stored mark. As soon as all common routemarks have been passed, the wheelchair switches to the next behaviour of the route. This solution is robust against the absence of single marks. Nevertheless, it tries to reproduce the moment of switching the behaviours as precise as possible by using multiple marks.

As a single recognised routemark is sufficient for switching from one behaviour to the next, behaviours might be switched too early if all routemarks of a certain place are stored because it is possible that the same marks have already been present earlier in this route segment. Since the decision when to switch to the next behaviour is based on the bearings of the marks, it is necessary to use only those marks that never had a bearing further back during the route segment than at its end.

Errors. In spite of the use of multiple routemarks for the switching of behaviours, errors during the autonomous drive of the wheelchair cannot be ruled out. Two types of errors are possible:

1. All routemarks for switching a behaviour have been missed. This can happen, e.g. if a routemark is hidden by a person.
2. During the autonomous drive, a behaviour is performed that is different from the version that has been trained. If, e.g., the wheelchair should follow the left wall and misses a turn-off because a person blocks the passage, the vehicle drives straight on where it should have turned left.

In both cases, two possibilities exist to finish the erroneous behaviour:

1. The execution of a behaviour lasts more than 50% longer than during the training. To be able to determine this fact, the wheelchair measures the duration of behaviours. If such an overflow has been detected, it is assumed that there was an error in the reproduction of the learned route.
2. The wheelchair finds different routemarks of the same type and switches to the next behaviour. In this case, the erroneous execution is not detected immediately but it can be expected that one of the subsequent behaviours cannot be completed and therefore will fail.

So, an error can only be determined if a behaviour lasts too long, but the cause of this error can also be a defective execution of a previous behaviour. Thus, a strategy for compensating errors must be able to correct errors in previous behaviours. If the wheelchair has missed some routemarks, it can track back the route segment and search again. If it has made an error in carrying out a behaviour, it has no other possibility than to drive back to a position before the error and to try again. As it does not know where the error occurred, it drives back the last route segment and searches for the missing routemarks; then it repeats this segment. If this again results in an error, it drives back the last two segments and repeats them etc. If it has repeated the complete previous route without being successful, the execution of the route fails. Such failures can only be recovered from on the level of survey knowledge, e.g. by also backtracking the previous route or by using a different one.

Backtracking. A straightforward approach for the implementation of backtracking along a route segment would be the use of “inverse behaviours”, i.e. the same behaviour but performed in backward direction. The problem of such an approach is that a behaviour can be carried out incorrectly in forward direction but its inverse behaviour may be executed accurately. In this case, the inverse behaviour would not drive the wheelchair back to its starting-point and therefore, it would not compensate the forward behaviour. To really cancel a behaviour, the wheelchair uses a threading tactic: it records the positions of is dead reckoning system during the execution of the behaviours and then uses the “follow a direction while driving backward”-behaviour to drive back the segment position by position. This allows it to cancel any behaviour, even if the behaviour was performed incorrectly.

As soon as the recognition of routemarks is not only seen as a binary decision but instead as a process with a particular uncertainty, the representation of knowledge becomes more complex. To develop a solution for this problem, several psychological findings could be employed, e.g. expected routemarks should be detected with a higher probability than unexpected ones. Marks in the same constellation could support each other in the recognition process.

6.4 Survey Knowledge

The autonomous generation of survey knowledge can be considered as the third layer of the architecture. This is the knowledge about the spatial relationship between the routes. On the basis of the survey knowledge, new routes could be generated from several learned ones and therefore shortcuts could be detected. The wheelchair could recognise that a particular segment of one route is also part of another route by comparing the routemark constellations stored for the routes. Thus, routes can be combined into a route graph that can be utilised to plan shortcuts as well as bypasses around obstacles. If it is necessary to find shortcuts or bypasses by exploration, i.e., without the corresponding route knowledge, dead reckoning must also be integrated into the navigation strategy.

6.5 Results

Although the wheelchair is 72 cm wide and 134 cm long, the basic behaviours have enabled it to drive through 94 cm wide door-frames and to turn into doors in a 176 cm wide corridor. Fig. 12 shows a trajectory that has been obtained from a combination of four different basic behaviours. The trajectory has been recorded by the wheelchair’s onboard odometry. As the odometers are not precise enough to dead reckon over 30 m, the recorded trajectory has been adjusted manually to allow the visualisation in the floor-plan. There are some obstacles in the office that are not shown in the plan, e.g. there is a coat-rack in the right corridor that the wheelchair has bypassed.

The wheelchair has been able to learn and to repeat a route consisting of four basic behaviours. It has been able to recover from both types of possible errors, i.e. from missing routemarks and from the erroneous performance of behaviours. Both errors have been forced by hiding routemarks and blocking passages. An important result for future work with natural 3D-routemarks is that it is necessary to always have a wide selection of routemarks because the worst thing that can happen is that the behaviour should be changed and no appropriate routemark to trigger this change is available.

7 Structure From Motion for the Visual Detection of Natural Landmarks and Routemarks

The aforementioned navigation approach is based on the recognition of routemarks. As an intermediate step, artificial routemarks are used that are easy to detect. Nevertheless, the long-term goal of the authors is to use natural features as routemarks. To this end it is desirable that the system can use local structures typically present in its surrounding as natural routemarks or landmarks. Such natural landmarks or routemarks should be recognised in a robust manner, independent of orientation, illumination or surface variations in dynamic environments. It is intended to use significant 3D-structures such as corners, doors or boxes. In the following such small 3D-structures used as landmarks or routemarks are denoted as semi-local 3D-marks, or 3D-marks for short.

The visual estimation of depth in a small (semi-local) image region is a fundamental prerequisite for the detection of such semi-local 3D-marks. The visual estimation of depth is usually based on at least two images. These are either taken simultaneously in stereo vision or subsequently in so-called “structure from motion” approaches (Barron 1984, Aggarwal and Nandhakumar 1988, Heeger and Jepson 1992). Here, the second approach is used because utilising a single camera is cheaper and takes less calibration effort, and structure from motion is computationally very efficient and thus advantageous for real time applications.

7.1 Method

The approach consists of two major steps: the estimation of the so-called *focus of expansion* which is the intersection of the heading direction vector and the image plane, and the subsequent estimation of *depth*.

The focus of expansion is estimated by a method that was developed in the authors’ group. It was described in detail by Herwig (1996). From two consecutive

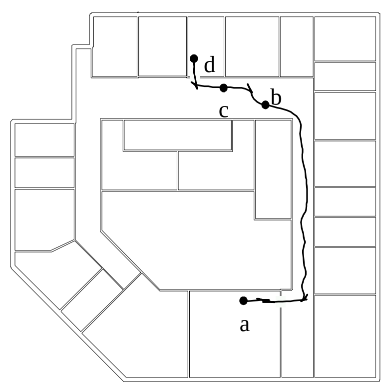


Fig. 12. A measured trajectory that has been generated by the combination of four basic behaviours. a) Follow left wall. b) Follow right wall. c) Turn into right door. d) Stop.

camera images, the intensity derivatives with respect to time *t* and spatial co-ordinates *x* and *y* are estimated. Assuming short-time constancy of brightness of a point in the observed scene, the normal optical flow field (Helmholtz 1909) is estimated from those derivatives. The focus of expansion is estimated from the normal optical flow field by minimising a cost function. This method has been shown to be robust with respect to observed scenes and noise, moreover it is computationally efficient and has been used in real time robotics applications.

In the second step, the ratio of local image depth to the observer's speed is estimated from the normal optical flow field and from the position of the focus of expansion, as Herwig (1996) proposed.

7.2 Computer Simulations

Currently, this method of estimating distance from image sequences is investigated with computer simulations using SimRobot (Siems et al. 1994).

From these images, the depth is estimated in the way described above. Presently, the influence of the algorithm's parameters and of the experimental settings on the estimated depth values is analysed. The most essential parameters of the algorithm are upper and lower thresholds for the intensity derivatives. Low values of intensity derivatives are likely to enhance the effect of noise. High values of intensity derivatives maybe due to quantisation errors; in such cases they cause inadequate flow field estimations. Moreover, the norm of the intensity gradient vector is required to be high enough in order to provide sufficient correspondence of the optical flow vector and the velocity vector of an image point (Verri and Poggio 1987, Aloimonos et al. 1993). Image points are not processed if their intensity derivatives do not match these criteria. Experimental settings include the walls' texture, the observer's speed and the resolution and noise intensity of the camera image.

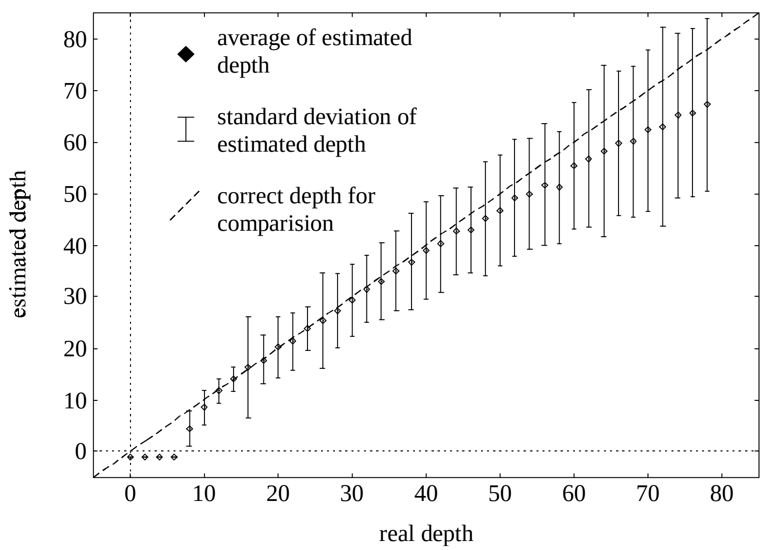


Fig. 13. Estimated depth vs. real depth.

7.3 Early Results

The results of typical simulations are shown in fig. 13 and fig. 14.

To judge the algorithm's performance for depth estimation, the camera approaches a single wall perpendicularly. In this set-up, depth values are equal for each image point. The results of such an experiment are shown in fig. 13. The trial started at a distance of 80 arbitrary units. Depth was estimated from consecutive pairs of images. Diamonds indicate the mean value of estimated depth of all image points. The error bars indicate the standard deviation. As this is the result of a simulation, the true distance is known. It is marked by the dashed line.

In the experiment shown in fig. 14, the camera approaches walls that are arranged as a step in distance. This simulation's set-up is sketched in part a) of the figure. Part b) shows the camera's view. The walls' texture consists of a quadratic grid of isotropic patches. The patches' intensity is maximal in their centre, dropping towards the periphery according to a cosine function. Part c) of the figure shows the estimated depth as grey-scale values. The cross-hair marks the estimated position of the focus of expansion. One may easily identify a depth-discontinuity from fig. 14c.

7.4 Future Perspective

As mentioned before, the long-term goal is to use three-dimensional structures as landmarks or routemarks for the navigation of the Bremen Autonomous Wheelchair. To achieve this, the distance images will have to be segmented into areas of nearly equal relative motion. In each such area, discontinuities of depth, and discontinuities and extrema of depth derivatives should be recognised. Significant depth discontinuities or extrema should be automatically selected and utilised as 3D-marks.

8 Conclusion

The taxonomy relates different navigational behaviours in a hierarchical and compositional way. It remains to be seen whether the distinction between navigation in space, using landmarks, and in networks of passages, using routemarks, is useful in psychology and artificial intelligence or robotics alike. The abstraction to survey knowledge, strategic navigation, dealing with exceptions etc. still require investigation.

The Bremen Autonomous Wheelchair has been a useful vehicle for experimenta-

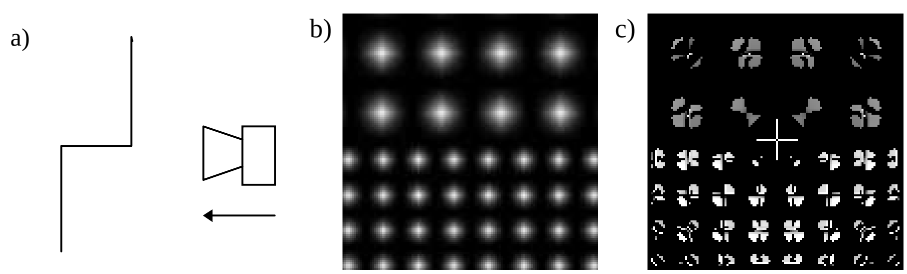


Fig. 14. A simulated camera approaching a distance step. a) Scheme of set-up. b) Camera view. c) Estimated depth coded as brightness. Brighter areas correspond to farther surfaces.

tion, not only to model biologically plausible navigational behaviours, but also, apparently, to develop efficient navigational mechanisms for a realistic technical application; this will still have to be shown in applications with users. A robust implementation of some basic behaviours and the recognition of artificial routemarks enable the wheelchair to autonomously follow trained routes. In the future, the identification of the artificial routemarks will be replaced by the recognition of natural 3D-landmarks or routemarks. Early results on this field indicate that the presented algorithm is capable of estimating depth in real time, with high spatial resolution and robust with respect to illumination, orientation, and noise.

9 References

Aggarwal, J. K., Nandhakumar, N.: On the computation of motion from sequences of images - a review. *Proc. of the IEEE* Vol. 76 (1988) 917-935.

Aloimonos, Y., Rivlin, E., Huang, L.: Designing Visual Systems: Purposive Navigation. In: Y. Aloimonos (Ed.), *Active Perception.* Lawrence Erlbaum (1993).

Barkowsky, T., Freksa, C., Berendt, B., Kelter, S.: Aspektkarten - Integriert räumlich-symbolische Repräsentationsstrukturen. In: C. Umbach, M. Grabski, R. Hörnig (Eds.): *Perspektive in Sprache und Raum.* Deutscher Universitätsverlag. Wiesbaden (1997) (to appear).

Barron, J.: A survey of approaches for determining optic flow, environmental layout and egomotion. *Tech. Report* RBCV-TR-84-5. Dept. of Computer Science, University of Toronto (1984).

Cartwright, B.A., Collet, T.S.: How honey bees use landmarks to guide their return to a food sorce. *Nature* Vol. 295 (1982) 560-564.

Cartwright, B.A., Collet, T.S.: Landmark Learning in Bees. *Journal of Comparative Physiology A* Vol. 151 (1983) 521-543.

Cartwright, B.A., Collet, T.S.: Landmark Maps for Honeybees. *Biological Cybernetics 57* (1987) 85-93.

Collet, T.S., Cartwright, B.A., Smith, B.A.: Landmarks learning and visuo-spatial memories in gerbils. *Journal of Comparative Physiology A* Vol. 158 (1986) 835-851.

Collet, T.S., Kelber, A.: The retrieval of visuo-spatial memories in honeybees. *Journal of Comparative Physiology A* Vol. 163 (1988) 145-150.

Collet, T.S., Baron, J.: Biological compasses and the coordinate frame of landmark memories in honeybees. *Nature* Vol. 368 (1994) 137-140.

Dyer, F.C.: Bees acquire route-based memories but not cognitive maps in familiar landscape. *Animal Behavior* Vol. 41 (1991) 239-246.

Franz, M.O., Schölkopf, B., Georg, P., Mallot, H.A., Bülthoff, H.H.: Learning View Graphs for Robot Navigation. In: Johnson, W.L. (Ed.): *Proc. 1st Int. Conf. on Autonomous Agents.* ACM Press, New York (1997a) 138-147.

Franz, M.O., Schölkopf, B., Bülthoff, H.H.: Homing by Parameterized Scene Matching. In: *Proc. 4th Europ. Conf. on Artificial Life* (1997b).

Frisch, K.v.: The dance language and orientation of bees. Belknap Press of Harward University Press, Cambridge, Massachusetts / Oxford University Press, London (1967).

Gallistel, C.R.: The organization of learning. Cambridge, Massachusetts: MIT Press (1990).

Gillner, S. Mallot, H.A.: Navigation and acquisition of spatial knowledge in a virtual maze. *Technical Report* 045, Max-Planck-Institut für biologische Kybernetik, Tübingen (1997).

Heeger, D. J., Jepson, A. J.: Subspace Methods for Recovering Rigid Motion I: Algorithm and Implementation. *Int. J. of Computer Vision* Vol. 7 No. 2 (1992) 95-117.

Helmholtz, H. v.: Handbuch der Physiologischen Optik. 3rd edn. Verlag von Leopold Voss, Hamburg (1909).

Herrmann, T., Schweizer, K., Janzen, G., Katz, S.: Routen und Überblickswissen – konzeptuelle Überlegungen. Bericht Nr. 1, Arbeiten des Mannheimer Teilprojekts im DFG Schwerpunktprogramm „Raumkognition“, Universität Mannheim (1997).

Herwig, Ch.: Visual Motion Processing for Active Observers. PhD Thesis. Krieg-Brückner, Roth, Schwegler (Eds.): *ZKW-Bericht Nr. 1/96*. Zentrum für Kognitionswissenschaften, Universität Bremen (1996).

Jörg, K.-W., Puttkamer, E. v., Richstein, H.-J.: Integration und Fusion heterogener Multisensorinformation zur geometrischen Weltmodellierung für einen Autonomen Mobilen Roboter. In: *Autonome Mobile Systeme*, 9. Fachgespräch. Schmidt, G. (Ed.). Technische Universität München. (1993) 287-298.

Kollmann, J., Lankenau, A., Bühlmeier, A., Krieg-Brückner, B., Röfer, T.: Navigation of a Kinematically Restricted Wheelchair by the Parti-Game Algorithm. In: *Spatial Reasoning in Mobile Robots and Animals, AISB-97 Workshop.* Manchester University (1997) 35-44.

Mittelstaedt, H., Mittelstaedt, M.-L.: Homing by Path Integration. In: Papi, F. & Wallraff, H.G. (Eds). *Avian Navigation* (1982) 290-297.

Moore, A., Atkesen, C.: The Parti-Game Algorithm for Variable Resolution Reinforcement Learning in Multidimensional State-Spaces. *Machine Learning* 21. Kluwer Academic Publishers, Boston. (1995) 199-233.

Müller, M., Wehner, R.: Path integration in desert ants, Cataglyphis fortis. *Proc. Natl. Acad. Sci. USA* Vol. 85. (1988) 5287-5290.

Nehmzow, U.: Animal and Robot Navigation. *Robotics and Autonomous Systems* Vol. 15, No. 1-2 (1995) 71-81

Poucet, B.: Spatial cognitive maps in animals: New hypotheses on their structure and neural mechanisms. *Psychological Review* Vol. 100 (1993) 163-182.

Röfer, T.: Image based homing using a self-organizing feature map. In: Fogelman-Soulie, F., Gallinari P. (Eds.): *Proc. Int. Conf. Artificial Neural Networks*, EC2 & Cie, Vol. 1 (1995) 475-480.

Röfer, T.: Navigation mit eindimensionalen 360° Bildern. In: Dillmann R., Rembold U., Lüth T. (Hrsg.): *Autonome Mobile Systeme 1995*. Informatik aktuell. Springer Verlag (1995) 193-202.

Röfer, T.: Controlling a Wheelchair with Image-based Homing. In: *Spatial Reasoning in Mobile Robots and Animals, AISB-97 Workshop*. Manchester University (1997a) 66-75.

Röfer, T.: Routemark-Based Navigation of a Wheelchair. *Third ECPD International Conference on Advanced Robotics, Intelligent Automation and Active Systems*, Bremen (1997b) (to appear).

Rossel, S., Wehner, R.: Polarization vision in bees. *Nature* Vol. 323 (1986) 128-131.

Schmid, U., Hernández, D., Krieg-Brückner, B., Eisenkolb a. Musto, A., Schill, K., Wiebrock, S., Wysotzki, F., Schlieder, C., Renz, J., Nebel, B.: Process Control and Planning for Movement in Space: Qualitative and Metric Approaches to Spatial Inference and the Analysis of Movements. Technical Report, TU Berlin (1997).

Schölkopf, B., Mallot, H.A.: View-based cognitive mapping and planning. *Adaptive Behavior* Vol. 3 (1995) 311-348.

Scholl, M.J.: Cognitive Maps as Orienting Schemata. *Journal of Experimental Psychology* Vol. 13, No. 4 (1987) 615-628.

Schweizer, K., Janzen, G.: Zum Einfluß der Erwerbssituation auf die Raumkognition: Mentale Repräsentation der Blickpunktsequenz bei räumlichen Anordnungen. *Sprache und Kognition* Vol. 15, No. 4 (1996) 217-233.

Siems, U.; Herwig, Ch.; Röfer, Th.: SimRobot – Ein Programm zur Simulation sensorbestückter Agenten in einer dreidimensionalen Umwelt. *ZKW Bericht 1/94*, Zentrum für Kognitionswissenschaften, Universität Bremen (1994).

Verri, A., Poggio, T.: Against Quantitative Optical Flow. *Proc. 1st IEEE Int. Conf. Computer Vision* (1987) 171-180

Wehner, R.: Celestial and Terrestial Navigation: Human Strategies—Insect Strategies. In: Huber, F., Markl, H. (Eds.): *Neuroethology and Behavioural Physiology*. Springer, Berlin, Heidelberg, New York (1983) 366-381.

Wehner, R., Menzel, R.: Do insects have cognitive maps? *Annu. Rev. Neurosci.* 13. (1990) 403-414.

Werner, S., Krieg-Brückner, B., Mallot, H. A., Schweizer, K., Freksa, Ch.: Spatial Cognition: The Role of Landmark, Route, and Survey Knowledge in Human and Robot Navigation. *GI Tagung '97, Aachen* (to appear).

Wiltschko, R., Wiltschko, W.: Magnetic Orientation in Animals. Springer, Berlin, Heidelberg, New York (1995).

Wiltschko, R.: The Navigation System of Birds. In: *Spatial Reasoning in Mobile Robots and Animals, AISB-97 Workshop*. Manchester University (1997) 5-14.

Wittmann, T.: Insektennavigation: Modelle und Simulationen. *Berichte aus der Biologie*. Shaker, Aachen (1996).