Biological Cybernetics manuscript No.
(will be inserted by the editor)

Insect visual homing strategies in a robot with analog processing

Ralf Möller

Artificial Intelligence Lab, Department of Computer Science, and Department of Zoology,
University of Zurich, Winterthurerstr. 190, CH-8057 Zurich, Switzerland
(e-mail: moeller@ifi.unizh.ch)

Received: date / Revised version: date

Abstract The visual homing abilities of insects can be explained by the snapshot hypothesis. It asserts that an animal is guided to a previously visited location by comparing the current view with a snapshot taken at that location. The average landmark vector (ALV) model is a parsimonious navigation model based on the snapshot hypothesis. According to this model, the target location is unambiguously characterized by a signature vector extracted from the snapshot image. This paper provides threefold support for the ALV model by synthetic modeling. First, it was shown that a mobile robot using the ALV model returns to the target location with only small position errors. Second, the behavior of the robot resembled the behavior of bees in some experiments. And third, the ALV model was implemented on the robot in analog hardware. This adds validity to the ALV model, since analog electronic circuits share a number of information processing principles with biological nervous systems; the analog implementation therefore provides suggestions how visual homing abilities might be implemented in the insect's brain.

1 Introduction

1.1 Models of insect navigation

Being able to return to a shelter or a food source is a vital ability for many animal species. Social insects are particularly impressive examples, since they accomplish this task with tiny nervous systems that are confined to less than 1mm^3 size and less than a million neurons. One of the homing strategies employed for instance by bees and ants is visual landmark navigation. The behavior exhibited by these animals in experiments with different landmark setups can be explained by the "snapshot hypothesis" (Wehner and Räber, 1979; Cartwright and Collett, 1983; Wehner et al., 1996). It claims that the animal memorizes the landmark panorama surrounding the target location. This "snapshot" guides the return journey of the animal in the vicinity of the target location. The animal continuously compares the landmark panorama visible from its current location with the snapshot. From the discrepancies between the two images it derives a home direction; a movement in this direction will reduce the image discrepancy and therefore bring the animal closer to the target location.

An algorithmic model based on the snapshot hypothesis was first presented by Cartwright and Collett (1983), in the following referred to as "snapshot model". It assumes that the horizonal portion of the landmark panorama is segmented into dark and bright sectors and stored as "snapshot image" in the insect's brain. On the journey home, a matching process establishes correspondences between neighboring sectors of the same type in snapshot and current view. The home direction is determined as the average of the movement directions that would reduce the differences in bearing and size of each pair of sectors. Such an image matching procedure between an image stored in some representation and the current view is characteristic for the snapshot model and other models of the same class (for a survey see Franz et al., 1998).

The image matching assumption is abandoned in the "average landmark vector model" (ALV model) (Lambrinos et al., 1998, 1999; Lambrinos, 1999). From the horizonal panorama visible at the snapshot location, a two-component vector, the "average landmark (AL) vector", is extracted and stored. As proved mathematically in this paper, this vector unambiguously characterizes the target location. In the same way, an AL vector is determined in the current location. The difference between the two AL vectors gives the home vector. Compared to the snapshot model, the ALV model is significantly simpler: the snapshot image is replaced by a vector, and the matching process is reduced to a vector subtraction. Nevertheless, the ALV model is closely related to the snapshot model, and in the vicinity of the target location even produces home vectors identical to those obtained from a version of the snapshot model (Lambrinos et al., 1999).

1.2 Synthetic modeling

"Synthetic modeling" is a novel biological methodology to gain insights in the mechanisms underlying some behavior

of a biological agent. Models developed to explain the animal's abilities are implemented on an artifical agent. The robot is exposed to an environment similar or even identical to the environment experienced by the animal. By observing the behavior and the internal states of the robot, the implemented models can be validated. The robot implementation of a model can include technical sensors mimicking the design of their biological counterparts, specialized electronic circuits or computer algorithms reproducing the information processing at different levels of modeling, as well as replicas of the body morphology and the actuators of the animal.

Synthetic modeling with artificial agents is an indispensable complement to computer simulations and in some cases even the only way to verify a biological model, especially when the agent-environment interaction is too complex to be simulated with sufficient accuracy. Simplifications of the complexity of the real world in computer simulations could severely misguide the development of models. In the past, synthetic modeling has therefore been used in cases with complex sensory input (visual, auditory), complex properties of the world (turbulences in water), and complex mechanical interaction between the agent and the world (legged locomotion), e.g. to investigate the visuomotor system of the housefly (Franceschini et al., 1992), cricket phonotaxis (Webb, 1995), visual odometry in bees (Srinivasan et al., 1997), lobster chemotaxis (Grasso et al., 1996), legged locomotion of insects (Cruse et al., 1995), and skylight and visual landmark navigation of desert ants (Lambrinos et al., 1997, 1999).

1.3 Synthetic modeling using analog hardware

The signal processing in biological nervous systems is analog, asynchronous, and parallel, and differs widely from the digital, synchronous, and sequential processing in traditional digital computers. In some cases, software implementations of models lack biological plausibility, since no statements can be derived if and how the mechanism could actually be implemented in a biological nervous system. Some operations which can be realized with minimal effort on a digital computer (like shifting an array of data by manipulating a single pointer variable) may require considerable effort in a neural system. It is possible that the complexity of an equivalent neural model exceeds the limits set by the brain size of the animal. Moreover, models which rely on the high precision achieved in a digital, noise-free computation may miss the biological reality with its noisy, unreliable processing elements by far.

In a number of synthetic modeling projects, digital computers have been replaced by analog electronic hardware; one of the first projects was an analog implementation of elementary motion-detectors of the housefly by Franceschini et al. (1992). Many projects using analog hardware to emulate neural systems have their roots in the field called "neuromorphic engineering" (Mead, 1989; Douglas et al., 1995), where analog subthreshold VLSI is the preferred technology. Analog electronic circuits share a number of information processing principles with nervous systems: signals are encoded in analog values, there is no global clock, and the processing is inherently parallel. Typical operations of biological neurons like weighted addition of signals are easy to implement in analog hardware. Both analog electronics and biological neurons are affected by noise and parameter changes through external influences. The implementation in analog hardware forces the designer to take these issues into account and will therefore lead to biologically plausible models.

1.4 Contributions of this paper

In this paper, the synthetic modeling approach is used to gain insights in the visual navigation abilities of insects. First, it is shown that the ALV model, which so far has only been tested in computer simulations, works on a mobile robot and produces only small position errors in homing experiments. Second, some results obtained in experiments with bees could be reproduced with the robot. Third, the robot is controlled by a completely analog circuit implementing the ALV model. This kind of implementation leads naturally to suggestions about the neural circuits that might mediate homing in insect brains.

The ALV model and its relations to other models of insect and rodent navigation is presented in Section 2. Section 3 describes the robot and the analog implementation of the ALV model. The results of homing experiments with the robot are reported in Section 4 and discussed in Section 5.

2 Navigation model

2.1 Average landmark vector model

Fig. 1 visualizes the homing mechanism of the ALV model in its simplest version. A unit vector points from the position of the agent towards each detected landmark feature, in this case the center of black sectors in the horizonal view; these vectors are called "landmark vectors". Their average — the average landmark (AL) vector — is an unambiguous signature for each location; see Section 2.5.3. The AL vector of the target location is stored. On the return journey, the AL vector of the current location is determined; the difference between this vector and the stored AL vector is the home vector which points approximately to the target location. By following the continuously updated home vector, the agent will return to that location. Note that the ALV model, as well as all other models presented here, requires some kind of external reference to align the views or vectors to the same compass direction.

The formal description of the ALV model presumes that the axes of the agent's coordinate systems are aligned with the corresponding axes of the world coordinate system. The positions of n landmark points in the plane are given by $\mathbf{x}_i$, $i = 1, \ldots, n$. From each agent position $\mathbf{x}$ in the plane, a landmark vector $\mathbf{L}_i(\mathbf{x})$ with unit length points towards landmark i:

$$\mathbf{L}_i(\mathbf{x}) = \frac{\mathbf{x}_i - \mathbf{x}}{||\mathbf{x}_i - \mathbf{x}||}. \quad (1)$$

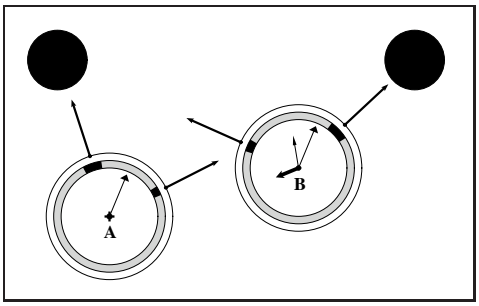

Fig. 1 Homing mechanism of the ALV model. Landmarks are shown as black circles. The target position is marked with a cross (A). Each grey ring visualizes the horizonal portion of the landmark panorama as perceived from the position in the center of the ring. Vectors attached to the outer ring depict landmark vectors. **A:** The AL vector of the target location is computed from the average of the landmark vectors and stored in memory (vector in the center). **B:** The difference of the AL vectors of current location (thin vector, small head) and target location (thin vector, wide head) gives the home vector (thick vector).

Visibility of all landmarks from all points is presumed in this description. For simplicity, the AL vector $\mathbf{A}(\mathbf{x})$ of position $\mathbf{x}$ is expressed as the sum (not the average) of the landmark vectors (see Section 2.5.1):

$$\mathbf{A}(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{L}_i(\mathbf{x}). \tag{2}$$

Given a target position $\mathbf{x}_0$, a home vector field $\mathbf{H}(\mathbf{x})$ can be computed by subtracting the AL vector of the target location $\mathbf{A}(\mathbf{x}_0)$ from the AL vector field $\mathbf{A}(\mathbf{x})$:

$$\mathbf{H}(\mathbf{x}) = \mathbf{A}(\mathbf{x}) - \mathbf{A}(\mathbf{x}_0). \tag{3}$$

In the homing process, the agent follows $\mathbf{H}(\mathbf{x})$ in order to return to the target location $\mathbf{x}_0$ (c is a constant that determines the speed):

$$\dot{\mathbf{x}} = c\mathbf{H}(\mathbf{x}). \tag{4}$$

Despite its parsimony, the ALV model successfully copes with complex environments with a high number of landmarks which are partly covering each other, as shown in the computer simulation in Fig. 2. Except for a few collisions with landmarks, all trajectories end in the target location.

2.2 Relation between ALV and snapshot model

The specific algorithm of the snapshot model (Cartwright and Collett, 1983) was determined in a series of computer simulations in a way that it reproduces behavioral data from bees. Fig. 3 (A) describes the matching process. Dark and bright sectors are extracted from the horizonal portion of the images at target and current location. Each sector in the snapshot is paired with the closest sector of the same type in the current view. From each pair of sectors, two contribution vectors are determined: a vector radial to the snapshot sector, pointing in a direction of movement that would reduce the difference in

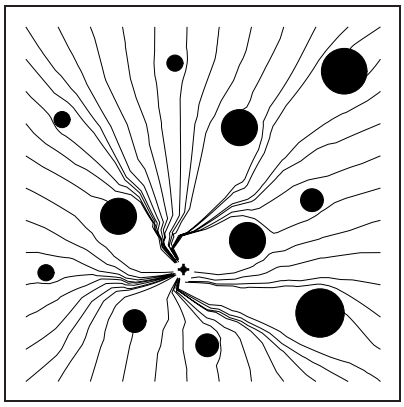

Fig. 2 Performance of the ALV model in a situation with 11 landmarks which are partly covering each other. Landmarks and target position are depicted as in Fig. 1. All trajectories start at the margin of the diagram.

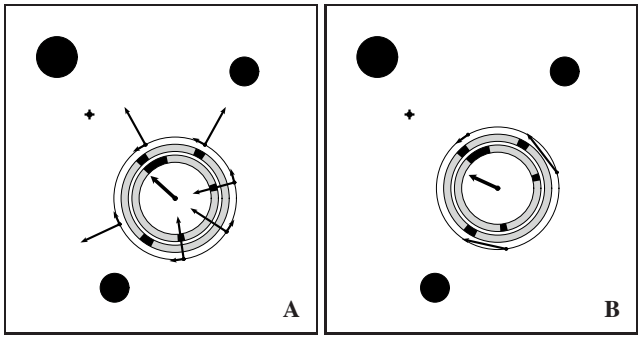

Fig. 3 Homing mechanism of the snapshot model (**A**) and the difference vector model (**B**). The inner grey ring shows the snapshot taken at the target location marked with a cross. The outer grey ring depicts the current view as visible from the center of the rings. Vectors attached to the outer circle are contribution vectors, the vector originating in the center is the home vector.

apparent size of the paired sectors, and a vector attached tangentially to the snapshot sector that would reduce the difference in bearing. The contribution vectors of both types have a constant length with a ratio of 1:3 between tangential and radial contributions, which was chosen in the original paper to fit the behavioral data. The average or sum of all contribution vectors gives the home vector.

The link between snapshot model and ALV model can be established by the "difference vector model" (DV model), a version of the snapshot model (Lambrinos et al., 1999). In the DV model, contributions from size differences are disregarded, only one type of sectors is considered in the matching process, and the tangential vectors — which are just one out of many possible choices — are replaced by secant vectors; see Fig. 3 (B). The matching process itself remains the same: the secant vectors are determined from a pair of neighboring sectors in snapshot and current view. Each secant vector is the difference between a unit vector pointing to the current view sector and a unit vector pointing to the corresponding snapshot sector. If the snapshot image $\mathcal{I}_0$ is described by the

set of sector centers

$$\mathcal{I}_0 = \{\mathbf{L}_1(\mathbf{x}_0), \ldots, \mathbf{L}_n(\mathbf{x}_0)\}, \quad (5)$$

and the current image $\mathcal{I}$ by

$$\mathcal{I} = \{\mathbf{L}_1(\mathbf{x}), \ldots, \mathbf{L}_n(\mathbf{x})\}, \quad (6)$$

the secant contribution vector $\mathbf{C}_i(\mathbf{x})$ is obtained from

$$\mathbf{C}_i(\mathbf{x}) = \mathbf{L}_{M(i)}(\mathbf{x}) - \mathbf{L}_i(\mathbf{x}_0), \quad (7)$$

where $M(i)$ is the index of the landmark visible in the current view that is paired with the image of landmark i in the snapshot. The home vector of the DV model is the sum of all contribution vectors:

$$\mathbf{H}^d(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{C}_i(\mathbf{x}) = \sum_{i=1}^{n} (\mathbf{L}_{M(i)}(\mathbf{x}) - \mathbf{L}_i(\mathbf{x}_0)). \quad (8)$$

A comparison between this home vector and the home vector of the ALV model obtained from equation (2) and (3)

$$\mathbf{H}(\mathbf{x}) = \sum_{i=1}^{n} (\mathbf{L}_i(\mathbf{x}) - \mathbf{L}_i(\mathbf{x}_0)) \quad (9)$$

reveals that the ALV model is identical to the DV model if $M(i) = i$, i.e. if the DV model establishes a perfect match between the landmark sectors in snapshot and current view. In a perfect match, the image of each landmark in the snapshot is paired with the image of the *same* landmark in the current view. On the one hand, this implies that the DV model and the ALV model yield identical home vectors in the vicinity of the target location where the matching procedure of the DV model leads to a correct pairing. On the other hand, this uncovers an interesting property of the ALV model: it *implicitly* results in a perfect match between the two views. Note that *none* of the models requires an *identification* of the landmark from its image.

The striking parsimony of the ALV model — only one vector has to be stored instead of a set of landmark vectors, and the matching procedure is replaced by a vector subtraction — results from a mathematical shortcut that is possible under the perfect-match condition $M(i) = i$: the ALV model *simultaneously* computes and sums all contribution vectors $\mathbf{C}_i(\mathbf{x})$ when it subtracts the sum of all landmark vectors in the snapshot image from the sum of all landmark vectors in the current view. This corresponds to a splitting of the sum in equation (9) into the two sums of equation (3) and (2).

2.3 Relation between ALV and surroundedness model

Anderson (1977) criticized the "retinal matching" assumption underlying the snapshot model which was put forward in an earlier paper by Collett and Land (1975). From his landmark navigation experiments he concluded that "the bee is not measuring the position of individual landmarks but the overall landmark configuration". As a measure for the overall configuration, Anderson defined the "surroundedness" by landmarks. The AL vector may be an alternative way of mathematically expressing "surroundedness". Actually, as will be demonstrated with a robot experiment in Section 4.3, the ALV model reproduces the results of Anderson's main experiment, where the search position of bees was shifted when a part of a circular array of landmarks had been removed between training and test. Being closely related to the snapshot model (Section 2.2) on the one hand and providing an overall measure of the landmark configuration with the AL vector on the other, the ALV model may bridge the gap between the retinal matching assumption and the surroundedness hypothesis. On the one hand, the snapshot model (in the DV version) and the ALV model return identical home vectors in the vicinity of the target location. On the other hand, in the ALV model the image information is reduced to a simple measure describing the overall arrangement of landmarks as the "center of gravity" of the visual cues in the image.

2.4 Relation between ALV and centroid model

The neurocomputational model of rat navigation suggested by O'Keefe (1991) is based on the computation of the "centroid", the geometric center of landmark cues in the environment. Some kind of compass system presumed, the agent can compute a vector pointing to the centroid location of the environment by adding the vectors that point from the current position to each landmark and that have a length equal to the distance from the landmark. By computing the difference between the centroid vector at the current location and the centroid vector at the target location, the agent can determine direction and distance to the target location.

The ALV model can be understood as an approximation of the centroid model under an "equal-distance assumption" (Franz et al., 1998). The landmark vectors $\mathbf{L}_i^c(\mathbf{x}) = \mathbf{x}_i - \mathbf{x}$ of the centroid model are replaced by vectors with the same direction but constant length in the ALV model; compare equation (1). Instead of using unit vectors, the length of the vectors could also be set to an estimate of the mean distance from all landmarks, which reveals the core of the approximation. Not having to estimate the distance to each landmark is an advantage of the ALV model. The trade-off is that homing becomes an iterative process: the home location can not be determined in one step as in the centroid model, but has to be continuously recomputed during the approach.

2.5 Properties of the ALV model

2.5.1 Average vs. sum of landmark vectors The sum in equation (2) has to be divided by the number of landmarks n to express the *average* of landmark vectors. In order to reduce the hardware effort, this normalization was left out in the analog circuit of the robot. As long as the number of visible landmarks is constant, division by n only results in a different scaling of the vectors. The scaling is arbitrary, since the length of the landmark vectors was also arbitrarily chosen as

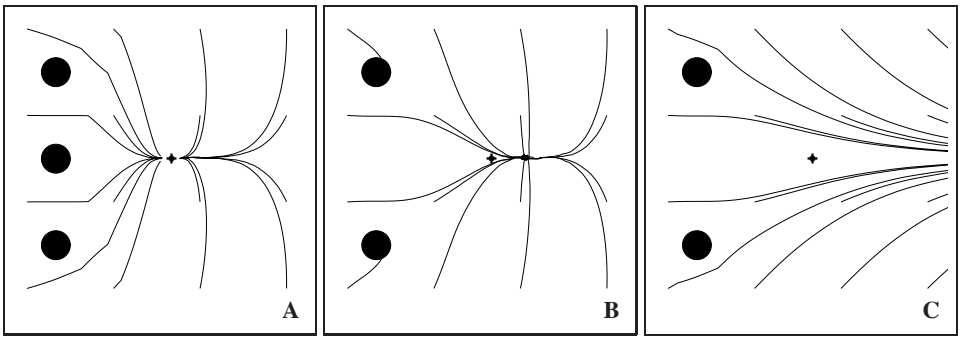

Fig. 4 Average of landmark vectors vs. sum of landmark vectors. **A:** Trajectories in the training situation (almost identical for both versions of the model). **B:** Model with averaging: removal of one landmark between training and test results in a small shift of the expected target position to the right. **C:** Model with summation: trajectories run to infinity. All trajectories start at points on a grid. The target position is marked with a cross.

1. Therefore, the resulting trajectories will be identical. A difference between the sum and the average of landmark vectors only becomes apparent in two cases: if landmarks are covering each other, or if the number of landmarks is changed between training and test. Fig. 4 visualizes the effect: for both versions of the ALV model, the average or sum of landmark vectors was computed in the target point of situation (A) and stored. Both models show approximately the same behavior in the training situation, except in regions where landmarks are covering each other. If one of the landmarks is removed in the test, but the stored vector remains unaffected, the behavior of the two versions differs considerably: while in the case of averaging the convergence point is only slightly shifted (B), trajectories run to infinity in the case of summation (C).

2.5.2 Suitable landmark features A prerequisite for the ALV model is that the same landmarks as in the target location will be detected when the robot is moved away from that location. The number of features in the image which are selected as landmark cues and therefore assigned a landmark vector should not change. This condition excludes mechanisms like the one shown in Fig. 5 (B), where a landmark vector points towards each dark pixel in the view. Trajectories starting at points close to the target position resemble those obtained by following the gradient on a potential function with a saddle point in the target location: independent of the time direction, trajectories approaching the target bend away before they reach this location. For comparison, Fig. 5 (A) shows the trajectories of the ALV model, if sector centers are used as landmark cues. In this case, all trajectories end in the target point (except for those running into landmarks). In the analog implementation, edges are used as landmark cues equivalent to sector centers, since the hardware effort for the detection of edges is smaller.

2.5.3 Convergence The parsimony of the ALV model is not only attractive from the modeling perspective, but also facilitates a mathematical analysis of its properties; this in turn can shed some light on the related snapshot model, where a mathematical treatment is complicated as a result of the use of unit vectors and the abrupt changes in the pairing of landmark features. Appendix A presents a proof of the global convergence of the ALV model: it is shown that all trajectories converge to the target location $\mathbf{x}_0$, except for those running into one of the landmarks (see the example in Fig. 5, A). This property is guaranteed, as long as not all landmarks are located on the same line through $\mathbf{x}_0$; this directly leads to the condition $n \geq 2$. The analysis also reveals that each AL vector unambiguously characterizes one location in the plane; there are no two positions in the plane that have identical AL vectors. The AL vector can therefore be interpreted as a transformation from Cartesian to curvi-linear coordinates.

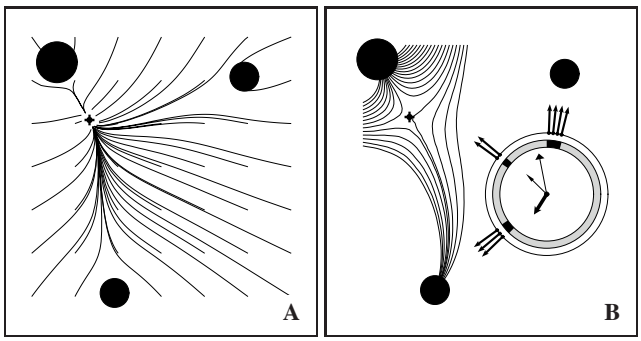

Fig. 5 Suitable landmark features for the ALV model. **A:** Trajectories of a model where sector centers are used as landmark cues. Trajectories start at points on a grid. **B:** Trajectories of a model where landmark vectors (outer ring) point towards each dark pixel in the image. The starting points of the trajectories are positioned on lines at the upper left and top left margin of the diagram. See Fig. 1 for an explanation of the contribution diagram.

2.5.4 Other properties DV model and ALV model share an advantage over the snapshot model: the length of the home vector relates to the distance from the goal (Lambrinos et al., 1999). This is not the case for the snapshot model, since there the contribution vectors have the same length independent of the difference in bearing or apparent size. The robot implementation benefits from this advantage: the speed of the robot depends on the length of the home vector and is therefore automatically reduced to zero when the goal is approached.

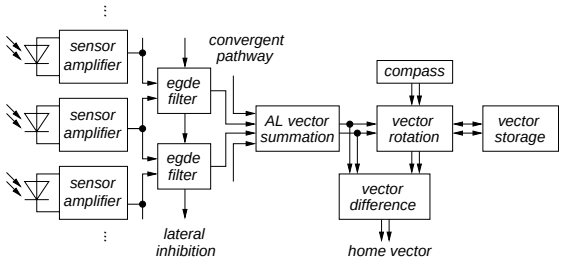

Fig. 6 Overview of the analog implementation of the ALV model.

This avoids overshooting and oscillations around the target position.

Another advantage of the ALV model concerns the alignment to an external compass reference. While in the models of the snapshot class either the robot or one of the images has to be rotated, only one of the AL vectors has to be rotated in the ALV model; in an analog implementation, this can be accomplished with a simple multiplier circuit (Section 3.5).

It is the basic version of the ALV model which is presented here and implemented on the robot. Several other variants have been tested in computer simulations. An extended version of the ALV model which regards the size of the landmarks was suggested by Lambrinos et al. (1999).

3 Robot hardware

3.1 Overview

The implementation of the ALV model is based entirely on discrete analog components; most of the active components are operational amplifiers (op-amps). An overview of the circuit is given in Fig. 6. Visual input comes from a ring of photo diodes (Fig. 7, A) mimicking the portion of ommatidia of both insect eyes that is facing the horizon. The signals of the photo diodes are amplified. Edges of one polarity are used as landmark cues; they are detected by combining the signals of two neighboring sensors. Unidirectional lateral inhibition between neighboring edge filters ensures that only one pixel per edge becomes active. The AL vector is determined from the binary representation of the edges. Using an electronic compass, the AL vector is rotated to world coordinates and stored, when a switch is pressed. In the homing process, the stored AL vector is transformed back to robot coordinates and subtracted from the AL vector of the current location. The components of the resulting home vector directly affect the speed of the two motors.

3.2 Sensors

The 32 Si photo diodes are horizontally arranged in a circular aluminum ring (Fig. 7, A). The angle between two neighboring sensors is $\beta = 11.25°$. Each diode faces an aperture with a diameter of $2.5mm$ in a distance of $18mm$ from the photo-sensitive surface. Care was taken to reduce the influence of light from outside the intended opening angle which

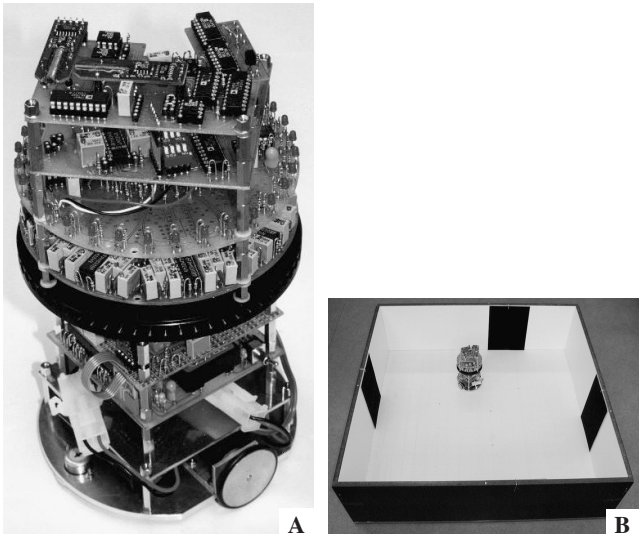

Fig. 7 A: Robot (height $22cm$, diameter $12.8cm$, weight $1.3kg$). The black ring contains the 32 photo diodes, the boards above the ring implement the ALV model, the boards below belong to the motor control. **B:** Arena for robot experiments ($1m \times 1m \times 0.3m$).

is reflected at the walls of the opening: a thread was cut in the front part of the hole, the aluminum was anodized with black color, and the thickness of the aperture ring was reduced. The effective opening angle (half-width of a Gaussian sensitivity function) was determined experimentally for different aperture diameters. For $2.5mm$ aperture diameter, an opening angle of $8°$ is obtained; see Section 3.3.

The signals of the photo diodes are amplified using a standard op-amp circuit. Amplification and offset of each amplifier were calibrated in a way that the amplified signals of all diodes were roughly identical when facing a white ($0.4V$) or a black surface ($0V$) under constant light conditions.

3.3 Edge filters

The outputs of two neighboring sensor amplifiers are compared in order to the extract edges that serve as landmark cues. Only edges of one polarity — clockwise black to white — are detected; the detection of edges of both polarities would have required additional hardware effort. For the flat landmarks used in the experiments (see Section 4) there is no difference between using the sector center (as in the simulations in Section 2.1) or one of the sector edges as landmark cue. For landmarks with complex shapes, the behavior of the model with different cues still has to be investigated; note that the proof in Appendix A is based on the assumption of landmark *points*. Fig. 8 (A) shows the schematics of the edge-filter circuit. One of the amplified photo signals is directly fed to the comparator, the other signal is slightly attenuated in a voltage divider and shifted with the threshold voltage $V_{ref} = 2V$.

Each active edge pixel influences the AL vector. Therefore the number of active pixels has to be kept constant in all positions and orientations of the robot; see also Section 2.5.2. This requires an additional mechanism, as will be explained

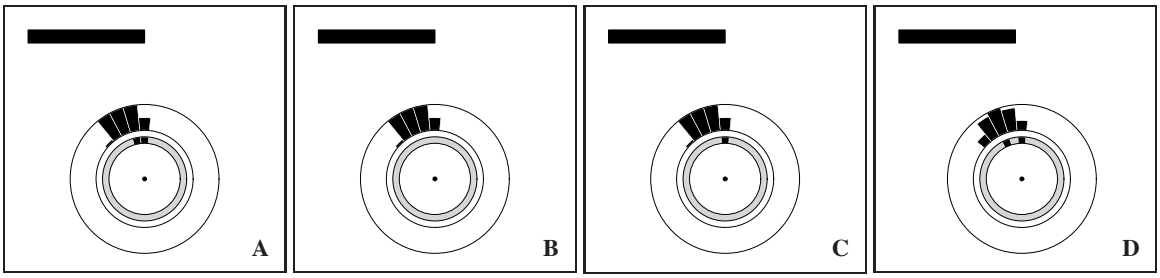

Fig. 9 Relation between opening angle, inter-sensor angle, and range of lateral inhibition. The black bar depicts a landmark with a detectable edge at the right side. The amount of light received by each of the 32 sensors (in this simulation scaled 0 to 1) is depicted by bars in the outer ring; the inter-sensor angle is $\beta = 11.25°$. Black fields in the grey ring visualize active edge pixels. **A:** Opening angle $\alpha = 11.25°$, edge detection threshold $\Theta = 0.4$, no lateral inhibition. Two neighboring pixels are active. **B:** $\alpha = 11.25°$, $\Theta = 0.6$, no lateral inhibition. No edge pixel is active. **C:** $\alpha = 11.25°$, $\Theta = 0.1$, unidirectional lateral inhibition between immediate neighbors. One pixel is active. **D:** $\alpha = 22.5°$, $\Theta = 0.1$, unidirectional lateral inhibition between immediate neighbors. Two non-neighboring pixels detect edges. In the analog circuit, Θ relates to V_{ref}.

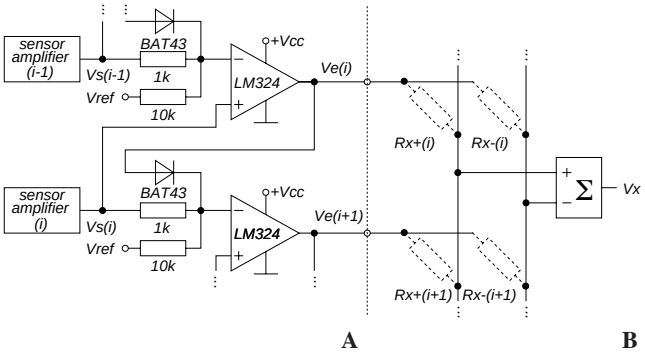

Fig. 8 **A:** Edge-filter circuit. **B:** Circuit for the computation of the x-component of the AL vector in robot coordinates. Dashed resistors encode the x-component for each landmark vector. The circuit for the y-component is identical except for the values of the resistor array.

in the following. In a real sensor, the opening angle can not be zero, i.e., the sensor will integrate over a certain visual field. In some cases, the visual field will include an edge. The sensor will then produce an intermediate signal as shown in Fig. 9. If the edge detection threshold is too small, two neighboring edge detectors will become active (A). On the other hand, if the threshold is increased, there will be cases when no edge is detected (B). With real visual signals, there is no way to adjust the threshold so that exactly one pixel is activated for each edge in the visual input.

This problem can be solved by introducing lateral inhibition, a fundamental mechanism in visual brains for which there is also evidence in the visual system of insects (Strausfeld and Nässel, 1981; Laughlin, 1981). With lateral inhibition, the threshold can be lowered to guarantee the safe detection of an edge: of the multiple candidates for edges, only one pixel will finally be activated. The range of lateral inhibition, i.e. the number of elements in the neighborhood that are affected by one pixel, depends on the relation between the opening angle of the sensors and the inter-sensor angle. For an environment with sharp visual edges like the experimental setup used in the robot experiments (Fig. 7, B), lateral inhibition can be restricted to immediate neighbors, if the opening angle α is smaller or equal to the inter-sensor angle β; see Fig. 9 (C). This is the case for the robot ($\alpha = 8°$, $\beta = 11.25°$). If the opening angle is larger (D), the range of lateral inhibition has to be increased: otherwise, the smooth transition of signals from maximal to minimal activation will cause multiple non-neighboring edge pixels to be activated.

In the edge-filter circuit (Fig. 8, A), lateral inhibition is realized by a diode connecting the output of an edge-filter with the negative input of one of the neighboring edge filters. A positive output voltage of the neighboring edge filter will set the negative input to a higher voltage than the positive, thereby switching off the comparator.

3.4 AL vector computation

Each active edge pixel contributes a radial landmark vector to the robot-centered AL vector. In the analog hardware, this is accomplished by connecting two resistors to the output of each edge filter that encode the two vector components. The resistors are part of a circuit implementing a weighted summation of the edge filter signals; Fig. 8 (B) shows the schematics for one vector component. Inactive edge elements have an output voltage of $0V$ and are therefore ignored in the summation; all active edge elements provide a constant positive output voltage and influence the AL vector according to the values of their two resistors. Since the landmark vectors can have positive and negative components, the resistors are either connected to the positive (R_{x+}) or negative (R_{x-}) input line of the adder circuit (Σ).

3.5 AL vector rotation and storage, home vector computation

The home vector is the difference between the AL vector of the current location and the AL vector of the target location, with both AL vectors relating to the same coordinate system. To compute a home vector in robot coordinates, the current AL vector provided by the circuit in Fig. 8 (B) can directly

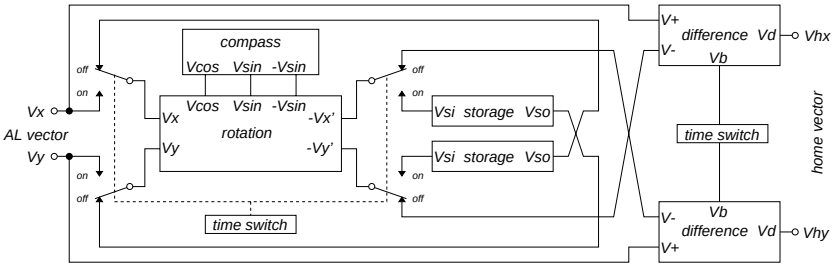

Fig. 10 Circuit for rotation and storage of AL vectors, and home vector computation. The 4 switches are shown in the "off" position used for homing; in the "on" position, the AL vector is rotated to world coordinates and stored. The blocks marked "difference" compute the difference $V_d = V_+ - V_-$; the time switch at V_b sets the home vector to zero, thereby stopping the movement.

be used in the difference computation, while the AL vector of the target location has to be rotated to the same coordinate system beforehand, since this vector was registered in a different orientation of the robot.

The circuit implementing rotation and storage of the AL vector as well as the computation of the home vector is presented in Fig. 10. Depending on the position of the switch block, the circuit accomplishes two transformations. In the target location, the switches are in position "on", whereby the robot-centered AL vector is rotated to world coordinates and stored. During homing, when the switches are "off", the stored vector is rotated back to robot coordinates according to the current orientation of the robot.

The transformation to world coordinates and back to robot coordinates can be accomplished with the same rotation circuit. The rotation angle depends on the signals of an electronic compass that measures the current orientation ϕ of the robot in world coordinates. The compass comprises two fluxgate magnetic field sensors arranged orthogonally to each other. Since the strength of the magnetic field is not constant in buildings, the signals obtained from the two sensors were normalized to constant length, thus providing two voltages proportional to the compass vector $(\cos\phi, \sin\phi)$. The "rotation" block in Fig. 10 is a straightforward implementation of the coordinate transformation

$$\mathbf{a}' = \mathbf{T}_\phi \mathbf{a} = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix} \mathbf{a} \qquad (10)$$

based on four precision multipliers and two adders. In the target location, where the robot is in orientation ϕ_0, the robot-centered AL vector $\mathbf{a}_0$ is transformed to world coordinates $\mathbf{a}'_0 = \mathbf{T}_{\phi_0}\mathbf{a}_0$ and stored. In the current location, with a robot orientation ϕ, the inverse transformation $\mathbf{a}''_0 = \mathbf{T}_\phi^{-1}\mathbf{a}'_0$ has to be applied to rotate the vector back to robot coordinates. Since $\mathbf{T}$ is orthogonal and therefore $\mathbf{T}^{-1} = \mathbf{T}^T$, the transposed matrix can be used in the inverse transformation. Multiplication by the transposed matrix was implemented with the same circuit by exchanging the components in the input and in the output vector; see the double x-shaped crossover in Fig. 10.

The components of $\mathbf{a}'_0$ are stored in two capacitors that are connected to switches and amplifiers with low leakage currents to achieve long storage times. Finally, the robot-centered AL vector of the target location $\mathbf{a}''_0$ is subtracted from the current AL vector in the two blocks marked "difference" in Fig. 10, which gives the home vector in robot coordinates.

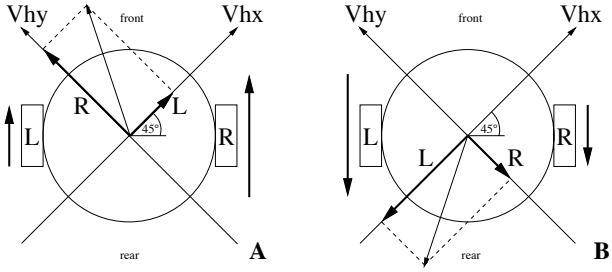

Fig. 11 Alignment of the robot coordinate system on the robot base. The boxes marked L and R depict the wheels. The home vector is shown as thin vector, its projection on the axes of the coordinate system (V_{hx}, V_{hy}) with thick arrows. The arrows next to the wheels represent speed and direction of wheel rotation. **A:** A home vector pointing towards the front of the robot will be stabilized in the front. **B:** A home vector pointing towards the rear of the robot will cause a movement of the robot that rotates the home vector towards the front.

3.6 Motor control

The robot uses differential steering, i.e., each wheel is independently driven by one motor; ball bearings support front and rear. The two components of the home vector can directly control the speed of the motors, if the robot coordinate system is aligned on the robot as shown in Fig. 11. The coordinate system is rotated by $45°$ so that the vector $(1, 1)$ points towards the front of the robot. The component corresponding to the axis pointing towards the left side of the robot determines the speed of the right wheel and vice versa. This arrangement will stabilize a home vector in the frontal direction (A); if the home vector is pointing to the rear, the robot will automatically turn around which rotates the home vector towards the front of the robot and thus changes the direction of movement (B).

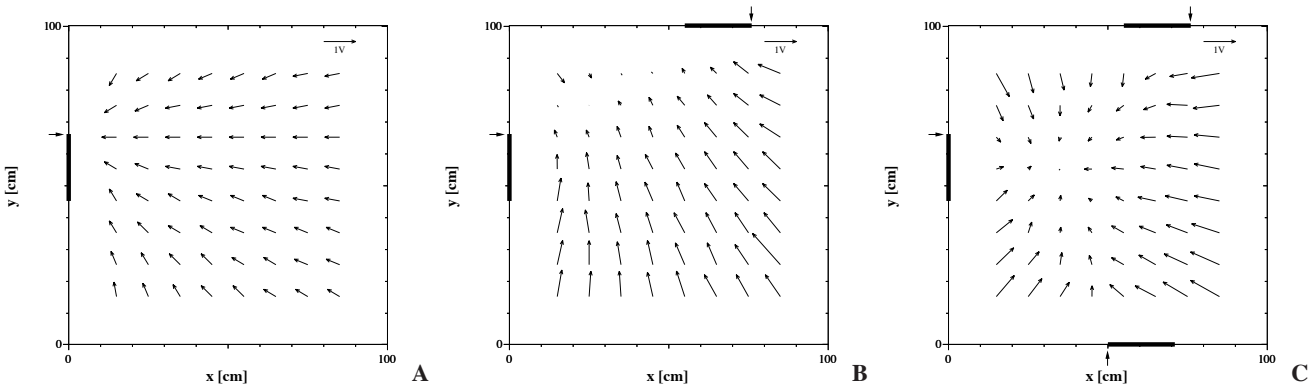

Fig. 12 AL vector voltages (V_x, V_y) measured at the output of the circuit in Fig. 8 (B) for 64 robot positions with one landmark (**A**), two landmarks (**B**), and three landmarks (**C**). Landmarks are depicted by bars; the detectable edges are marked with arrows. The vector scaling $(1V)$ is shown in the upper right corner.

4 Robot experiments

4.1 AL vector computation

The following experiments were done in an $1m \times 1m$ arena with white walls ($30cm$ high) and floor; see Fig. 7 (B). Black pieces of paper ($21cm \times 29cm$) attached upright to the walls served as landmarks. Light came from the ceiling lamps of the room. Fig. 12 shows the AL vector voltages measured (using multimeters with computer interface) while the robot was placed at 64 locations on a grid and aligned with the world coordinate system. For one landmark (A), all AL vectors have constant length and point to the detectable edge of the landmark. For two and more landmarks (B, C), the AL vectors also vary in length. Note that on a line connecting the two landmark cues in (B) the AL vectors are approximately zero. Target locations on this line can not be unambiguously identified, but with three or more landmarks which are not located on a line (C), the AL vector assigned to each location is unique (see Section 2.5.3 and Appendix A).

4.2 AL vector rotation and storage, home vector computation

Rotation and storage of the AL vector introduce errors, which will result in a shift of the approach point to the location that corresponds to the modified AL vector. The effect of errors in the AL vectors on the change of the approach point depends on the landmark configuration and on the position of the target point. Fig. 13 allows a graphical assessment for the setups used in the robot homing experiments. The diagrams were obtained by iteratively determining $\mathbf{x}$ for a given $\mathbf{A}(\mathbf{x})$; the effects of limited visual resolution were not considered. On each curve, either the angle or the length of the home vector is constant. The denser the resulting grid, the smaller is the shift of the approach point caused by a certain absolute error in angle or length of the AL vector. In general, position errors increase with increasing distance from the point where the AL vector is zero, and decrease with increasing number of landmarks.

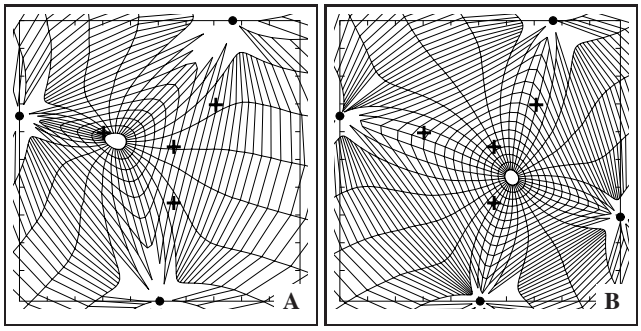

Fig. 13 Iso-length and iso-angle curves of the AL vector for three (**A**) and four landmarks (**B**). The angular distance between iso-angle curves is $10°$; iso-length curves differ by 0.1. Landmark positions (dots on the frame) and target positions (crosses) correspond to the setups used in the robot experiments. Compare (A) with Fig. 12 (C). Disruptions of the grid occur where the numerical procedure was not converging.

That the precision of the compass system is critical has been revealed in a first series of experiments, where the observed homing precision was insufficient. The original semi-mechanical compass had to be replaced by a fluxgate compass with higher precision. To assess the precision of the new compass system, the compass vector was measured while the robot was manually rotated on the spot by $360°$ in steps of $15°$. The standard deviation of the measured angle from the actual orientation of the robot was $2.1°$, the standard deviation of the length was 0.6%. The angular deviation can partly be caused by distortions of the earth magnetic field in the building: since the compass coils are not mounted in the center of the robot, they measure the field at slightly different positions during the rotation.

Errors introduced by the rotation circuit were estimated by rotating the robot on the spot (full turn, steps of $15°$) in a setup where two landmarks where visible under an angle of $80°$. For each orientation, both the robot-centered and the world-centered AL vector were measured (at (V_x, V_y) in the circuit in Fig. 8 (B), and at the outputs V_{so} of the vector stores in Fig. 10 in switch position "on", respectively). Mean values and standard deviations were computed for the AL vec-

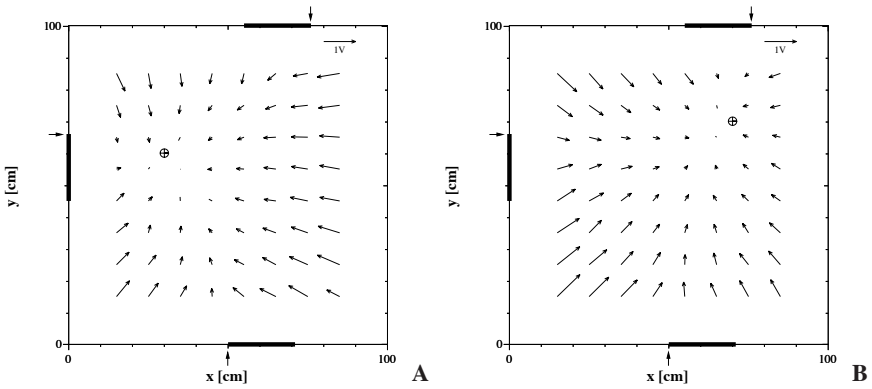

Fig. 14 Home vector fields (V_{hx}, V_{hy}) for two different target locations in the setup with three landmarks. Target locations are marked with a cross-circle. Landmarks and vectors are depicted as in Fig. 12.

tor before and after rotation to world coordinates; the robot-centered vector was aligned with the actual orientation of the robot beforehand. In the untransformed AL vector, the angle deviates by $3.2°$, the length by 2.8%. These deviations mainly result from the coarse image discretization, partly from errors in the resistors encoding the landmark vector components (Fig. 8, B). In the transformed AL vector, which should ideally be constant, the standard deviation is $3.1°$ and 2.8% for angle and length, respectively. A comparison demonstrates, that the errors introduced by the rotation circuit are certainly not larger and probably considerably smaller than the errors caused by the discretization of the image.

Measurements have shown that the vector store can be excluded as a major source of error: changes in the stored voltages are only $50\mu V/sec$ (vector components go up to $3V$). This allows experiments with the same stored vector for $15min$ and more without noticeably affecting the stored vector.

Fig. 14 shows two home vector fields obtained for two different target positions in a setup with three landmarks. Target and landmark positions were selected arbitrarily. The robot was first placed at the target location, and the AL vector of that location was stored. Then the robot was moved to 64 positions on a grid in the arena, where the two voltages of the home vector (V_{hx}, V_{hy}) were measured at the output of the circuit in Fig. 10. To test the rotation circuit, the orientation of the robot was changed by $60°$ between target point and grid points. It is clear that all home vectors point approximately to the target location; their length becomes shorter in the vicinity of the target.

4.3 Homing experiments

For the homing experiments presented in Fig. 15 (A–C), the robot was first placed at the target location, where the AL vector was registered, and then moved to different starting points, mostly close to the walls of the arena. A pen was mounted in the center of the robot between the two wheels. After some seconds, a time switch released the motors and the robot started to move. The trajectory was drawn by the pen on paper covering the floor of the arena; the trajectories were afterwards digitized from a photo of the paper (this method causes the small distortions of the diagrams). The V-shaped parts in some of the trajectories result from changes in the movement direction of the robot from backwards to forward (see Section 3.6).

For the three experiments shown in Fig. 15, the mean deviation of the final points from the target point was $64 \pm 11mm$ $(n = 11)$ in (A), $58 \pm 17mm$ $(n = 11)$ in (B), and $36 \pm 15mm$ $(n = 8)$ in (C), with n denoting the number of return trips. The experiment was repeated for three and four landmarks in four different target locations (Fig. 13). The mean deviation was in the range from $47 \pm 16mm$ $(n = 10)$ to $95 \pm 38mm$ $(n = 10)$ in the 9 experiments done in the setup with three landmarks and from $36 \pm 14mm$ $(n = 8)$ to $57 \pm 17mm$ $(n = 8)$ in the 4 experiments done in the setup with four landmarks. Over all runs, the mean deviation was $68 \pm 35mm$ $(n = 99)$ for the setup with three landmarks and $48 \pm 21mm$ $(n = 28)$ for the setup with four landmarks. The improved precision of homing with an additional landmark is significant (one-tailed Mann-Whitney U-test, $p = 0.004$). Three runs where the robot collided with the walls of the arena and could not recover from this state have been disregarded.

Fig. 16 shows the behavior of the robot in experimental setups similar to those used in bee experiments by Cartwright and Collett (1983) (there: Fig. 9a) and Anderson (1977) (there: Fig. 1). Cartwright and Collett (1983) trained bees to an array of three landmarks and tested them in another array where the distances between the landmarks were halved. They observed that "during the tests, the bees would search were the compass bearings of the landmarks on its retina matched those experienced at the food source during training". The same behavior is shown by the robot: all final positions of the trajectories in the test setup end in the location where also the bees search preferably (Fig. 16, A, B). Anderson (1977) used a circle of eight landmarks and removed three of the landmarks between training and test. As a result, the search position of the bees shifted inside the semi-circle. The final positions of

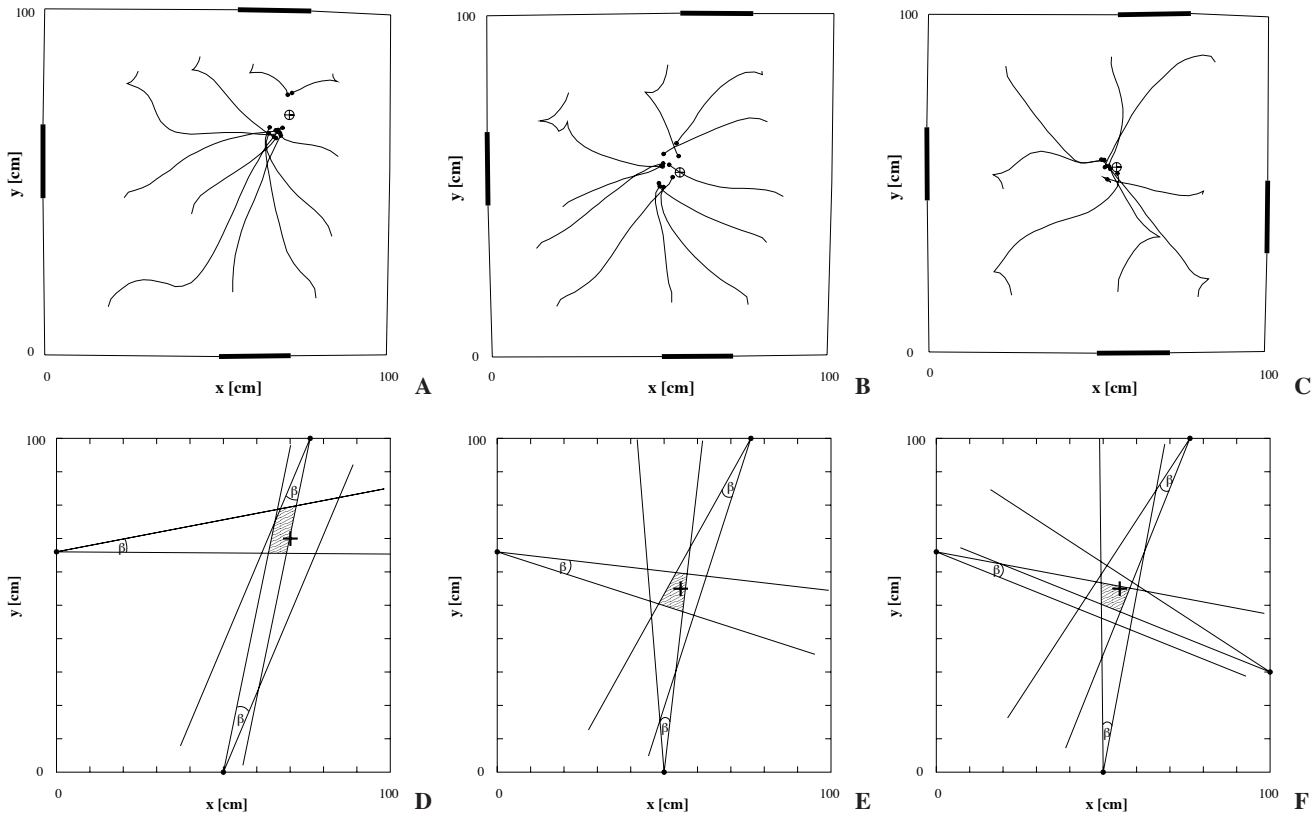

Fig. 15 A–C: Robot trajectories for two target locations in the setup with three landmarks, and for one target location in the setup with four landmarks. The cross-circle marks the target location where the AL vector was stored. Black bars depict the landmarks, small dots the final positions of the robot. **D–F:** Geometrical reconstruction of regions where the perceived view does not change (hatched area) for each of the three experiments. Dots on the margin depict landmarks, the cross marks the target position. $\beta = 11.25^\circ$ is the inter-sensor angle.

the robot also lie inside the semi-circle, although the shift is more pronounced in the robot experiments (Fig. 16, C, D) [1].

5 Discussion

5.1 Precision of homing

The mean deviations of the final points from the target point can almost entirely be explained as an effect of the low visual resolution of the robot. In a relatively large area around the target point, the robot perceives the same image. Consequently, the target location can not be located with higher precision. These "iso-view" regions have been constructed geometrically for the experiments in Fig. 15 (D–F). The view sector of each active edge pixel — which has an angular size equal to the inter-sensor angle $\beta = 11.25^\circ$ — was attached to the corresponding landmark; the cross section of all view sectors gives the iso-view region (hatched area). Depending on the orientation of the robot, different sensors are activated which results in iso-view regions with similar size but different shape; in the geometrical reconstruction, the orientation was chosen in a way that the shape of the iso-view region was in accordance with the final points of the trajectories obtained in the experiments. It is clear that iso-view regions can be found where the final points are located close to the margin or inside the regions. Since the mean deviations in *all* experiments are in the same range as the mean deviations in the experiments in Fig. 15, it is likely that also for the other experiments iso-view regions can be constructed which explain the position errors.

The improved homing precision with increasing number of landmarks ($68mm$ mean deviation for three landmarks compared to $48mm$ for four landmarks) could be the result of two effects. First, the AL vector grid (formed by iso-angle and iso-length curves) becomes more dense (compare Fig. 13 A and B), and, second, the iso-view region (Fig. 15, D–F) becomes smaller (or remains unchanged) with an additional landmark. As stated above, the precision of homing seems to be mainly determined by the given visual resolution. This may change with improved visual resolution: the area of the iso-view regions will decrease, and the homing error will mostly be influenced by the precision of the compass system, depending on the density of the AL vector grid in the target location. For comparison: the inter-ommatidial angles of desert ants *Cataglyphis bicolor* (Zollikofer et al., 1995) in the horizontal direction are a factor of $2-3$ smaller than the inter-sensor angles of the robot.

[1] The starting points in the experiment in Fig. 16 (B) had to be moved closer to the landmarks, since the landmarks and the gaps between them had to be reduced in size and the low visual resolution causes the edge detection to fail in larger distances from the landmarks.

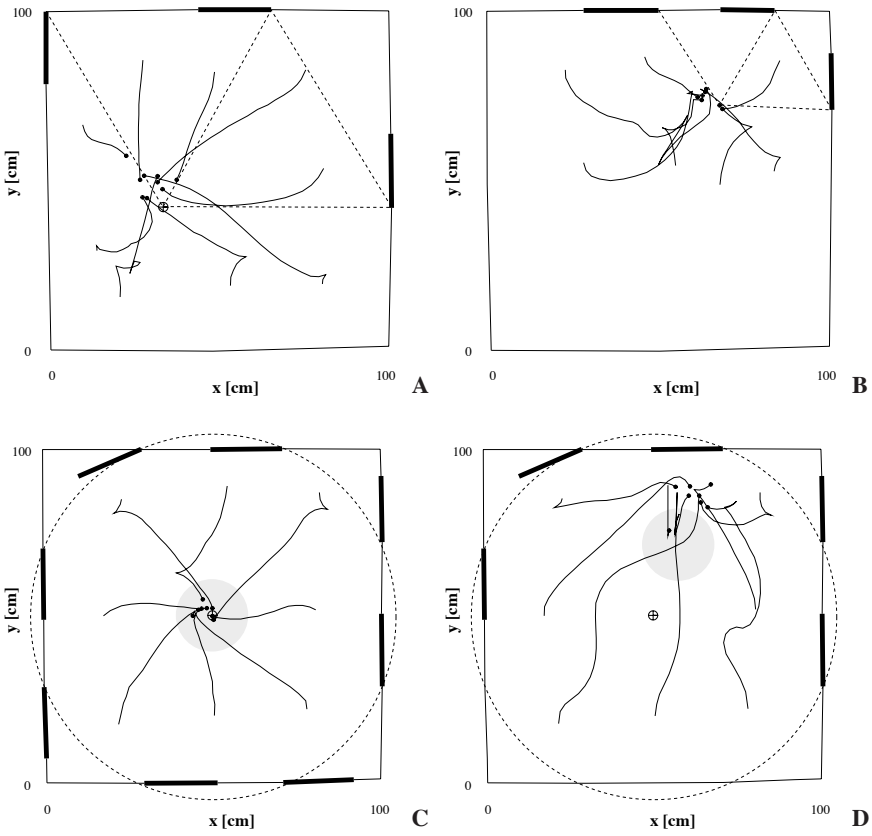

Fig. 16 Robot trajectories in experimental setups similar to those used in bee experiments by Cartwright and Collett (1983) (**A**, **B**) and Anderson (1977) (**C**, **D**). Left column: test with unaltered training setup; the target location is marked with a cross-circle. Right column: test with altered setup. The dashed lines in A and B connect the target position in the training and the expected target position in the test with the detectable edges of the landmarks. All detectable edges in C and D are located on the dashed circle. The grey spots in C and D show the approximate locations where the bees' search concentrated.

In summary, the data show that homing is as precise as possible with the available visual resolution. The second factor limiting the precision of homing is the internal rotation of the AL vector: as mentioned in Section 4.2, a redesign of the compass system was necessary in order to reduce the errors introduced by rotation.

5.2 Reproduction of bee experiments

In two experiments described in Section 4.3 (Fig. 16), the robot approached approximately the same location in the tests where bees also concentrated their search. These experiments demonstrate that the version of the ALV model which was implemented on the robot exhibits correspondences with the behavior observed in insects. The experiment in Fig. 16 (A, B) demonstrates the relation between ALV model and snapshot model (Section 2.2): the robot seems to restore the *image* perceived in the target location. The experiment in Fig. 16 (C, D) provides support for the link to the concept of "surroundedness" (Section 2.3). The robot apparently only memorizes an *overall measure* of how it is surrounded by landmarks, the "center of gravity" of all landmark cues in the image, in this case expressed by a zero AL vector. In the test, it moves to the location inside the remaining landmarks, where this measure (but not the image) is identical.

The simplicity of the implemented model restricts the experiments that can be reproduced. First, landmark size is not detected; this may also account for the deviation between the final points of the robot trajectories and the region were the bees search in the experiment with the semi-circle (Fig. 16, D). A measure of size could be gained by considering edges of both polarities (Lambrinos et al., 1999), but this would have required additional hardware effort. Second, the robot does not compute the average landmark vector, but only the sum of all landmark vectors (see Section 2.5.1). This simplification leads to completely different behavior when the number of landmarks changes between training and test, as was shown in Fig. 4. The semi-circle experiment could be reproduced despite changes in the number of landmarks, since the stored AL vector was zero, and therefore the missing normalization had no effect.

To find a version of the ALV model that optimally reproduces the behavioral data, a systematic variation of its structure and parameters is necessary; for this purpose, a computer simulation is more suitable. The robot experiments provide the basis for computer simulations, since they demonstrate that, first, the model in general does not rely on assumptions

which are not fulfilled in the real world, and, second, it does not exhibit a behavior which widely differs from the data of behavioral experiments.

5.3 Analogies to neural architectures

Correspondences between analog electronic hardware and nervous systems have been pointed out in Section 1.3. In the analog circuit implementing the ALV model, strong similarities can be found in the first processing stages. These similarities concern general properties of insect nervous systems; nothing is known so far about the neural circuits that realize visual homing in insect brains.

The selection of edges as landmark features is supported by the fact that "the landmark guidance system of bees can operate successfully using edges alone" as was shown in experiments by Cartwright and Collett (1983). Lateral inhibition, a principle underlying both the edge filters in the circuit and the interaction between them, is common to many visual systems; in insect brains, certain neurons in the lamina are supposed to mediate lateral inhibition (Strausfeld and Nässel, 1981; Laughlin, 1981). The specific way in which edge filters and lateral inhibition interact in the circuit is more a caricature of a system that would enable detection of edges in complex visual scenes. The artifical setup with clearly detectable edges and constant light conditions is probably the weakest point of this work, but the visual resolution and the simplicity of the circuit put some restrictions on the properties of the environment. Feature-detecting circuits have to be developed where a group of neighboring neurons responds with constant total activity to one specific feature, as it is required by the model (see Section 2.5.2).

The computation of the AL vector from the outputs of the feature-detecting circuit finds a biological counterpart in the spatial summation of input signals, which is a fundamental function of neurons. It is very unlikely, though, that synaptic weights could be genetically determined to encode vector components. An alternative was suggested by Möller et al. (1999): Since the influence of the pre- on the postsynaptic neuron also depends on the length of the neural processes connecting both neurons, vector components could be encoded solely by a specific spatial arrangement of the cells. Neurons involved in the AL vector encoding would probably be identified by electrophysiology as "large-field neurons" as they are known to exist in the medulla (Strausfeld and Nässel, 1981), since they will respond to visual stimuli over a large portion of ommatidia. Cartesian coordinates have been arbitrarily chosen in the model, but this decision does not entail problems; in fact, the two coordinates can directly influence the motor system without a transformation to another system by a simple crossover of fibers (see Section 3.6).

The circuit used for the rotation of the AL vector will probably not have a direct biological counterpart. The use of precision multipliers is a technical solution, which is not possible given the noise and imprecision of the neural substrate. A more likely solution is a topological encoding of the vector, such as the ring structures used in the neural model of path integration by Hartmann and Wehner (1995). Which type of compass guides the rotation is not crucial for the model: it could be a magnetic compass that is known to be used by bees for landmark navigation (Collett and Baron, 1994) or a polarized light compass employed for instance by desert ants (Wehner, 1994).

Experiments have shown that *Cataglyphis* ants can store landmark-based information used in pin-pointing the nest entrance over at least 20 days (Ziegler and Wehner, 1997). Such long-term storage will presumably be based on modifications at the synapses rather than on self-stabilizing neural feedback circuits as suggested by Hartmann and Wehner (1995). The current storage solution based on capacitors can only guarantee storage times in the range of $15min$. An alternative would be the use of non-volatile analog memory storage based on floating-gate silicon MOS transistors (see e.g. Diorio et al., 1997).

It may also be the case that not the vector components are stored, but a representation with a closer relation to the image — the vector encoding could only be exploited for the computation of the home vector, which would save the matching process. Evidence indicating that the image information is not disregarded comes from decision experiments. Bees had to decide between two landmark setups where only the images perceived in the expected target locations differed but the AL vectors would not (Cartwright and Collett, 1983, Fig. 15); the bees preferred the array with the "correct" image.

In general, the ALV model may define the lower limit of complexity for neural circuits that accomplish visual homing. The number of neurons and synaptic connections required for a neural implementation of the model is very small. Visual input and AL vector are linked by a feature-detecting circuit with only local interconnections and a single convergent pathway (see Fig. 6). The majority (64) of the 91 operational amplifiers of the ALV circuit belong to the retinotopically organized part of the circuit (from sensor signal to edge-filter output), even though the feature detection circuit is very simple. This indicates that visual homing mechanisms might mainly be implemented close to the periphery of the insect visual system. About the complexity of the neural circuits mediating the alignment of vectors or images only speculations are possible, since so far it is not even clear on the functional level, how the alignment is realized by insects; experiments show that landmark images seem to be retinotopically bound (Wehner et al., 1996).

6 Conclusions and future work

It was shown that the ALV model successfully guides a mobile robot to a target location with only small positional errors. The precision of homing is limited by visual resolution and precision of the compass system. In some landmark setups, the behavior of the robot resembles the behavior of bees. The analog circuit that implements the ALV model on the robot provides some insights as to how visual homing might be implemented in insect brains.

Future work will try to eliminate the limitations posed by the low visual acuity and the simple circuits for landmark detection, which restrict the operation to artificial landmarks and room light conditions. For larger environments with ambiguous places, the system has to be extended so that multiple AL vectors can be stored. Larger distances could then be covered by approaching the corresponding target locations in a sequence; the stored vectors will thus form a "route map" of the environment.

Acknowledgments. The project is supported by the Swiss National Science Foundation (grants 2000-053915.98 and 5002-044889 to Rolf Pfeifer), the Swiss Federal Office for Education and Science (VIRGO TMR network), the Human Frontier Science Program, and by a personal grant from the Kommission zur Förderung des akademischen Nachwuchses der Universität Zürich. Many thanks to Bernhard Schmid and colleagues, Sonja Bisch-Knaden, Koh Hosoda, Markus Knaden, Jürgen Knobloch, Hiroshi Kobayashi, Axel Könies, Dimitrios Lambrinos, Marinus Maris, and Peter Paschke for their valuable contributions.

A Proof of global convergence

$\mathbf{H}(\mathbf{x})$ is a gradient system, i.e., there exists a scalar potential $U(\mathbf{x})$ which satisfies $\mathbf{H}(\mathbf{x}) = -\text{grad}\, U(\mathbf{x})$. The potential is given by $U(\mathbf{x}) = \sum_{i=1}^{n} U_i(\mathbf{x})$ with

$$U_i(\mathbf{x}) = ||\mathbf{x}_i - \mathbf{x}|| - \frac{\mathbf{x}_i - \mathbf{x}_0}{||\mathbf{x}_i - \mathbf{x}_0||}(\mathbf{x}_i - \mathbf{x}).$$

The second order derivatives of $U(\mathbf{x})$ are

$$U_{xx} = \sum_{i=1}^{n} \frac{\eta_i^2}{\lambda_i}, \;\; U_{yy} = \sum_{i=1}^{n} \frac{\xi_i^2}{\lambda_i}, \;\; U_{xy} = -\sum_{i=1}^{n} \frac{\xi_i \eta_i}{\lambda_i},$$

with $\eta_i = y_i - y, \xi_i = x_i - x, \lambda_i = ||\mathbf{x}_i - \mathbf{x}||^3$, and $\mathbf{x} = (x, y)$. The determinant of the Jacobian matrix of $U(\mathbf{x})$ is

$$D = U_{xx}U_{yy} - U_{xy}^2 = \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\eta_i^2 \xi_j^2}{\lambda_i \lambda_j} - \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\xi_i \eta_i \xi_j \eta_j}{\lambda_i \lambda_j},$$

and can be rewritten as

$$D = \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\eta_i \xi_j (\eta_i \xi_j - \eta_j \xi_i)}{\lambda_i \lambda_j}.$$

Elements in the sum with $i = j$ disappear, for the remaining elements the sum can be split:

$$D = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \frac{\eta_i \xi_j (\eta_i \xi_j - \eta_j \xi_i)}{\lambda_i \lambda_j} + \sum_{j=1}^{n-1}\sum_{i=j+1}^{n} \frac{\eta_i \xi_j (\eta_i \xi_j - \eta_j \xi_i)}{\lambda_i \lambda_j},$$

which after exchanging indices of the second sum and factoring out gives

$$D = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \frac{(\eta_i \xi_j - \eta_j \xi_i)^2}{\lambda_i \lambda_j}.$$

The following condition guarantees that $D(\mathbf{x}) > 0$:

$$\exists i, j \; (i \neq j) : (y_i - y)(x_j - x) - (y_j - y)(x_i - x) \neq 0.$$

Geometrically this means that at least two of the landmarks should not lie on the same line through $\mathbf{x}$. If this condition is fulfilled for $\mathbf{x}_0$, $U(\mathbf{x})$ has an isolated local minimum at the fixed point $\mathbf{x}_0$ of equation (4), since $D(\mathbf{x}_0) > 0, U_{xx}(\mathbf{x}_0) > 0$, and grad $U(\mathbf{x}_0) = \mathbf{0}$.

$U_i(\mathbf{x})$ is the sum of a norm function and a plane and therefore convex in the whole plane. Since the sum of convex functions is convex, also $U(\mathbf{x})$ is a convex function. An isolated local minimum $\mathbf{x}_0$ of a convex function is the only local and therefore the global minimum of that function. From this it can be concluded that all trajectories following the negative gradient will run into $\mathbf{x}_0$. Trajectories passing one of the landmark positions $\mathbf{x}_i$ have to be treated separately, since the gradient is not defined in these points. Note that this proof can also be extended to more than 2 dimensions.

Since $\mathbf{x}_0$ is the only local extremal point of $U(\mathbf{x})$, this point is the only point where the gradient $\mathbf{H}(\mathbf{x})$ becomes zero, i.e., $\mathbf{H}(\mathbf{x}) \neq \mathbf{0} \; \forall \mathbf{x} \neq \mathbf{x}_0$. Equation (3) therefore yields $\mathbf{A}(\mathbf{x}) \neq \mathbf{A}(\mathbf{x}_0) \; \forall \mathbf{x} \neq \mathbf{x}_0$: there is no second point in the plane with the same AL vector.

References

Anderson AM (1977) A model for landmark learning in the honey-bee. J Comp Physiol A 114:335–355

Cartwright BA, Collett TS (1983) Landmark learning in bees. J Comp Physiol A 151:521–543

Collett TS, Baron J (1994) Biological compasses and the coordinate frame of landmark memories in honeybees. Nature 368:137–140

Collett TS, Land MF (1975) Visual spatial memory in a hoverfly. J Comp Physiol 100:59–84

Cruse H, Bartling C, Cymbalyuk G, Dean J, Dreifert M (1995) A modular artificial neural net for controlling a six-legged walking system. Biol Cybern 72:421–430

Diorio C, Hasler P, Minch BA, Mead C (1997) A complementary pair of four-terminal silicon synapses. Analog Integrated Circuits and Signal Processing 13:153-166

Douglas R, Mahowald M, Mead C (1995) Neuromorphic analogue VLSI. Ann Rev Neurosci 18:255–281

Franceschini N, Pichon JM, Blanes C (1992) From insect vision to robot vision. Phil Trans R Soc Lond B 337:283–294

Franz MO, Schölkopf B, Mallot HA, Bülthoff HH (1998) Where did I take that snapshot? Scene-based homing by image matching. Biol Cybern 79:191–202

Grasso F, Consi T, Mountain D, Atema J (1996) Locating odor sources in turbulence with a lobster inspired robot. In: Maes P, Mataric MJ, Meyer J-A, Pollack J, Wilson SW (eds) From Animals to Animats 4. MIT Press Cambridge, Ma, pp 104–112

Hartmann G, Wehner R (1995) The ant's path integration system: a neural architecture. Biol Cybern 73:483–497

Lambrinos D, Maris M, Kobayashi H, Labhart T, Pfeifer R, Wehner R (1997) An autonomous agent navigating with a polarized light compass. Adapt Behav 6(1):175–206

Lambrinos D, Möller R, Pfeifer R, Wehner R (1998) Landmark navigation without snapshots: the average landmark vector model. In: Elsner N, Wehner R (eds) Proc. Neurobiol. Conf. Göttingen. Georg Thieme Stuttgart, p 30a

Lambrinos D, Möller R, Labhart T, Pfeifer R, Wehner R (1999) A mobile robot employing insect strategies for navigation. Robotics and Autonomous Systems, special issue: Biomimetic Robotics (to appear)

Lambrinos D (1999) Navigation in biorobotic agents. PhD thesis, Department of Computer Science, University of Zurich, Switzerland

Laughlin S (1981) Peripheral visual systems of invertebrates. In: Autrum H (ed) Handbook of Sensory Physiology, vol VII/6B. Springer Berlin, pp 133–280

Mead C (1989) Analog VLSI and Neural Systems. Addison-Wesley Reading, Ma

Möller R, Maris M, Lambrinos D (1999) A neural model of landmark navigation in insects. Neurocomputing 26-27, pp 801-808

O'Keefe J (1991) The hippocampal cognitive map and navigational strategies. In: Paillard J (ed) Brain and Space. Oxford University Press, pp 273–295

Srinivasan MV, Chahl JS, Zhang SW (1997) Robot navigation by visual dead-reckoning: inspiration from insects. Int J Pattern Recognition and Artificial Intelligence 11(1):35–47

Strausfeld NJ, Nässel DR (1981) Neuroarchitectures serving compound eyes. In: Autrum H (ed) Handbook of Sensory Physiology, vol VII/6B. Springer Berlin, pp 1–132

Webb B (1995) Using robots to model animals: a cricket test. Robotics and Autonomous Systems 16(2-4):117–134

Wehner R (1994) The polarization-vision project: championing organismic biology. In: Schildberger K, Elsner N (eds) Neural Basis of Adaptive Behaviour. Fischer Stuttgart, pp 103–143

Wehner R, Räber F (1979) Visual spatial memory in desert ants, *Cataglyphis bicolor* (Hymenoptera: Formicidae). Experientia 35:1569–1571

Wehner R, Michel B, Antonsen P (1996) Visual navigation in insects: coupling of egocentric and geocentric information. J Exp Biol 199:129–140

Ziegler PE, Wehner R (1997) Time-courses of memory decay in vector-based and landmark-based systems of navigation in desert ants, *Cataglyphis fortis*. J Comp Physiol A 181: 13–20

Zollikofer CPE, Wehner R, Fukushi T (1995) Optical scaling in conspecific *Cataglyphis* ants. J Exp Biol 198:1637–1646