

A Self-Organizing Neural Network Architecture for Navigation Using Optic Flow

Seth Cameron

Stephen Grossberg

Frank H. Guenther

Department of Cognitive and Neural Systems and Center for Adaptive Systems, Boston University, Boston, MA 02215, U.S.A.

This article describes a self-organizing neural network architecture that transforms optic flow and eye position information into representations of heading, scene depth, and moving object locations. These representations are used to navigate reactively in simulations involving obstacle avoidance and pursuit of a moving target. The network's weights are trained during an action-perception cycle in which self-generated eye and body movements produce optic flow information, thus allowing the network to tune itself without requiring explicit knowledge of sensor geometry. The confounding effect of eye movement during translation is suppressed by learning the relationship between eye movement outflow commands and the optic flow signals that they induce. The remaining optic flow field is due to only observer translation and independent motion of objects in the scene. A self-organizing feature map categorizes normalized translational flow patterns, thereby creating a map of cells that code heading directions. Heading information is then recombined with translational flow patterns in two different ways to form maps of scene depth and moving object locations. Most of the learning processes take place concurrently and evolve through unsupervised learning. Mapping the learned heading representations onto heading labels or motor commands requires additional structure. Simulations of the network verify its performance using both noise-free and noisy optic flow information.

1 Introduction: Optic Flow, Heading, and Visual Navigation

As we move through the world, we experience flowing patterns of light on our retinas. Scientists have studied the ability of humans to use this optic flow for a variety of tasks, including determination of heading (e.g., Crowell & Banks, 1993; Van den Berg, 1992, 1993; Warren & Hannon, 1988, 1990; Warren & Kurtz, 1992), observer velocity (e.g., Monen & Brenner, 1994), three-dimensional structure (e.g., Cornilleau-Peres & Droulez, 1993; Hildreth, Grzywacz, Adelson, & Inada, 1990; Treue, Andersen, Ando, & Hildreth, 1995), the locations and speeds of self-moving objects (e.g., Brenner,

1991), and distances to objects (e.g., Prazdny, 1980; Simpson, 1988, 1993). With the exception of determining observer velocity (Monen & Brenner, 1994), humans have proved to be very proficient in using optic flow to perform these tasks.

This article describes a neural network that uses optic flow information and eye position to determine the heading of a moving observer. This heading network is embedded in a larger architecture that includes neural networks for determining distances to visible objects and the locations of moving objects. The resulting representations of heading, scene depth, and moving object locations are then used to navigate reactively around visible obstacles and to pursue moving targets. The aim of this project is to use insights from biology to design a robust system that can be used for navigation in autonomous mobile vehicles. To this end, the system is designed as an interconnected collection of self-organizing neural networks. The primary advantage of this approach is that it removes the need to hand-calibrate visual navigation systems for mobile robots. The network will automatically adjust to a camera system's focal length, pixel layout, sensor size, and tracking speed without writing new software or performing carefully controlled calibrations. This competence also renders the system tolerant to error in sensor responses and changes in sensor and motor properties due to wear.

A central problem addressed by the current architecture is the determination of the heading direction of a moving observer from the optic flow field. Gibson (1950) remarked that the optical flow pattern experienced by an observer moving along a straight line through a static environment contains a singularity he termed the focus of expansion (FOE). Gibson further hypothesized that the visual system might use the global pattern of radial outflow originating from the singularity to determine the translational heading of the observer. However, psychophysical experiments rejected this hypothesis because the flow pattern on the retina is radically altered by eye movements (Regan & Beverly, 1982). If the observer's eyes rotate during translational movement, the resulting flow pattern is a superposition of two vector fields such that the FOE no longer necessarily coincides with heading direction. The current architecture learns to cancel the effects of eye rotations so that a moving robot can maintain an accurate sense of heading while visually searching a scene or visually tracking objects.

The issue of whether humans use extraretinal signals from eye rotations to derotate the flow field has been hotly debated in the experimental literature, since it raises difficult methodological questions concerning the nature of the motion, its range of speeds, and the types of environmental cues that are available (Banks, Ehrlich, Backus, & Crowell, 1996; Regan & Beverly, 1982; Rieger & Toet, 1985; Royden, 1994; Royden, Banks, & Crowell, 1992; Royden, Crowell, & Banks, 1994; van den Berg, 1993; Warren, 1995; Warren & Hannon, 1988, 1990; Warren, Li, Ehrlich, Crowell, & Banks, 1996). Warren (1996) has summarized much of the relevant literature by noting that extraretinal information is especially important in determining heading from

impoverished visual environments, but less so in naturalistic scenes with a rich, discontinuous three-dimensional structure. Our model indicates the types of self-tuning mechanisms that are able to make effective use of extraretinal information.

Since Gibson wrote, vision researchers have proposed a wide variety of methods to extract ego motion from optical flow information.

The differential invariants method (Koenderink & van Doorn, 1975; Waxman & Ullman, 1985) solves for the motion of planar surfaces. A rigidly moving planar surface is a special case that gives rise to flow fields in which the flow vectors vary as a quadratic function of image position. Waxman and Ullman (1985), for example, first subdivided the flow field into patches that are approximately planar. The three-dimensional structure was then recovered in closed form from the parameters of the second-order flow field. An advantage of this technique is that the motion in each patch is computed independently, so scenes with multiple moving objects can be better interpreted. One drawback is that it requires dense optic flow fields to compute derivatives. Systems that use differential invariants are also vulnerable to error in the flow field (Hatsopoulos & Warren, 1991). Optic flow error is essentially unavoidable due to the aperture problem (e.g., Perrone, 1992).

According to the local differential motion method (Cutting, 1986; Hildreth, 1992; Longuet-Higgins & Prazdny, 1981; Rieger & Lawton, 1985), if two points have the same image location but are at different depths, then the vector difference between the two flow vectors is oriented toward the FOE. This technique relies on locating adjacent image features separated in depth so that their flow vector difference is oriented approximately toward the FOE. It fails to operate when approaching a wall with no depth variation, and as with the differential invariant techniques, it does not work well on sparse or noisy flow fields.

Heeger and Jepson (1990) proposed a solution to the heading calculation problem that uses *least-square minimization*. Their technique, termed the *subspace algorithm*, is robust and comparatively insensitive to noise. It has the advantage of not requiring proprioceptive information from eye muscles to cancel the effects of eye rotations, but it does not learn its operating parameters.

Several researchers have recently posited *neural network models* for determining optic flow. Because of the parallel processing capabilities embodied in these networks, neural network solutions tend to be more tolerant of noise in the flow field. Hatsopoulos and Warren (1991) describe a neural network motivated by neurophysiological results from the middle temporal (MT) and medial superior temporal (MST) areas in primate visual cortex. Although the model is tolerant to both speed and directional noise in the flow field, the authors point out that it has two major shortcomings: (1) unlike humans, the model's accuracy degrades drastically in the presence of eye rotations, and (2) the model relies on supervised learning (i.e., it requires an external teaching signal that would not be available to a developing ani-

mal). Perrone (1992) incorporated both translational and rotational detector cells in a model that offers an approximate solution to heading detection during eye rotations. Lappe and Rauschecker (1993) describe a neural network based on the algorithm of Heeger and Jepson (1990), but this network requires tracking fixation eye movements to determine heading and operates only within a hemisphere of heading directions. The Heeger and Jepson algorithm is forced to operate within a hemisphere of headings because it cannot distinguish between motions separated by 180 degrees. It relies on locating a minimal inner product between sets of heading weights and incoming flow vectors. The minimum occurs for weights that are orthogonal to the flow vectors. However, if a heading is reversed, the flow vectors will reverse, yet remain orthogonal to the weight vector. The Heeger and Jepson algorithm cannot distinguish opposite motions, making it difficult to use on robots with full translational capabilities. The current system can operate over the full sphere of heading directions.

In addition to determining heading, the current architecture represents the locations of visible obstacles for the purpose of navigating around them. The machine vision community has published hundreds of articles about recovering three-dimensional structure from image motion. In general, the focus of this effort has been to build scene representations that reflect ground truth as accurately as possible. Over the years, a multiplicity of algorithms have emerged that attempt to deal with the inherent ambiguity in the task (e.g., sparse data and the aperture problem). Since the goal has been to reproduce scene geometry as accurately as possible, algorithmic simplicity and self-tuning have not been emphasized. Rather, most techniques assume that all relevant camera parameters are available. In fact, camera calibration is itself an active research topic. Most camera calibration techniques rely on imaging features at known locations (Holt & Netravali, 1991). Other techniques have been designed to use arbitrary scenes but require known movements (Dron, 1993), and more recently, a few techniques have attempted calibration without specific objects or movements (Luong & Faugeras, 1992).

Despite the impressive amount of depth-from-motion research, it is still unclear whether identically reproducing the three-dimensional structure of the world is either feasible or necessary for autonomous robot navigation. The complexity, fragility, and computational load of most of these algorithms have not produced the desired results in mobile robotics. Only recently have researchers begun to concern themselves with how to solve visual navigation problems using simple, biologically plausible calculations embedded in self-tuning systems. Although some initial work has been done on very simple problems (Heikkonen & Oja, 1993), the current system stands apart by using simple self-tuning calculations to learn representations of heading, scene structure, and moving object locations in the presence of eye rotations. The architecture uses parallelism to capitalize on the redundancy in the flow field in order to achieve noise tolerance. No claim is made that this system will provide complete and accurate results

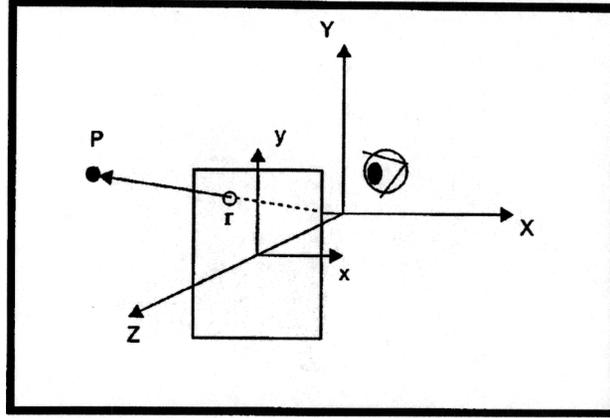


Figure 1: Viewer-centered coordinate frame and perspective projection.

in all situations. In fact, no system that relies solely on optic flow can produce a veridical representation of the world that it can use for completely error-free navigation. However, the current system's performance on simulations with noisy flow information suggests that its parallel, self-tuning, and computationally efficient calculations are sufficient to provide useful information for an autonomous robot.

2 The Optic Flow Field

Optic flow results from the projection of object movement in the three-dimensional world onto a two-dimensional image plane. The motion of a point in three-dimensional space has six degrees of freedom: translational velocity $\mathbf{T} = (T_x, T_y, T_z)^t$ and rotation $\mathbf{\Omega} = (\Omega_x, \Omega_y, \Omega_z)^t$. If the position of an observer's eye is $\mathbf{R} = (X, Y, Z)^t$, then motion may be described by $\mathbf{V}_{\text{observer}} = \mathbf{T} + (\mathbf{\Omega} \times \mathbf{R})$. If the observer is moving through a static environment, a viewer-centered coordinate frame may be established in which the observer appears to be standing still and each point in the environment is moving with the opposite motion, $\mathbf{V} = -(\mathbf{T} + \mathbf{\Omega} \times \mathbf{R})$.

Figure 1 shows how a point, P , in three-dimensional space is projected onto a point $\mathbf{r} = (x, y)^t = f(X/Z, Y/Z)^t$ in the image plane, where f denotes the focal length of the lens ($f = 1$ was used in the simulations). Image velocity is defined as the derivative, with respect to time, of the x and y components of scene point projections and can be written as follows (Heeger & Jepsen, 1990):

$$\theta(x, y) = (v_x, v_y) = p(x, y)\mathbf{A}(x, y)\mathbf{T} + \mathbf{B}(x, y)\mathbf{\Omega}, \quad (2.1)$$

where $p(x, y) = 1/Z$ is the inverse depth of points in the image plane, and $\mathbf{A}(x, y)$ and $\mathbf{B}(x, y)$ are matrices that depend only on image position and focal length, not on any unknowns:

$$\mathbf{A}(x, y) = \begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix}$$

$$\mathbf{B}(x, y) = \begin{bmatrix} \frac{xy}{f} & -\left(f + \frac{x^2}{f}\right) & y \\ f + \frac{y^2}{f} & -\frac{xy}{f} & -x \end{bmatrix}$$

The various processing stages of the current system can be thought of as decomposing the right side of equation 2.1 until we are left with a map representing the direction of the vector \mathbf{T} , which is the observer's heading direction.

In the following description, we will be interested in the speed and direction of the flow components at each retinal location since these components form the input to the system. The speed at a retinal position is:

$$v = \|\theta\| = \sqrt{v_x^2 + v_y^2}, \quad (2.3)$$

and the direction is:

$$\phi = \text{atan} \left(\frac{v_y}{v_x} \right).$$

3 System Description

Figure 2 shows an overview of the neural network architecture. The following paragraphs describe the major components of this system.

3.1 Motion Field. The input layer is a population of cells sensitive to image flow on the retina. Several researchers have proposed physiologically plausible models that yield population encoding of optic flow vectors (e.g., Wang, Mathur, & Koch, 1989). A typical approach uses a separate floret of cells for each sampling point on the retina. Each cell of a floret is tuned to a preferred direction, $e_k = [\cos(2\pi k/n), \sin(2\pi k/n)]$, where $k = 1, \dots, n$. Florets of this form constitute the first stage of the system, called the motion detector field. Motion at the i th retinal location with speed v and direction ϕ generates a response S_{ki} according to the following equation:

$$S_{ki} = \left[v \cos \left(\phi - \frac{2\pi k}{n} \right) \right] \quad 1 \leq k \leq n, 1 \leq i \leq m.$$

For robotic applications, it suffices to represent the flow field with its x and y components:

$$S_{xi} = v \cos(\phi) \quad S_{yi} = v \sin(\phi). \quad (3.2)$$

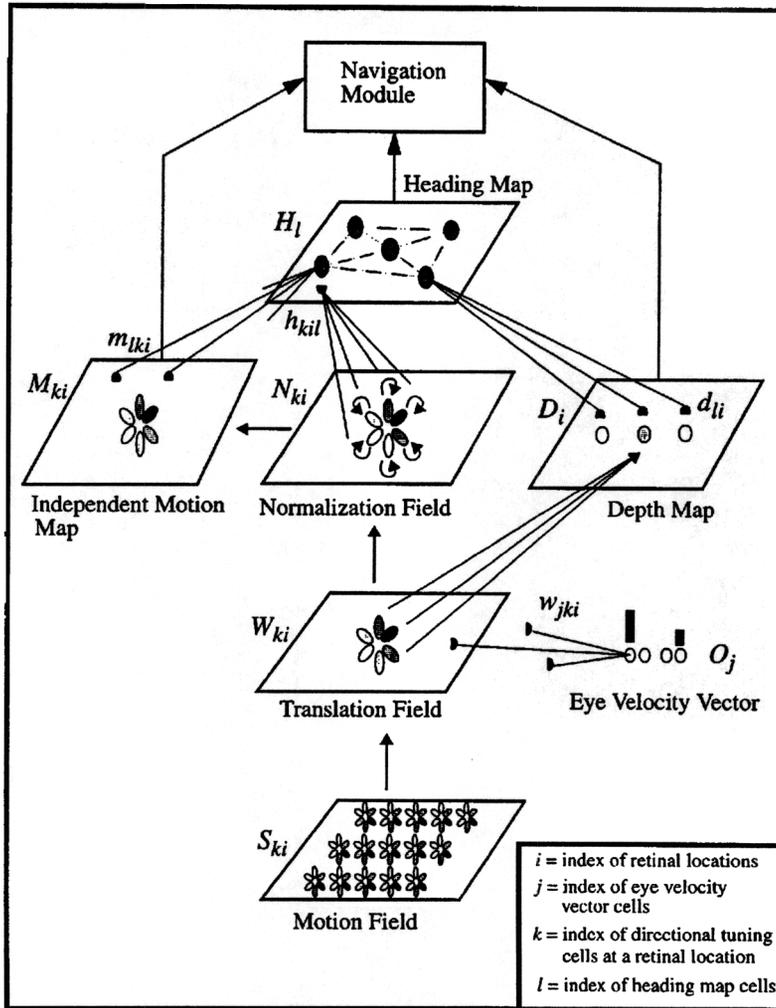


Figure 2: Overview of the system. Cell activities are represented by uppercase letters, and the modifiable synaptic weights projecting to these cells are represented by the same letters in lowercase. Each floret of velocity-tuned cells corresponds to a single retinal location. Simulations reported in this article utilized simplified versions of these florets that contained cells for only the x and y components of the flow fields. See text for details.

The simulations in this report used a motion detector field that consisted of these two directionally tuned cells at each of 49 (7×7) retinal locations unless noted otherwise.

Although the primary goal of the present model is not to achieve detailed biological fidelity, it is of some interest that responses of many directionally tuned cells in the cortical motion processing stream—for example, cells in the MST area of monkeys (Orban, Lagae, Raiguel, Xiao, & Maes, 1995; Tanaka & Saito, 1989)—increase with speed.

3.2 Eye Velocity Vector. The eye velocity vector consists of two pairs of opponent cells that represent pitch velocity (O_1, O_2) and yaw velocity (O_3, O_4) of the eye when it is rotating in its orbit. For positive rotations about the x -axis, O_1 is linearly related to the rotational speed and O_2 is zero. For negative rotations, O_1 is zero and O_2 is linearly related to rotational speed. O_3 and O_4 code rotations about the y -axis in a similar manner. The eye velocity vector activities project through adaptive inhibitory weights to the translation field as described below. This inhibitory input learns to cancel the portion of the flow field corresponding to eye rotations. It is expected that the system could easily be expanded to cancel rotational flow due to other sources, such as head or wheel rotations, as long as the appropriate velocity vectors are available. Only eye rotations are included in the current implementation for the sake of simplicity. The eye velocity vector in the current system is most naturally interpreted as either an efference copy of eye velocity commands or a proprioceptive representation of eye velocity. In a mobile robot with an active vision system, this vector would correspond to motor velocity commands or velocity feedback from eye position sensors or both. In biological models of eye and arm movement control, processing stages exist that can generate an efference copy of commanded movement speed and direction (Bullock & Grossberg, 1988; Grossberg & Kuperstein, 1989).

In most situations, people fixate on a point as they move. This behavior generates a mixed optic flow signal resulting from the simultaneous translation of the body and rotation of the eyes. Unless the point of fixation coincides with the direction of heading, the FOE that results when fixating during egomotion does not correspond to the direction of body translation. The current system uses an estimate of eye rotational velocity to negate internally the rotation part of the signal from the flow field. There are at least two other approaches to solving the heading problem during eye rotations. A less accurate approach is to sample a large part of the flow field and subtract off any component that is common at all depths. This, however, has the advantage of not requiring explicit knowledge of eye rotations (Perrone, 1992). Another approach, exemplified by the Heeger and Jepson (1990) algorithm, uses more sophisticated heading calculations that work in the presence of rotational components, again without requiring explicit knowledge of eye rotations. These more complex algorithms could be used

in place of the heading calculations performed by the current model while maintaining the current depth and independent motion algorithms.

3.3 Translation Field. As shown in Figure 2, each cell O_j in the eye velocity vector has a set of inhibitory weights w_{jki} that project to each directionally tuned cell k in the floret at every position i in the translation field. Each translation field cell also receives excitatory input from the corresponding cell in the motion detector field. After learning (see section 4), the net result of these inputs is a field of cells that represents only the component of the flow field that is due to translational motion. That is, the translation field activity contains only the $p(x, y)A(x, y)T$ component of equation 2.1.

A variant of the vector associative map (VAM) neural architecture (Gaudiano & Grossberg, 1991) is used to adjust the weights projecting from the eye velocity vector in such a way as to cancel out the portion of flow corresponding to voluntary rotations of the eyes as registered at the eye velocity vector. The VAM allows both error-based learning and performance to take place simultaneously. This property has enabled VAMs to control self-adapting mobile robots autonomously (Gaudiano, Zalama, & Lopez-Coronado, 1996; Zalama, Gaudiano, & Lopez-Coronado, 1995). Using such a VAM, the activity of a cell in the translation field is formed by subtracting the inhibitory input from the eye velocity vector cells (corresponding to the rotational component of the flow field) from the excitatory input projecting from the motion detector field,

$$W_{ki} = S_{ki} - \sum_j O_j w_{jki}, \quad (3.3)$$

and the learning equation for the weights projecting from the eye velocity vector is:

$$\frac{dw_{jki}}{dt} = \alpha_1 W_{ki} O_j, \quad (3.4)$$

where α_1 is a learning-rate parameter ranging from 0.9 for noise-free conditions to 0.01 for noisy conditions. Before learning, the weights are set equal to zero.

The eye rotation cancellation map learns best when the incoming flow is purely rotational. If desired, this can be guaranteed by gating learning off if translational motion commands (e.g., limb movement or wheel rotation) are active. During purely rotational movements of the eye, the learning law of equation 3.4 adjusts the weights so that the inhibitory input from the eye velocity vector exactly cancels the excitatory input from the motion detector field. When the two inputs are equal, W_{ki} in equation 3.4 goes to zero, and learning stops. At this point the system is tuned such that the rotational component of the flow field is effectively "filtered out" by eye velocity vector projections at the translation field.

During movements with both a translational and rotational component, only the rotational component is reliably correlated with the presynaptic activity of the eye velocity vector cells. The translational flow component will vary depending on heading direction and will not be strongly correlated with eye velocity vector activity. This translational component thus amounts to a form of noise in the learning process. A slower learning rate can be used to “average out” the noisy translational component, as well as other forms of noise, such as noisy sensor information. The ability to learn correct parameters in the face of sensor noise or translational motion is demonstrated in the simulations of section 4.

3.4 Normalization Field. The next stage of cells in the system performs a normalization operation on the outputs of the translation field. This is done because the magnitude of the flow vectors at the translation field contains information pertinent to observer speed and object distances only, not observer heading direction. In other words, we are not interested in the magnitude of the $p(x, y)\mathbf{A}(x, y)\mathbf{T}$ term from equation 2.1. Only the direction of vector \mathbf{T} is needed. Normalization removes the unpredictable changes in flow vector magnitudes due to varying object distances and translation speeds. The normalization calculation is:

$$N_{ki} = \frac{[W_{ki}]^+}{\sqrt{\sum_m ([W_{mi}]^+)^2}}, \quad (3.5)$$

where $[]^+$ denotes a rectification function such that $[x]^+ = x$ for $x > 0$, $[x]^+ = 0$ otherwise. All N_{ki} are set to zero for retinal locations where the total flow signal is zero. Although not implemented in the simulations, it may be useful to have a threshold for eliminating small but nonzero activities in the normalization map because optic flow detectors operating on a real image will be noisy. These locations typically include points with no visible objects, very distant objects, or points near the FOE for a given heading. If we assume that the detectors are more accurate at detecting the direction of large-flow signals than small-flow signals, then setting the threshold slightly higher would allow the network to base its heading decision on larger, more reliable signals.

3.5 Heading Map. The next stage of cells is a self-organizing feature map (SOFM) that encodes heading. Cells in the heading map receive weighted excitatory input projections from the normalized flow field. Heading can be determined by classifying the pattern across these inputs. Kohonen (1982) described a learning algorithm, based on earlier work of von der Malsburg (1973) and Grossberg (1976), that can be used to self-organize a topographical map whose cells classify input patterns. In our application, neighboring cells in the map code similar heading directions. During learning, neighbor-

hood interactions cause adjacent cells in the heading map to code similar headings. The interactions also ensure that the map uses all of its cells to encode headings. Otherwise, a single cell might learn large weights early on and continue to monopolize learning even though its match with the input pattern is poor. In addition, the neighborhood interactions cause heading map cells to distribute themselves automatically according to the frequency statistics of sampled headings. Greater discrimination among the more common headings is possible because these headings attract a higher density of map cells.

The heading map consists of a group of cells with activities H_l that perform a maximum inner product classification (Grossberg, 1976; Kohonen, 1982) using weights h_{kil} projecting from cells with activities N_{ki} in the normalized flow field. For each presentation of a normalized flow pattern, one cell in the heading map will receive the largest total input. Early in development, this maximally excited cell and its N nearest neighbors are set to activity level $1/(N + 1)$, and all other map cell activities are set to zero. Weights projecting to each of the active cells are adjusted toward the flow pattern. Initially, large neighborhoods help the map cells, which begin with small, random weights chosen from a uniform distribution between 0 and 0.1, to "unfold" properly and thereby cover the entire range of experienced headings. Over time, the neighborhood size N shrinks to zero so that flow patterns are categorized more precisely. The following equations were simulated:

$$H_l^* = \sum_{ki} N_{ki} h_{kil}$$

$$H_{\max} = \text{cell with maximum activity } H_l^*$$

$$H_l = \frac{1}{N + 1} \text{ for } H_{\max} \text{ and } N \text{ neighbors}$$

$$H_l = 0 \text{ otherwise}$$

$$\frac{dh_{kil}}{dt} = \alpha_2 (N_{ki} - h_{kil}) H_l,$$

where H_l^* is the input to the l th heading map cell, and α_2 is a learning rate parameter that starts at 0.1 and shrinks with a linear decay rate to 0.001 over 2000 learning trials. The neighborhood N starts as a 15×15 square centered at the maximally active cell. The width of this square shrinks by one after every 100 heading samples until the neighborhood consists of only the maximally active map cell.

After training, a cell in the heading map will respond maximally to flow generated by a particular heading. However, most headings will fall between those encoded by cells in the heading map. During performance, winner-take-all competition among the heading cells is relaxed so that several candidate heading cells survive the competition based on the size of

their total input. Allowing distributed activation across heading cells creates an interpolated output. For example, if an input heading falls halfway between headings encoded by two cells, then each of those cells will probably survive the competition and contribute approximately half of their information to the rest of the network. Activity in the surviving heading cells is also normalized so that the heading representation is a weighted average of active heading cells. Heading cell activity is thus calculated during performance as follows:

$$\begin{aligned}
 H_i^* &= \sum_{ki} N_{ki} h_{kil} \\
 H_i^{**} &= 0 \quad \text{if } H_i^* < (\max_m [H_m^*] - \Gamma) \\
 H_i^{**} &= H_i^* \quad \text{otherwise} \\
 H_i &= \frac{H_i^{**}}{\sum_m H_m^{**}}, \tag{3.8}
 \end{aligned}$$

where Γ is a threshold parameter determining how many cells survive the competition. For the simulations, Γ was set to $\max_m [H_m^*]/15$. In words, the heading map cell activities H_i are a contrast-enhanced version of their total inputs H_i^* . Other methods, such as the softmax of Bridle (1989), could be used to provide the same functionality as this thresholding process. Note that the sizes of the total inputs to the map cells determine the “neighborhood” during performance, whereas the active neighborhood during training is determined based on proximity to the maximally active cell. This was done because it provided the best map unfolding during training and noise tolerance during performance. Since the trained map is topographically organized, however, the maximally active cells during performance will still typically be neighboring cells in the map.

As described thus far, the heading map can only discriminate between headings, not *identify* heading in terms of absolute azimuth and elevation angles. This is sufficient for the proper formation of the depth and independent motion maps in the system. However, testing the network’s ability to determine heading accurately and using the heading estimate to perform navigation require that retinotopic labels be attached to each cell in the heading map. A simple technique for assigning a retinotopic label to a heading cell with activity H_i is to find the index i of the smallest weight d_{ii} projecting from that cell to the retinotopically organized depth map. The smallest weight will be located near the FOE on the retina, which is equivalent to the retinotopic heading. Alternatively, the following technique could be used to assign heading labels in a mobile robot:

1. Move toward a visual cue, such as a point of light, at eye level and adjust the wheels until the image of the light is stationary on the optic

sensor. When the image is stationary, the navigator is heading directly toward the point of light in body-centered coordinates.

2. As the robot approaches the light, rotate the eye so that the light image falls on many different regions of the optic sensor. Since the navigator is heading toward the light in body-centered coordinates, the projected point of light should still be motionless even though it is projected to a different retinal location. However, the retinotopic heading will have changed. The floor and other objects in the scene will create a flow pattern that excites a different cell in the heading map.
3. For each eye position, map the maximally excited heading cell to the current retinotopic location of the imaged point of light.

In the heading map simulations, heading labels were supplied by systematically sampling heading directions and labeling each cell with the heading that maximally excited it. The heading estimate is calculated using the following equation:

$$H_{MAP} = \sum_l H_l \Phi_l, \quad (3.9)$$

where Φ_l is the preferred heading of the l th heading map cell, and H_l is determined using equation 3.8. This labeling procedure was done only so the accuracy of the self-organizing map could be compared against other reported models. In the navigation simulations, all navigation took place in a retinotopic coordinate system so the heading labels were applied using the smallest weight technique.

Because the system bases its heading estimate on a flow field where only the rotational component due to self-generated eye movements has been removed, it has an advantage over many other heading estimation algorithms during curvilinear motion. This can be seen in the following example. Imagine that the navigator is following the circumference of a circle. Algorithms designed to ignore rotational components of the flow field without using extraretinal information cannot distinguish whether the rotational component of the flow field is due to eye movement or the curvilinear body movement. Therefore, during curvilinear motion, these algorithms will ignore the rotational component of the motion and always report that the navigator is heading straight forward. In this case, however, "straight forward" is defined in a coordinate system that is rotating as the navigator proceeds around the circle. In other words, the heading estimation algorithm will be unable to distinguish if it is indeed heading straight or in a circle. In contrast, the current architecture will recognize that the rotational part of the optic flow due to the curvilinear motion is not associated with an internally generated eye motion. This unremoved rotational flow component will bias the heading estimate in the direction of the curve, so that the navigator will be aware that it is constantly drifting away from "true forward motion."

3.6 Depth Map. In addition to providing heading information, optic flow can be used to determine the distance to the object at each retinal location (if one is present) in a scene. For visual navigation, it suffices to form a distance measure that is scaled by the speed of the observer—for example, an inverse time-to-collision measure $p\|T\|$ for each retinal location, where p is the inverse depth, or proximity, of an object, as in equation 2.1, and $\|T\|$ is the translation speed of the observer. The general problem of determining scene depth from a sequence of images is a difficult one, largely due to the unknown rotational component of the flow field and the unknown heading direction. Solutions have been proposed (e.g., Koenderink & van Doorn, 1987; Longuet-Higgins, 1981; Prazdny, 1980; Waxman & Ullman, 1985; Waxman & Wohn, 1988; see Simpson, 1993, for a review), but these solutions typically require rather complex calculations. In contrast, once the rotational component of the flow field is removed and heading direction is known, as in the current network, calculating the time to collision at each retinal position is relatively straightforward (see also Perrone, 1992). By removing the rotational flow component from equation 2.1, the optic flow generated by translational motion alone reduces to:

$$\begin{aligned} v_{Tx} &= p\|T\|(-fT'_x + xT'_z) \\ v_{Ty} &= p\|T\|(-fT'_y + yT'_z), \end{aligned} \quad (3.10)$$

where v_{Tx} and v_{Ty} are the x and y components of the translational flow field, f is the focal length of the imaging system, (T'_x, T'_y, T'_z) are the components of a unit vector that defines the direction of translation, and (x, y) specifies retinal location. From equation 3.10, one can derive the relationship between the magnitude of the flow vector at a retinal location, v_T , and time to collision:

$$v_T = p\|T\|\sqrt{(-fT'_x + xT'_z)^2 + (-fT'_y + yT'_z)^2}. \quad (3.11)$$

Note that for a specific heading (T'_x, T'_y, T'_z) and motion sampling position (x, y) on the retina, the square root term in equation 3.11 is a constant, which we can rename k_{Txy} . In the current simulations, which involve only two directional cells at each retinal location of the translation field, the function g that determines the magnitude of the translational flow velocity at the i th retinal location is:

$$g(W_i) = \sqrt{W_{1i}^2 + W_{2i}^2} = v_T. \quad (3.12)$$

In order to produce a depth measure that is invariant across all headings and retinal locations, k_{Txy} must be removed from the flow speed measure v_T . This is accomplished in the system by logarithmically compressing the flow speed represented at the translation field before passing it to the cell representing the corresponding retinal location in the depth map, then subtracting off an average value of this compressed flow speed. Specifically,

the retinotopically organized depth map cell activities are governed by the following equation:

$$D_i = \log[g(W_i)] - \sum_l H_l d_{li},$$

where d_{li} is the weight projecting from the l th heading map cell to the i th depth map cell. Tal and Schwartz (1994) have demonstrated that a logarithmic relationship between neuron firing rate and input activity is a property of integrate-and-fire neurons. This suggests that the logarithmic processing in equation 3.13 could also be implemented in a biological system.

Using a VAM learning mechanism, the weight d_{li} learns to represent the average of the compressed flow velocity signal at the i th retinal location when heading in the direction coded by the l th heading map cell. The learning equation governing the weights d_{li} is:

$$\frac{dd_{li}}{dt} = \alpha_3 D_i H_l,$$

where α_3 is a learning-rate parameter that was set equal to α_2 for the simulations.

Training the network according to equations 3.13 and 3.14 during random movements leads to depth map cell activities that each codes the following depth measure for objects at the corresponding retinal location:

$$\begin{aligned} D_i &= \log v_T - |\log v_T| \\ &= \log(p\|T\|k_{Txy}) - |\log(p\|T\|k_{Txy})| \\ &= (\log p\|T\| + \log k_{Txy}) - |(\log p\|T\|) + \log k_{Txy}| \\ &= (\log p\|T\| + \log k_{Txy}) = (|\log p\|T\|| + \log k_{Txy}) \\ &= \log p\|T\| - |\log p\|T\||, \end{aligned}$$

where the term $|\log p\|T\||$ is a constant determined by the environmental experience of the network during learning. For correct operation, the system should be trained in an environment that consists of objects in the same depth range as the environment it will encounter during performance so that an appropriate value of this constant is learned. A large, positive cell activity in the depth map corresponds to a short time to collision; these large cell activities identify the retinal locations of nearby obstacles for use by the navigation module. This depth measure has several advantages. First, it is easily learned and calculated in the network using the VAM mechanism already described. Second, it is invariant with respect to retinal position. In particular, the same value of the depth measure corresponds to the same time to collision regardless of the retinal location, even though objects at the same depth generate different velocity signals at different places on

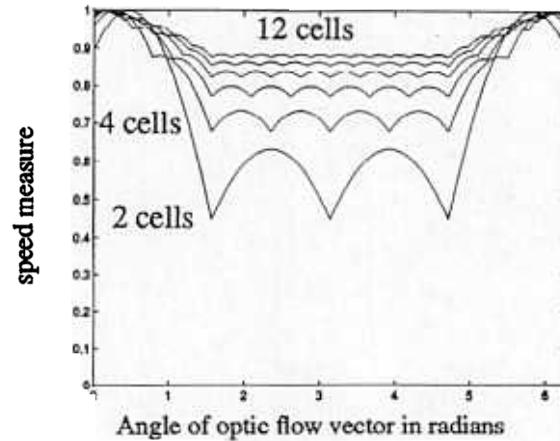


Figure 3: Speed measure produced by equation 3.17 as a function of translational flow angle and number of cells in the floret. The speed measure is scaled so that the maximum output for each collection of cells is 1.0. More cells reduce the angular dependence of this flow speed measure.

the retina. Third, logarithmic processing leads to a depth measure that is compressed such that nearby objects garner a larger percentage of the cell's dynamic range. This allows the depth map to represent proximity of nearby objects more accurately than distant objects, which is a useful property for avoiding collisions. Finally, because the depth map encodes an inverse time-to-collision measure $p\|T\|$, a large signal means a short time to collision. Therefore, the most salient signals in the depth map are those that present the most danger.

A larger number of directional cells at each retinal location (as would be expected in motion processing pathways *in vivo*) enables a more biologically plausible function g than the one defined by equation 3.13 to determine the magnitude of translational flow velocity:

$$g(W_i) = \sum_k [W_{ki}]^+ \quad (3.16)$$

When there are a small number of directionally tuned cells at each retinal location, this measure depends on the direction of the flow vector at each translational flow speed. Larger numbers of directionally tuned cells eliminate this dependence (see Figure 3).

3.7 Independent Motion Map. When one is sitting still, it is a common experience to have one's attention drawn to a moving object. This is not

surprising since the isolated motion signal uniquely identifies the location of the object. During locomotion, however, large optic flow signals fill the entire retina, yet independently moving objects may still retain their perceptual saliency. For example, the optic flow velocities in peripheral vision can be quite large when one is driving down a street, yet the location of a moving vehicle is still salient because the optic flow signal it creates differs from the pattern of flow generated by surrounding stationary objects.

The independent motion map in Figure 2 is formed by a VAM mechanism, similar to the one used for estimating depth, that detects independently moving objects. Whereas the depth VAM compares the magnitude of a flow signal to a learned average, the motion VAM compares the direction of flow with an expected pattern of flow. The net input to a cell in the independent motion map is:

$$I_{ki} = N_{ki} - \sum_l H_l m_{lki},$$

where N_{ki} is the excitation from the corresponding cell in the normalization field and the m_{lki} are adaptive weights tuned to represent the expected flow field when heading in the direction represented by heading map cell output H_l . The cell outputs M_{ki} in the independent motion map are calculated from this input as follows:

$$\begin{aligned} M_{ki} &= I_{ki} \quad \text{for} \quad \sum_k I_{ki} > \Lambda \\ M_{ki} &= 0 \quad \text{for} \quad \sum_k I_{ki} \leq \Lambda, \end{aligned} \quad (3.18)$$

where

$$\Lambda = \max \left(\sum_k I_{ki} \right) - 0.1$$

This is a dynamic threshold based on the maximum disturbance in the independent motion map that has the effect of maintaining only the activities of pixel locations with approximately the same level of activity. Activity in the independent motion map is the vector difference between the current normalized flow vector and the expected normalized flow vector. The length of this vector is reflected in the total activity of the florets at a given pixel location of the independent motion map. A flow vector that differs by 180 degrees will have maximal activity, while a flow vector that differs only slightly will have very little activation. If there is any noise in the system, it will unavoidably appear as activity in the map. The threshold, Λ , is introduced as a simple way to "clean up" the independent motion

map. Its chosen value can be flexible depending on the needs of the navigator. We expect that, as designed, the independent motion map would be most useful with a fairly high threshold. In this way, the robot's attention would be drawn only to pixel locations with significant deviation from the expected flow direction. Of course, this means that the robot might miss slowly moving objects or objects that match its heading. However, without some threshold there will be continuous low-level activity across the entire map that might be distracting.

The weights m_{ik} are adjusted as follows:

$$\frac{dm_{ik}}{dt} = \alpha_4 I_{ki} H_i, \quad (3.20)$$

where α_4 is a learning-rate parameter that was set equal to α_2 for the simulations. This learning law leads to weights m_{ik} that represent the average directional flow pattern seen at retinal location i when moving in the heading direction coded by H_i . After training, if the pattern of flow received from the normalization field corresponds to the expected pattern for heading H_i , then weighted inhibition from H_i will cancel the input from the normalization field, and all activity in the independent motion map will be quenched. However, independently moving objects will likely generate flow directions that differ from the directions expected for the perceived heading and retinal locations. A nonzero output M_{ik} indicates that an independently moving object is located at the retinal location indexed by i since the flow in direction k at this retinal location is significantly larger than the expected flow for the current heading as encoded by the weights m_{ik} . The larger the value of Δ , the larger the angular difference between the normalized flow signal and the expected flow signal must be before it produces a positive activation in the independent motion map.

It is possible for an object to avoid detection by the independent motion map if that object is moving toward the observer such that the flow caused by its motion is exactly aligned with the translational flow field. Such an object would still be avoided by the current system since the faster flow would be registered as a shorter time to collision in the depth map, allowing the robot to navigate around the object before it gets too close. Also, noise in the flow field can lead to false positives at the independent motion map. However, these false positives will typically last for only a few time steps and can be largely eliminated by averaging over time.

Although learning in the weights projecting to the heading, depth, and independent motion maps occurs concurrently, correct values of the independent motion map and depth map weights depend on the heading directions coded by cells in the heading map. These weights thus do not stabilize until shortly after the weights projecting to the heading map have stabilized.

3.8 Navigation Module. Together, the heading, depth, and independent motion maps provide a representation of the visual world that can be used to navigate around obstacles toward stationary or moving targets. Goal-based navigation in the presence of obstacles is carried out in the simulations reported here using a very simple approach-avoidance algorithm. We do not pose this algorithm as a new or creative solution for navigation; rather, we use it simply to show that the maps built up by the system are robust enough to allow successful navigation in the presence of large amounts of sensor noise even with a relatively “stupid” navigation algorithm.

For simplicity, the navigator was limited to fixed-speed motion on the ground plane ($\|T\| = 1$), thus requiring the generation of only an azimuthal translation command, T_x . Navigation toward a target involves keeping the output of the heading map H_{MAP} , defined in equation 3.9, equal to the direction of the goal G in retinal coordinates. If the target and heading do not match, then a nonzero difference vector generates motor commands that adjust the navigator’s translation T_x toward the target,

$$T_x(t + 1) = T_x(t) + \varepsilon(G - H_{MAP}),$$

where ε is a gain factor set to 0.02. Translational commands were limited to $-45^\circ < T_x < 45^\circ$.

Obstacles are ignored until one becomes a threat by registering a short time to collision as indicated by cell activity in the depth map, defined by equation 3.13, greater than a threshold value β . A value of $\beta = 3.5$ was used in the simulations reported here. The center of mass C_x of these suprathreshold cells in the depth map is calculated, and the heading is altered to move away from this center according to the following algorithm:

$$\begin{aligned} \text{if } (H_{MAP} > C_x) \text{ then } T_x(t + 1) &= T_x(t) + \delta \\ \text{if } (H_{MAP} < C_x) \text{ then } T_x(t + 1) &= T_x(t) - \delta, \end{aligned}$$

where δ is a small, positive constant set to 0.2 in these simulations. The navigator veers away from the looming obstacle until it is no longer a threat. Once clear, the approach signal regains dominance, and the navigator resumes progress toward the goal.

4 Simulations

4.1 Training the System. An important advantage of this system over previous ones for heading perception in mobile robots is its ability to self-organize through an action-perception cycle rather than requiring teaching signals and supervised learning. The system is trained by randomly generating rotational and/or translational movements, then using the combination of eye velocity information and the flow pattern resulting from the movements to tune the parameters in the translation field and the heading

map. These learning processes are detailed in the following paragraphs. Throughout training and testing, optic flow information was corrupted by varying amounts of three types of noise:

1. **Directional noise.** Each flow vector is perturbed by a uniform randomly distributed angle between plus or minus the amount of directional noise.
2. **Speed noise.** Each flow vector is multiplied by a uniform randomly distributed number between 0 and 2.
3. **Aperture noise.** Each flow vector is perturbed by a uniform randomly distributed angle between plus or minus the amount of aperture noise; then the magnitude of the flow signal is reduced by the cosine of the angular difference between the original and perturbed vectors. Aperture noise attempts to model the uncertainty of using local flow detectors to measure the motion of a luminance edge. Our use of noise that is uncorrelated across pixel locations is only a rough approximation to the aperture noise that would arise in a real vision system, which could lead to higher error levels than those seen in the simplified simulations reported here.

During training and performance, simulated objects in the field of view were placed randomly at distances between 1 and 200 units from the navigator. The units are relative to a navigator focal length of 1 and a navigator speed of 1.

4.2 Translation Field. The weights projecting to the translation field from the eye velocity cells learn to cancel the rotational optic flow generated by eye movements. The easiest way for this to happen is to generate random eye motions without any translational movement (imagine an infant scanning around a room before it can crawl). Therefore, learning in the translation field was carried out before learning began in the higher stages of the model. Under these circumstances, a fast learning rate may be used, and the system is completely trained after only 20 to 30 random eye movements. Figure 4A shows the weights projecting from each eye velocity vector cell to the floret of cells at one retinal location after training with purely rotational movements of the eyes. To illustrate more clearly the pattern of weights projecting from each eye velocity vector cell, 12 directional cells were used in each floret of the translation field in this simulation.

If necessary, the weights may also be trained in the presence of translational movements. Random translations have an effect similar to noise on the desired training signal. In an actual implementation, noise may also result from limitations of sensor arrays such as those due to the aperture problem (Wallach, 1976; Perrone, 1992), although this problem may be reduced by motion preprocessing stages that convert aperture ambiguities

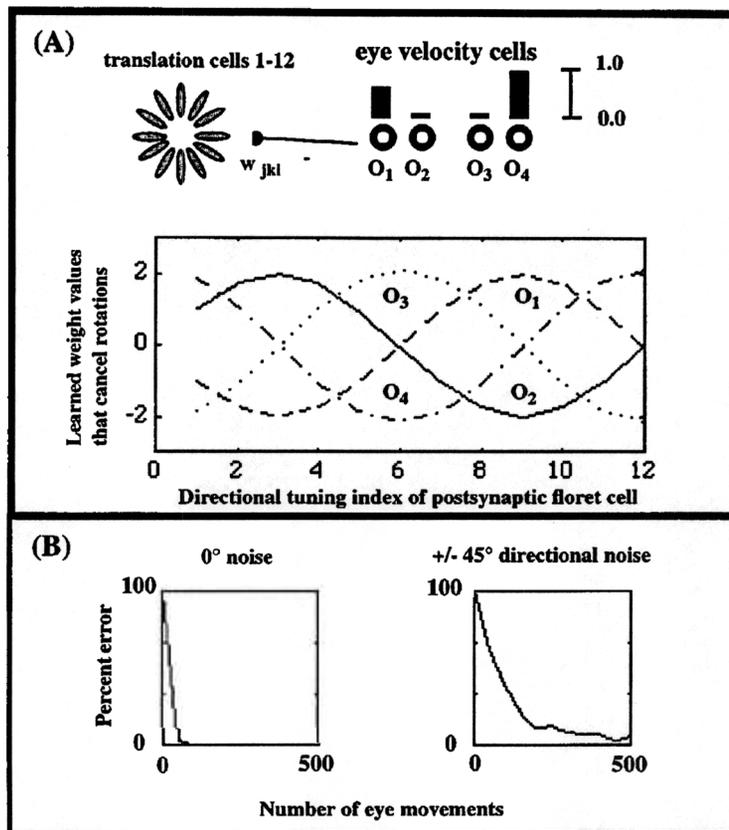


Figure 4: Learning at the translation field. (A) Inhibitory weights w_{jki} in equation 3.4 projecting from eye velocity signals after training with purely rotational eye movements. Each of the four curves on the plot indicates the weight values projecting from one of the four eye velocity vector cells to all of the floret cells at one retinal location. An eye rotation corresponding to the activation of a single eye velocity vector cell results in a flow pattern that takes a cosine shape across each floret of cells (see equation 3.1). The cosine shapes of these curves thus indicate that the weights have successfully learned to cancel the flow field components due to eye rotations as reflected at the eye velocity vector. (B) Noise tolerance while learning eye velocity parameters without noise (left) and with ± 45 degrees of random directional noise (uniformly distributed) added to the motion detector field input. Percentage error is measured as the amount of residual activity at the translation field during a rotational movement divided by the amount of activity that would occur without rotational nulling.

into coherent representations of object speed and direction (Chey, Grossberg, & Mingolla, 1996).

A slower learning rate is required for stable learning with noise. Still, the system can learn to cancel the effects of eye rotations on the flow field in the presence of noise with a relatively small number of movements, requiring on the order of 500 randomly generated eye movements to tune the parameters. Such a simulation is summarized in Figure 4B, which shows how the error decreases as training proceeds in the noiseless case and with ± 45 degrees of random (uniformly distributed) directional noise added to each cell's input at the motion detector field.

4.3 Heading Map. Like learning at the translation field, learning at the heading map is carried out during an action-perception cycle. Here, however, the goal is to train the system to use the translational component of the flow field to determine heading direction. This is done by randomly generating translational movements of the eye (as if it were mounted on a moving body), then using the resulting translational flow field to self-organize a map representation of heading direction.

As noted in section 3, the heading map is a variant of a SOFM. During learning, the cells in the heading map spread out to code different heading directions. The angular separation of neighboring heading cells will depend on the number of cells in the map and the statistical distribution of heading directions sampled during learning (Kohonen, 1982). This provides the map with the desirable feature of efficiently allocating its resources toward more commonly experienced input patterns. For example, when the system is trained on a body that spends most of its time moving forward (as would be expected for most animals), it develops an accuracy bias as illustrated in Figure 5A. The heading map on the left is the result of training on a set of 2000 movements with headings distributed uniformly between ± 45 degrees in both azimuth and elevation. The map on the right was trained on a set of headings biased toward small deviations from straight ahead. The tightly grouped heading cells in the center of the map (corresponding to straight-ahead movement) allows the map to code these directions more accurately than more peripheral directions. In this regard, Crowell and Banks (1993) noted that people are more accurate at judging headings with small eccentricities (forward) than those with large eccentricities (sideways), and similar degradation of heading detection as the FOE moves away from the fovea was noted by Warren and Kurtz (1992). When trained with a distribution of headings as shown in the right side of Figure 5A, the system develops a similar accuracy bias.

As noted for learning in the translation field, it is important for the heading map to be tolerant of noise in the direction and speed of local optic flow signals. Figure 5B shows that the heading map is still able to organize topographically when trained with noisy optic flow information. In these "aperture noise" simulations, the effects of the aperture problem were

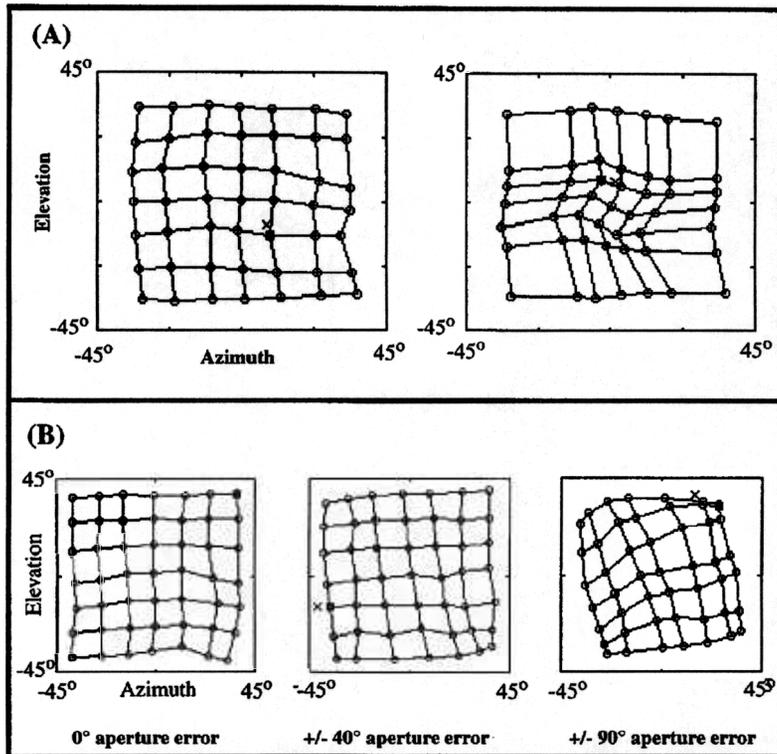


Figure 5: Unfolding of the heading map. (A) Example heading maps with uniform heading sampling (left) and sampling biased toward straight ahead (right). When the distribution of training samples is biased toward straight ahead, the distribution of map cells is more concentrated for movements near straight ahead, resulting in more accurate heading estimates for these movement directions. (B) Heading map after 2000 training steps for three levels of simulated aperture error in the motion detection field.

simulated by randomly perturbing each perceived flow vector by an angle uniformly distributed between ± 0 , 40, and 90 degrees and then reducing the magnitude of each flow vector by a factor equal to the cosine of the perturbed angle minus the actual angle. Two thousand randomly generated movements with headings between ± 45 degrees for azimuth and elevation were used to train the network. In all noise simulations, the same level of noise was used for both training and testing of the network.

Figure 6A shows heading estimate accuracy under various kinds and amounts of noise in the optic flow signal. To allow comparisons with the

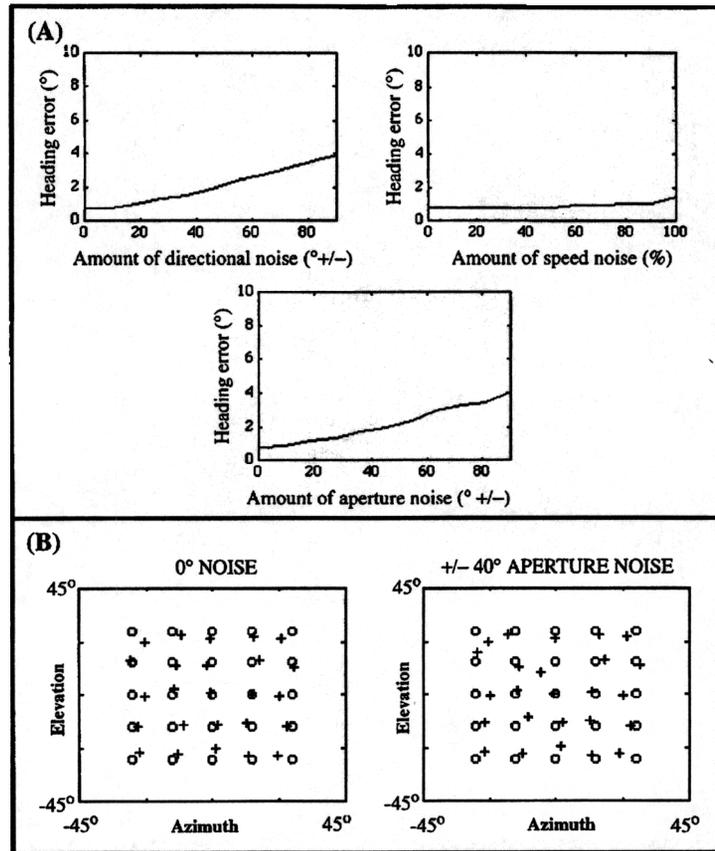


Figure 6: Heading estimation under various noise conditions. (A) Error in heading judgment plotted as a function of the amount of directional noise, speed noise, and aperture noise in the flow field for a network trained with headings between ± 25 degrees. (B) Example of heading error for 0 degree noise and 40 degree aperture noise for a network trained with headings between ± 45 degrees (top). The circles mark the actual heading direction, and the plus signs mark the estimated heading direction.

simulations of Hatsopoulos and Warren (1991), training was carried out on headings between ± 25 degrees in azimuth and elevation, and error was averaged over test headings between ± 20 degrees. The directional noise simulations indicate that the network performs with about 0.75 degree average error with no directional noise and 3.9 degrees average error with ± 90 degrees directional noise. This is comparable to both human performance and

the performance reported in Hatsopoulos and Warren (1991), which showed average errors of approximately 0.8 degree for no noise and 3.4 degrees for ± 90 degrees noise. The network's performance under aperture noise (see bottom of Figure 6A) is similar to the directional noise results. Finally, a simulation using "speed noise" (an increase or decrease of the magnitude of the flow vector), reported in Figure 6A, shows that speed noise has little effect on the network's performance, again in keeping with psychophysical results showing that speed noise has little effect on the ability of humans to determine heading.

Some comments should be made regarding these comparisons. First, the current network achieves comparable results despite using a learning scheme that trains from randomly generated headings as compared to the learning scheme of Hatsopoulos and Warren that requires network-using optic flows that correspond to known headings. Second, the current network works in the presence of eye rotations due to the learned removal of rotational flow components at the translation field; the Hatsopoulos and Warren does not work in the presence of eye rotations. Although the results reported here use a slightly larger retina (7×7 versus 5×5) and more cells for heading detection (49 versus 25), the number of input patterns required to train the network adequately is less in the current system than in the Hatsopoulos and Warren (1991) network (2000 versus 4000).

Figure 6B shows a graphical representation of heading estimation performance under two different amounts of aperture noise, this time for a network trained with headings between ± 45 degrees. The open circles denote an arbitrarily selected grid of sample headings. The plus marks plot the responses of the heading map.

A possible criticism of these simulations is that the aperture problem is not simply uniform random noise but instead is systematic in nature for objects that span many retinal locations and thus may cause errors that do not simply average out as they appear to in the previous simulations. The results of a simulation designed to test further the system's ability to deal with the aperture problem are illustrated in Figure 7. Several changes distinguish this simulation from those presented in Figure 6. First, in previous simulations, the scenes consisted of point objects placed at random depths for each pixel location. A flow signal was calculated for each pixel location, and then various types of random noise were added to the flow field. Here, larger rectangular objects replace the point objects. A more realistic aperture effect is simulated by assuming that the flow detectors can detect activity only at luminance boundaries and, due to the aperture effect, can detect only the component of flow that is perpendicular to the luminance edge. At corners, the ambiguity of the aperture effect is removed because both the horizontal and vertical flow detectors can respond. Second, because most of the field now contains no information, the network was modified so that no learning takes place at pixels without activity in the translation field. Third, the retina was increased to 20×20 pixels so that objects with long,

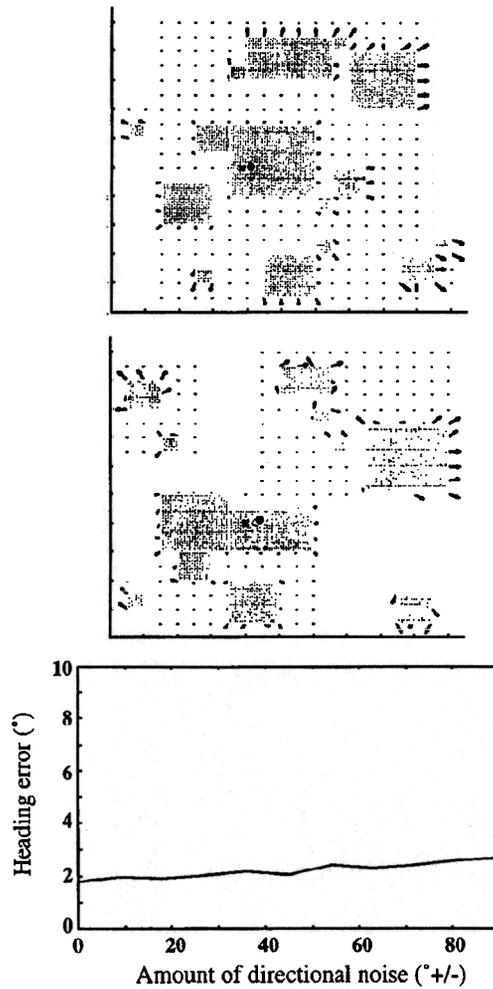


Figure 7: Simulation providing a more realistic treatment of the aperture problem. (Top) Example scene where aperture noise is correlated to the viewed objects. The objects are squares of various dimensions placed at random depths averaging 100 units away. It is assumed that the optic flow detectors can detect only the component of the flow field that is perpendicular to the luminance edge. The "x" marks the actual heading and the open circle marks the heading predicted by the heading map. (Center) Scene with correlated aperture noise plus ± 90 degrees directional noise. The "x" marks the actual heading direction and the open circle marks the heading estimated by the heading map. (Bottom) Plot of average heading map error versus amount of directional noise added in addition to the systematic aperture noise error induced by shapes in the scene.

straight edges could be observed. Fourth, the learning rate and neighborhood shrinking rate were reduced by one-half. Finally, the number of trials was increased from 2000 to 8000. This last change reflects the fact that for any given trial, three-fourths of the pixels are not receiving information and therefore cannot learn. With these new scenes, it simply takes longer for each pixel to experience enough training data.

The top panel of Figure 7 is an example of the more realistic scenes and aperture effect used to train and test the network. The center panel shows a typical training scene with the aperture effect and an additional ± 90 degree directional noise. The bottom panel shows a plot of the average error in heading estimation as an increasing amount of directional noise is added in addition to the systematic aperture effect error induced by shapes in the scene. For zero added directional noise, the average heading error is about ± 1.8 degrees, indicating that the network deals very well with the more realistic aperture effect. The average heading estimation error with an additional ± 90 degrees of aperture noise increases to only ± 2.7 degrees.

4.4 Depth Map. Figure 8 shows the effect of directional flow noise on relative depth estimation. The optic flow field used in these simulations had 49 cells arranged in a 7×7 grid. A random depth was selected for objects at each of the 49 retinal locations. Then movements were made toward those objects along 36 randomly chosen headings between ± 37 degrees in both azimuth and elevation. Because the weights learned for the depth map depend on the nature of the flow field experienced during training, it is important that the flow field used to train the map is similar to the flow fields that will be encountered after learning ceases. Specifically, the training flow field should contain flow information corresponding to the typical range of object depths that will be encountered during performance.

The first three panels of Figure 8 (the top two and the bottom left) compare the depth measure calculated by the system to the actual depth of the object at each retinal location. (For purposes of comparison, actual depth was processed according to equation 3.15 and scaled to compensate for translational speed.) For each of these simulations, noise was present during both training and performance. The effects of noise during training are quite small relative to the effects of noise during performance; that is, most of the error in the plots is attributable to noise during performance rather than incorrect values of system parameters learned during training. The final plot in Figure 8 (bottom right) shows error versus the amount of directional noise. The system was trained in the absence of noise for this plot. Error was calculated as the difference between the actual object depth and the depth predicted by the network divided by the actual object depth. This was multiplied by 100 to obtain the percentage error, and the result was averaged over all retinal locations and 36 different headings. As indicated in Figure 8, the depth map is unusually robust to noise because the optic flow signals were perturbed in direction but not magnitude. Recall that for

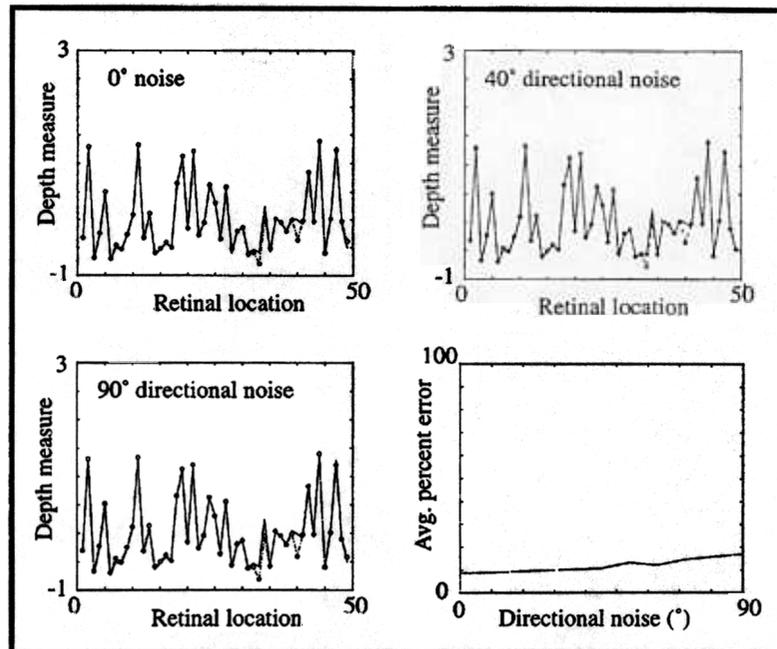


Figure 8: Effect of different levels of directional noise on relative depth estimates. Each of the 49 cells on the 7×7 retina estimates the relative depth of the object in its receptive field. The first three plots (the top two and the lower left) show network depth estimates (solid lines) compared to actual relative depths (dotted lines) for three different levels of directional noise presented during both training and performance. The final plot shows the average percentage error of depth estimates as a function of directional noise for a network trained in the absence of noise.

a given heading, object depth depends on only the magnitude of the optic flow signal. Therefore, as long as the heading estimate is reasonably correct (see Figures 6 and 7), the depth estimates will also be correct regardless of directional noise.

If any speed noise is present in the flow field, depth estimation will unavoidably degrade. This is evident from Figure 9, which shows the effect of aperture noise on relative depth estimation. Since aperture noise contains a speed component in addition to the directional component, depth map accuracy degrades significantly more than with directional noise. Nonetheless, the depth map still performs well with 40 degree aperture noise, and later simulations will show that the 100% average error in depth estimates under 90 degree aperture noise is still adequate for navigation because multiple

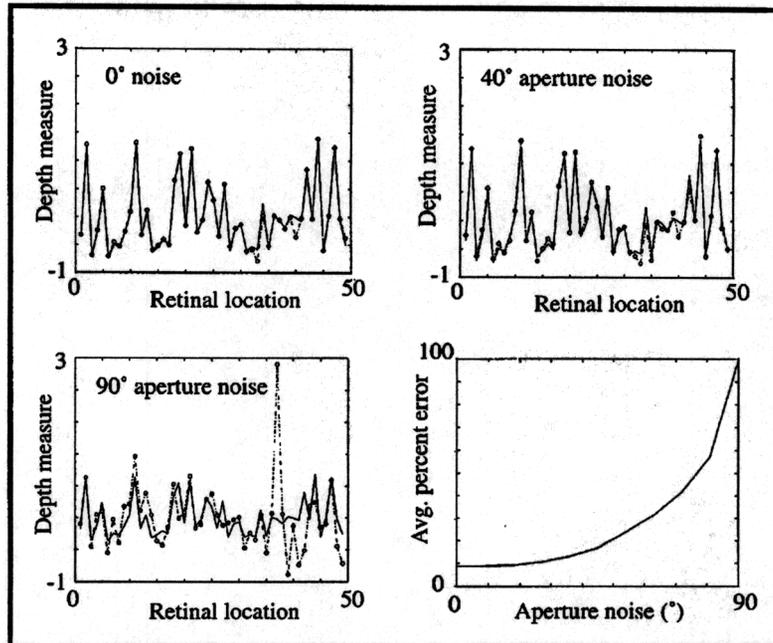


Figure 9: Effect of different levels of aperture noise on relative depth estimates. Each of the 49 cells on the 7×7 retina estimates the relative depth of the object in its receptive field. The first three plots (the top two and the lower left) show network depth estimates (solid lines) compared to actual relative depths (dotted lines) for three different levels of aperture noise presented during both training and performance. The final plot shows the average percentage error of depth estimates as a function of aperture noise for a network trained in the absence of noise.

snapshots of the scene, taken as the navigator moves, tend to average out the unwanted effects of noise over time. Performance could be improved further if predictions about the continuity of surfaces in the field of view were used to perform neighborhood smoothing operations on the output of the depth map, but this topic is beyond the scope of this article. It should also be noted that the worst depth estimation errors occur for objects near the FOE, where optic flow signals tend to be very small. This is another inherent problem with using optic flow for depth estimation. One technique that can help alleviate this problem for real-world scenes is to ignore retinal locations with very small depth weights and fill in depth estimates by averaging estimates from neighboring cells.

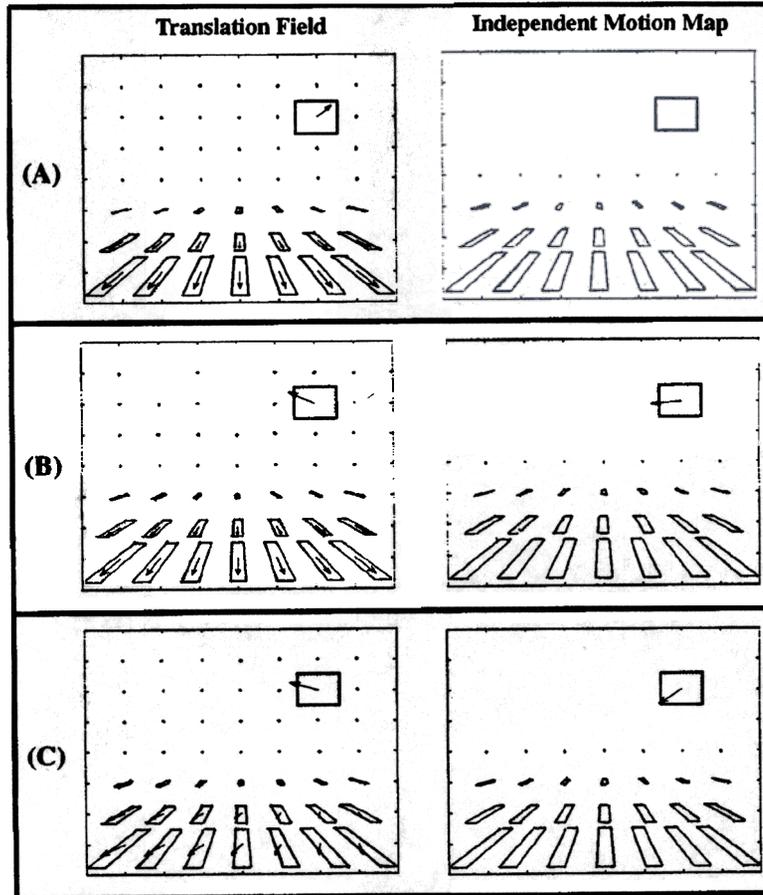


Figure 10: Independent motion detection with no noise. Each panel shows the visual field as seen by the network's retinal array. Arrows in the left panels indicate flow components in the translation field, and arrows in the right panels indicate flow components at the independent motion map after thresholding. (A) The navigator is moving straight ahead over a tiled ground plane. The suspended box is stationary with respect to the ground plane, so no activity remains in the independent motion map. (B) The navigator is moving straight ahead over a tiled ground plane. The suspended box is moving to the left. The flow field at the retinal location that corresponds to the box does not match the expected field, so activity at that location pops out in the independent motion field. (C) Here the navigator is moving at 37 degrees to the right while the box still moves to the left, indicating that independent motion may be detected for a wide range of navigator headings.

4.5 Independent Motion Map. Figures 10 and 11 illustrate the performance of the independent motion map. The scene consists of a tiled ground plane, a distant wall, and a box floating above the ground plane between the observer and the wall. Each panel shows the visual field as seen by the network's retinal array. Objects in the visual field are indicated by dotted lines. Arrows in the left panels indicate flow components in the translation field, and arrows in the right panels indicate flow components at the independent motion map after thresholding according to equations 3.18 and 3.19.

Figure 10 indicates performance at the independent motion map in the absence of noise. In Figure 10A, the navigator moves forward, while the box remains stationary. Since optic flow in the scene is commensurate with the flow expected by the activated heading map cells, all activity in the independent motion map is quenched. In Figure 10B, the navigator again moves forward, but this time the box is moving independently to the left. The flow generated by the box does not fit with the flow pattern established by the rest of the scene, so it is singled out in the independent motion map. Figure 10C shows an example of detecting the same box motion while the navigator moves at 37 degrees to the right instead of straight ahead. One might note that the direction coded by the active cells in the independent motion map corresponds only roughly to the actual motion direction of the object with respect to the navigator. This highlights the fact that this map is primarily useful for identifying the retinal locations of moving objects rather than their direction of movement relative to the navigator. The directional error arises because the motion network detects mismatches between expected direction and the incoming flow direction. The difference of these two directions will point approximately in the actual direction of the moving object. However, the exact calculation of direction requires knowledge of both the depth of the moving object and the navigator's speed so that the component of optic flow due to object motion may be separated from the component due to self-motion.

Figure 11 illustrates independent motion map performance under various noise conditions. Figures 11A and 11B illustrate the performance of the independent motion detector in the presence of ± 40 degree and ± 90 degree aperture noise, respectively. Noise was present during both training and performance. The network successfully detects the moving box in both conditions, but the network erroneously detects motion at several retinal locations in the ± 90 degree condition. The independent motion detector is the most sensitive part of the network to directional noise since it relies on the accuracy of optic flow directions at each retinal location and cannot average over the entire retina. Directional noise can be countered to some degree by choosing a higher value of the threshold Λ in equation 3.18 but raising the threshold also increases the chances of missing a moving object. Figure 11C shows the performance of the independent motion map in the presence of 100% speed noise. Because both the heading and independent

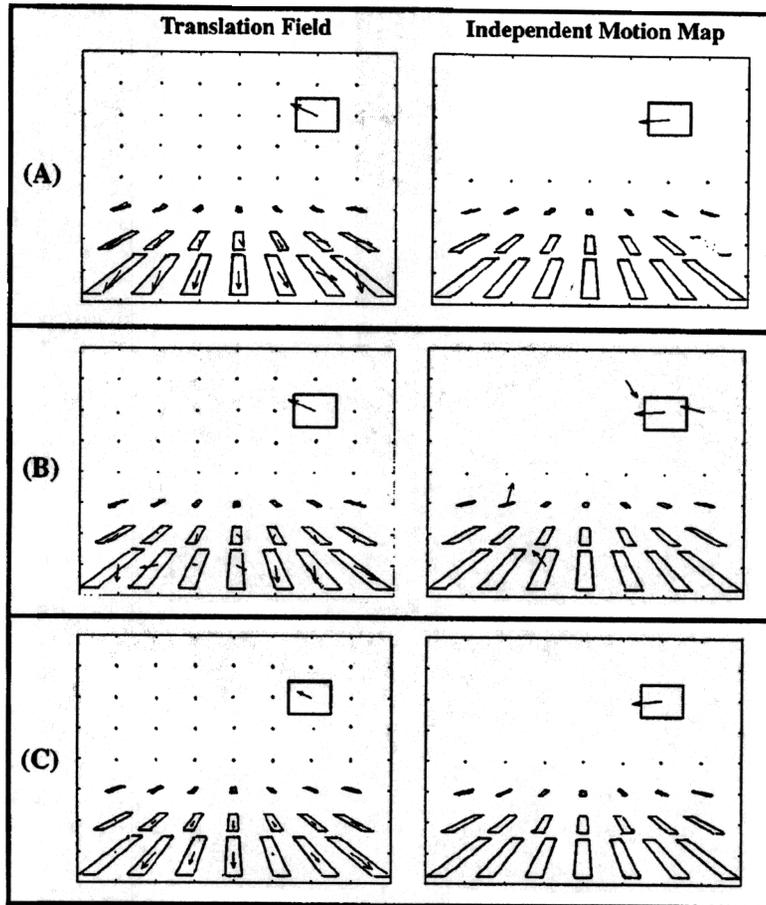


Figure 11: Independent motion detection under various noise conditions. (A) Example of independent motion map performance when the incoming flow field is randomly perturbed by ± 40 degree aperture noise. (B) At ± 90 degree aperture noise, some errant vectors survive the threshold along with the moving box. (C) Performance of the independent motion map in the presence of 100% speed noise. Independent motion detection is essentially unaffected by perturbations in the magnitude of the flow signals.

motion maps are essentially unaffected by perturbations in the magnitude of optic flow signals, the moving box is easily detected in the speed noise condition. Faster-moving objects perturb the optic flow signals more than slower-moving objects and are therefore easier to detect.

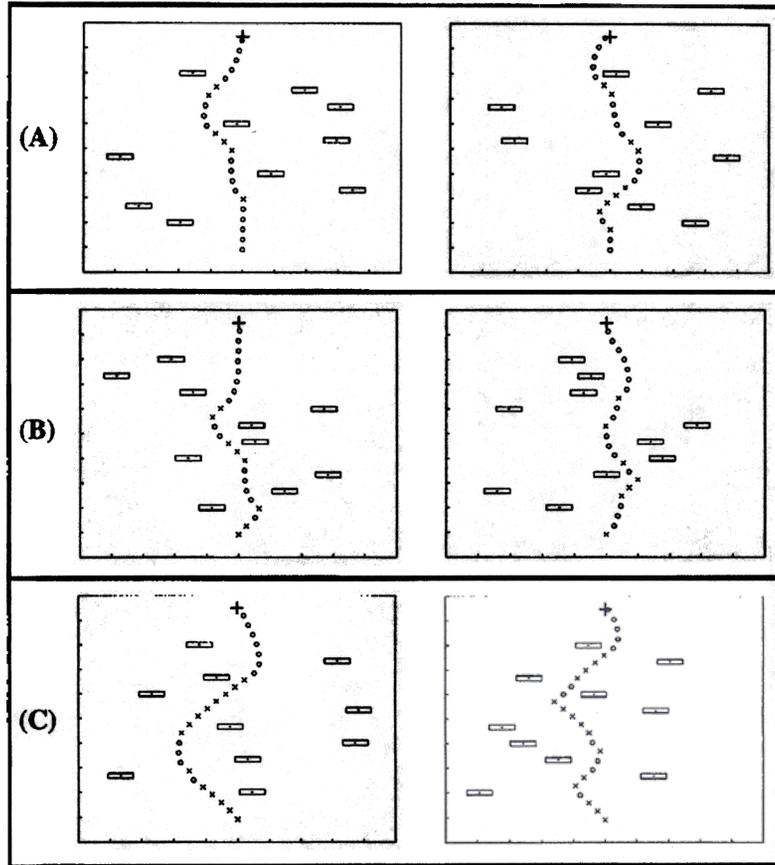


Figure 12: (A) Two example paths taken by the approach-avoidance algorithm, viewed from above. The circles represent times when the approach signal was dominant. The "x" symbols represent times when the avoidance signal was dominant. The navigator starts at the bottom and attempts to reach the plus sign while avoiding the blocks. (B) Two example paths taken by the approach-avoidance algorithm in the presence of 90 degree aperture noise. (C) Two example paths taken by the approach-avoidance algorithm in the presence of 100% speed noise.

4.6 Navigation. The simulations shown in Figure 12 demonstrate the utility of the self-organizing heading and depth maps for visual navigation. Figure 12A shows an overhead view of a field of rectangular obstacles and the path that the navigator takes to reach the goal indicated by the plus

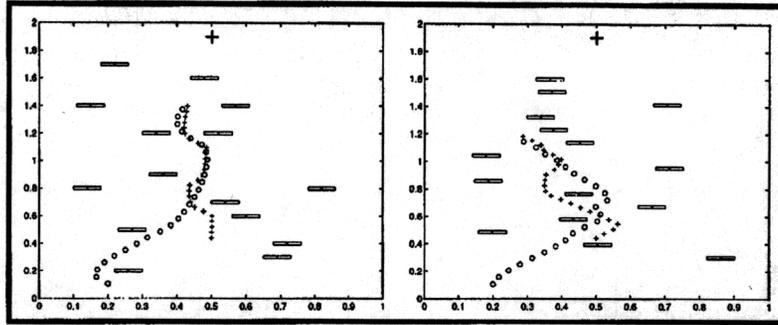


Figure 13: Pursuit behavior viewed from above. The first navigator, designated by the open circles, is pursuing the navigator designated by the plus symbols. The "plus" navigator is slightly slower but gets a head start in its attempt to reach the goal. The other navigator is faster and attempts to capture the first navigator before it reaches its goal. The first navigator (open circles) detects the location of the other navigator using its independent motion map. In both simulations, the open circle navigator is able to overtake the slower plus navigator before it reaches its goal.

sign. The task for the navigator is to start at the bottom center and reach the plus sign at the top of the scene without hitting any obstacles. No high-level path planning was used, only the simple approach-avoidance algorithm described in section 3. The open circles mark places along the path where the navigator was in approach mode, and the "x" symbols mark the places where the navigator was in avoidance mode.

Because the components of the network are robust to noisy optic flow fields, the approach-avoidance calculation based on the network's output is also robust to noise. Figure 12B shows two example paths taken by the navigator in the presence of 90 degree aperture noise. The main difference between performance in 0 degree noise and 90 degree noise is that at higher noise levels, the navigator occasionally misjudges steps and clips the corners of obstacles. Also, random depth map errors occasionally cause the navigator to veer unnecessarily. Figure 12C shows two example paths taken by the navigator in the presence of 100% speed noise. This 100% speed noise is guaranteed to degrade the performance of the depth map by at least that amount, and one can see by the x's that the navigator is sometimes avoiding ghosts. However, on average the ghosts tend to cancel each other, and the network extracts useful information about the true location of the obstacles.

Finally, Figure 13 shows examples of motion pursuit simulations that require all elements of the visual navigation network. Aperture noise of ± 30 degrees was used in these simulations. One navigator, designated by

the plus symbol, attempts to reach a goal indicated by a plus sign at the top of the picture. A second navigator, designated by the open circle symbol, is slightly faster and uses the first navigator's changing location as its goal. The second navigator uses its independent motion map to determine the location of the first navigator. As shown in the two simulations of Figure 13, the second navigator is typically successful in its attempt to overtake the slower navigator.

5 Concluding Remarks

The network described in this article was developed primarily as a module for autonomous robot navigation. Its features include self-organization, fast learning, noise tolerance, operation in the presence of eye movements, and reasonable memory and computational demands. This system represents one of the first self-tuning systems to attempt seriously to solve the problem of navigation using optic flow, although the structure and learning principles were inspired by several previous computational and neural models (Hatsopoulos & Warren, 1991; Lappe & Rauschecker, 1993, 1995; Perrone, 1992). Based on the good performance seen in the simplified simulations, it is expected that the system will also perform adequately on real images. Work is in progress to transfer these algorithms to mobile robots. Efforts spent developing a self-tuning system reduce the burden of software rewrites that would otherwise be required to accommodate robot hardware changes. The current system could be expanded by introducing feedback between the modules. For example, independently moving objects could be detected more accurately by creating a feedback loop that removes pixels containing suspected independently moving objects from the normalization field. Therefore, the next iteration of the heading estimate will be less corrupted than the previous, which will improve the performance of both the depth and motion maps.

Although this was not the primary goal of the current model, many of its cell types also show similarities to MT and MST cell properties. Cells in MT are sensitive to the orientation and velocity of visual stimuli (Rodman & Albright, 1987), as are cells in the motion detection, translation, and normalization fields in the model. Many cells in MST respond maximally to radially expanding patterns and patterns that include expansion and full field translation, suggesting a role in heading detection and/or depth estimation (Lagae, Maes, Raiguel, Xiao, & Orban, 1994); cells in the heading map are similarly tuned. MST receives strong fiber projections from MT (e.g., Maunsell & Van Essen, 1983; Ungerlieder & Desimone, 1986), as do heading map cells in the normalization field. Cells in MT and MST have complex properties, however, that are by no means completely explained by the current model or other models of heading direction (Lagae et al., 1994; Graziano, Andersen, & Snowden, 1994).

To the extent that the present model does capture brain heuristics, the use of eye velocity information to nullify the rotational component of the flow field suggests that MST cells may change their flow field sensitivities in the presence of eye movements. Bradley, Maxwell, Andersen, Banks, and Shenoy (1996) have presented evidence consistent with this hypothesis by showing that during an active pursuit eye movement, expansion cells shift their preferred focus of expansion in a direction that would compensate for the eye rotation. This shift does not occur during a simulated rotation, suggesting that it is due to extraretinal information.

Finally, it must be noted that optic flow alone can be only an imperfect indication of object motions and distances. For example, no optic flow is available for objects centered at the FOE, and thus the distance of these objects cannot be determined using optic flow until they become close enough to subtend a significant portion of the retina away from the FOE. An independently moving object that is heading directly down the line of sight through the FOE will generate flow vectors that are directionally consistent with the flow vectors caused by the translational motion of the navigator, and such an object will thus not be identified as moving by the independent motion map. However, such an object would correctly register a shorter time to collision in the depth map, thus allowing the navigator to move around it even though it was not identified as independently moving. An independently moving object that covers a large portion of the retina will generate false heading estimates. This is evidenced in humans by the mistaken perception that their stationary car is rolling because the truck next to them is moving slowly in the opposite direction. Despite these shortcomings, optic flow is a rich and usually reliable source of information that should not be dismissed simply because it is difficult to measure and provides misleading or no information in certain circumstances. The purpose of our architecture is to extract as much information from optic flow as possible using a self-tuning network with relatively uncomplicated computational elements. Our approach distinguishes itself from previous work by providing a unified treatment of heading, depth, and independent object motion within a single architecture. In addition, our approach deals with the effects of self-generated eye motions thoroughly and effectively and has the unique advantage of self-organization. We have shown, using simplified simulations, that our network is able to provide sufficiently robust information to approach targets while avoiding obstacles, even if those targets are moving. This does not mean that our system, or any other system that attempts to process optic flow alone for that matter, is sufficient to allow error-free navigation in all real-world situations. Robust navigation requires the integration of many information sources, and our network has been designed to offer one piece of the puzzle by contributing useful heading, range, and independent motion estimates that a robot may use to aid in tasks such as moving object pursuit and obstacle avoidance.

Acknowledgments

S. C. was supported in part by the Defense Advanced Research Projects Agency (ONR N00014-92-J-4015) and the Office of Naval Research (ONR N00014-91-J-4100, ONR N00014-92-1309, and ONR N00014-95-1-0657). S. G. was supported in part by the Office of Naval Research (ONR N00014-95-1-0409 and ONR N00014-95-1-0657). F. H. G. was supported in part by the Alfred P. Sloan Foundation and the Office of Naval Research (ONR N00014-95-1-0409). The authors thank Diana Myers for her assistance in the preparation of the manuscript.

References

- Banks, M. S., Ehrlich, S. M., Backus, B. T., & Crowell, J. A. (1996). Estimating heading during real and simulated eye movements. *Vision Research*, *36*, 431–443.
- Bradley, D. C., Maxwell, M., Andersen, R. A., Banks, M. S., & Shenoy, K. V. (1996). Mechanisms of heading perception in primate visual cortex. *Science*, *273*, 1544–1547.
- Brenner, E. (1991). Judging object motion during smooth pursuit eye movements: The role of optic flow. *Vision Research*, *31*, 1893–1902.
- Bridle, J. S. (1989). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In F. Fougelman-Soulie & J. Herrault (Eds.), *Neuro-computing: Algorithms, architectures, and applications*. Berlin: Springer-Verlag.
- Bullock, D., & Grossberg, S. (1988). Neural dynamics of planned arm movements: Emergent invariants and speed-accuracy properties during trajectory formation. *Psychological Review*, *95*, 49–90.
- Chey, J., Grossberg, S., & Mingolla, E. (1996). Neural dynamics of motion grouping: From aperture ambiguity to object speed and direction. *Journal of the Optical Society of America A* (*14*), 2570–2594.
- Cornilleau-Peres, V., & Droulez, J. (1993). Stereo-motion cooperation and the use of motion disparity in the visual perception of 3-D structure. *Perception and Psychophysics*, *54*, 223–239.
- Crowell, J., & Banks, M. (1993). Perceiving heading with different retinal regions and types of optic flow. *Perception and Psychophysics*, *53*(3), 325–337.
- Cutting, J. E. (1986). *Perception with an eye towards motion*. Cambridge, MA: MIT Press.
- Dron, L. (1993). Dynamic camera self-calibration from controlled motion sequences. In *Proceedings of the 1993 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 501–506). Los Alamitos, CA: IEEE Computer Society Press.
- Gaudiano, P., & Grossberg, S. (1991). Vector associative maps: Unsupervised real-time error-based learning and control of movement trajectories. *Neural Networks*, *4*, 147–183.

- Gaudio, P., Zalama, E., & Lopez-Coronado, J. (1996). An unsupervised neural network for low-level control of a wheeled mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics*, 26, 485–496.
- Gibson, J. J. (1950). *The perception of the visual world*. Boston: Houghton Mifflin.
- Graziano, M. S. A., Andersen, R. A., & Snowden, R. (1994). Tuning of MST neurons in macaque MST. *Society for Neuroscience Abstracts* 16, 7.
- Grossberg, S. (1976). Adaptive pattern classification and universal recoding. I: Parallel development and coding of neural feature detectors. *Biological Cybernetics*, 23, 121–134.
- Grossberg, S., & Kuperstein, M. (1989). *Neural dynamics of adaptive sensory-motor control* (Exp. ed.). Elmsford, NY: Pergamon Press.
- Hatsopoulos, N. G., & Warren, W. H., Jr. (1991). Visual navigation with a neural network. *Neural Networks*, 4, 303–317.
- Heeger, D. J., & Jepson, A. (1990). Visual perception of three-dimensional motion. *Neural Computation*, 2, 129–137.
- Heikkonen, J., & Oja, E. (1993). Self-organizing maps for visually guided collision-free navigation. In *Proceedings of 1993 International Joint Conference on Neural Networks* (Vol. 1, pp. 669–672). New York: IEEE.
- Hildreth, E. C. (1992). Recovering heading for visually-guided navigation. *Vision Research*, 32, 1177–1192.
- Hildreth, E. C., Grzywacz, N. M., Adelson, E. H., & Inada, V. K. (1990). The perceptual buildup of three-dimensional structure from motion. *Perception and Psychophysics*, 48, 19–36.
- Holt, R. J., & Netravali, A. N. (1991). Camera calibration problem: Some new results. *CVGIP: Image Understanding*, 54, 368–383.
- Koenderink, J. J., & van Doorn, A. J. (1975). Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer. *Optica Acta*, 22, 773–791.
- Koenderink, J. J., & van Doorn, A. J. (1987). Facts on optic flow. *Biological Cybernetics*, 56, 247–254.
- Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, 59–69.
- Lagae, L., Maes, H., Raiguel, S., Xiao, D.-K., & Orban, G. A. (1994). Responses of macaque STS neurons to optic flow components: A comparison of areas MT and MST. *Journal of Neurophysiology*, 71, 1597–1626.
- Lappe, M., & Rauschecker, J. P. (1993). A neural network for the processing of optic flow from ego-motion in man and higher mammals. *Neural Computation*, 5, 374–391.
- Lappe, M., & Rauschecker, J. P. (1995). Motion anisotropies and heading detection. *Biological Cybernetics*, 72, 261–277.
- Longuet-Higgins, H. C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, 293, 133–135.
- Longuet-Higgins, H. C., & Prazdny, K. (1981). The interpretation of moving retinal images. *Proceedings of the Royal Society of London B*, 208, 385–397.
- Luong, Q.-T., & Faugeras, O. D. (1992). Self-calibration of a camera using multiple images. In *Proceedings of the 11th IAPR International Conference on Pattern Recognition* (Vol. 1, pp. 9–12). Los Alamitos, CA: IEEE Computer Society Press.

- Maunsell, J. H. R., & Van Essen, D. C. (1983). The connections of the middle temporal visual area (MT) and their relationship to a cortical hierarchy in the macaque monkey. *Journal of Neuroscience*, *3*, 2563–2586.
- Monen, J., & Brenner, E. (1994). Detecting changes in one's own velocity from the optic flow. *Perception*, *23*, 681–690.
- Orban, G. A., Lagae, L., Raiguel, S., Xiao, D.-K., & Maes, H. (1995). The speed-tuning of medial superior temporal (MST) cell responses to optic-flow components. *Perception*, *24*, 269–285.
- Perrone, J. A. (1992). Model for the computation of self-motion in biological systems. *Journal of the Optical Society of America A*, *9*, 1–18.
- Prazdny, K. (1980). Egomotion and relative depth map from optical flow. *Biological Cybernetics*, *36*, 87–102.
- Regan, D., & Beverly, K. I. (1982). How do we avoid confounding the direction we are looking and the direction we are moving? *Science*, *215*, 194–196.
- Rieger, J. H., & Lawton, D. T. (1985). Processing differential image motion. *Journal of the Optical Society of America A*, *2*, 354–359.
- Rieger, J. H., & Toet, L. (1985). Human visual navigation in the presence of 3D rotations. *Biological Cybernetics*, *52*, 377–381.
- Rodman, H. R., & Albright, T. D. (1987). Coding of visual stimulus velocity in area MT of the Macaque. *Vision Research*, *27*, 2035–2048.
- Royden, C. S. (1994). Analysis of misperceived observer motion during simulated eye rotations. *Vision Research*, *34*, 3215–3222.
- Royden, C. S., Banks, M. S., & Crowell, J. A. (1992). The perception of heading during eye movements. *Nature*, *360*, 583–585.
- Royden, C. S., Crowell, J. A., & Banks, M. S. (1994). Estimating heading during eye movements. *Vision Research*, *34*, 3197–3214.
- Simpson, W. A. (1988). Depth discrimination from optic flow. *Perception*, *17*, 497–512.
- Simpson, W. A. (1993). Optic flow and depth perception. *Spatial Vision*, *7*, 35–75.
- Tal, D., & Schwartz, E. (1994). Weber-Fechner transduction: A logarithmic compressive nonlinearity is a generic property of integrate and fire neurons. In *Proceedings of the World Congress on Neural Networks* (Vol. 4, pp. 360–367). Hillsdale, NJ: Erlbaum.
- Tanaka, K., & Saito, H. (1989). Analysis of motion of the visual field by direction, expansion/contraction, and rotation cells clustered in the dorsal part of the medial superior temporal area of the macaque monkey. *Journal of Neurophysiology*, *62*, 626–641.
- Treue, S., Andersen, R. A., Ando, H., & Hildreth, E. C. (1995). Structure-from-motion: Perceptual evidence for surface interpolation. *Vision Research*, *35*, 139–148.
- Ungerleider, L. G., & Desimone, R. (1986). Cortical connections of visual area MT in the Macaque. *Journal of Comparative Neurology*, *248*, 190–222.
- van den Berg, A. V. (1992). Robustness of perception of heading from optic flow. *Vision Research*, *32*, 1285–1296.
- van den Berg, A. V. (1993). Perception of heading. *Nature*, *365*, 497–498.
- von der Malsburg, C. (1973). Self-organization of orientation sensitive cells in the striata cortex. *Kybernetik*, *14*, 85–100.

- Wallach, H. (1976). *On perception*. New York: Quadrangle.
- Wang, H. T., Mathur, B. P., & Koch, C. (1989). Computing optical flow in the primate visual system. *Neural Computation*, *1*, 92–103.
- Warren, W. H., Jr. (1995). Self-motion: Visual perception and visual control. In W. Epstein & S. Rogers (Eds.), *Handbook of perception and cognition: Perception of space and motion* (pp. 263–325). San Diego: Academic Press.
- Warren, W. H., Jr. (1996). The state of flow. In T. Watanabe (Ed.), *High level motion processing*. Cambridge, MA: MIT Press.
- Warren, W. H., Jr., & Hannon, D. J. (1988). Direction of self-motion is perceived from optical flow. *Nature (London)*, *336*, 162–163.
- Warren, W. H., Jr., & Hannon, D. J. (1990). Eye movements and optical flow. *Journal of the Optical Society of America A*, *7*, 160–169.
- Warren, W. H., Jr., & Kurtz, K. J. (1992). The role of central and peripheral vision in perceiving the direction of self-motion. *Perception and Psychophysics*, *51*, 443–454.
- Warren, W. H., Jr., Li, L. Y., Ehrlich, S. M., Crowell, J. A., & Banks, M. S. (1996). Perception of heading during eye movements uses both optic flow and eye position information. *Investigative Ophthalmology and Vision Science*, *37*.
- Waxman, A. M., & Ullman, S. (1985). Surface structure and three-dimensional motion from image flow kinematics. *International Journal of Robotics Research*, *4*, 72–94.
- Waxman, A. M., & Wohn, K. (1988). Image flow theory: A framework for 3-D inference from time-varying imagery. In C. Brown (Ed.), *Advances in computer vision* (Vol. 1, pp. 165–224). Hillsdale, NJ: Erlbaum.
- Zalama, E., Gaudiano, P., & Lopez-Coronado, J. (1995). A real-time, unsupervised neural network for the low-level control of a mobile robot in a nonstationary environment. *Neural Networks*, *8*, 103–123.