12. Fast Processing of Image Motion Patterns Arising from 3-D Translational Motion

Venkataraman Sundareswaran[1,2], Scott A. Beardsley[1] and Lucia M. Vaina[1,3]

[1] Brain and Vision Research Laboratory, Department of Biomedical Engineering
Boston University, Boston, MA, USA

[2] Rockwell Scientific Company
Thousand Oaks, CA, USA

[3] Department of Neurology
Harvard Medical School, Boston, MA, USA

1 INTRODUCTION

As we drive on the highway, walk down the street, or move around in the house, we make constant use of a complex perceptual mechanism: vision. In these situations, we rely on the visual system's ability to process the perceived motion of the visual scene across the retina, termed optic flow, for our perception and estimates of self-motion. When self-motion is known, it is possible to identify obstacles and moving objects, determine *time to collision*, and the three-dimensional structure of the environment.

For most people, the perception of motion and its application in everyday tasks are seemingly effortless. In some cases, however, visual impairments seriously degrade motion perception to the point of reduced functionality in real world tasks such as walking and navigation, in which estimate of self-motion is crucial. For these impairments, mobility-assistive devices capable of processing visual motion in real-time and calculating perceptually relevant information would be invaluable. However, theoretical and computational limitations significantly restrict our ability to develop devices capable of processing the full visual motion field in real-time.

Several methods have been proposed in the literature to compute the motion of the observer from optic flow; for reviews see Barron, et al., 1994, Heeger & Jepson, 1992, or Hummel & Sundareswaran, 1993. In these

methods, however, the *aperture problem* restricts the recovery of the full optical flow field. Specifically, only the component of optical flow in the direction of the local image intensity gradient can be reliably computed without the application of computationally expensive local and/or global smoothness constraints (Adelson & Bergen, 1986, Horn & Schunck, 1981, Nagel & Enkelmann, 1986). For estimates of self-motion based on full optic flow, the application of such constraints via iterative or least-squares techniques significantly increases the amount of pre-processing necessary to recover self-motion estimates, limiting their real-time utility.

Computationally efficient algorithms based on the local projection of optical flow, termed "normal flow", have been developed to recover estimates of *time to contact* and heading direction in real-time (Aloimonos & Duric, 1992, Alok & Aditya, 1995, Camus, 1995, Coombs, et al., 1998, Fermuller & Aloimonos, 1995, Herwig, et al., 1998, Horn & Weldon, 1988, Negahdaripour & Horn, 1989, Sinclair, et al., 1994). In a robotic navigation task, Herwig et al. (1998) used normal flow in conjunction with a cost function based on the half-plane constraint (Aloimonos, et al., 1993) to estimate heading direction. They found that robust heading performance could be achieved in the presence of statistical noise and for small amounts of rotation (<0.05 rad/s). In a real-time wandering task, Coombs et al. (1998) used normal peripheral flow in conjunction with flow field divergence estimates to compute *time to contact* for obstacle avoidance. Together with a direction centering mechanism based on the maximal peripheral flows, they demonstrated that robust image motion cues such as *time to contact* could be extracted from normal flow and used in real-time to safely navigate complex visual environments for extended periods of time.

In this chapter, we present a method for the *rapid* calculation of the observer's translational velocity based on normal flow estimates and we demonstrate its applicability in two tasks: in the first a camera is actively controlled to orient itself in the direction of translation, and in the second, the method is used for fast collision detection. The goal of this work is to develop methods for performing tasks using visual information rather than building a representation based on visual processing. Together with the continued reductions in the size and increasing power of computer systems we expect that such methods could play a central role in the real-time vision processing applications necessary for the development of autonomous robotic systems and mobility assistive devices for the visually impaired.

2 BACKGROUND

The relative motion of an observer with respect to a rigid, unknown environment can be represented in a coordinate system centered in the observer's center of projection. We will use perspective projection on a plane as the camera model. The two-dimensional motion of the image intensity pattern on the image plane, namely the *optic flow*, depends on the three-dimensional motion (translation and rotation) of the observer. For instance, an optic flow pattern radially expanding from an image location corresponds to translational motion towards the 3-D location projected at that image location. Rotational motion produces elliptical patterns, and a combination of the two (translation and rotation) produces complex optic flow patterns that are the summation of the individual patterns. In the work outlined below we are interested in the *inverse* problem of determining the motion of the observer from optic flow patterns.

Any method to compute optic flow must deal with the aperture problem which restricts measurement of the local motion component to the direction of the local intensity gradient (Horn, 1985). This direction is normal to the local *edge*; hence the 2-D motion along this direction is referred to as "normal flow." Typically, a regularization approach is taken to compensate for the lack of motion information along the edge (Anandan, 1989, Heeger, 1988, Hildreth, 1984, Horn, 1985). However, regularization involves a function minimization process that can be time-consuming; for an exception, see Hildreth (1984). Also, regularization assumes smoothness across the optic flow field and often introduces deviations from the actual motion field, in turn reducing the accuracy of 3-D motion and structure calculations based on the optic flow. Due to these reasons, we are interested in developing a method that computes relative 3-D motion directly from the normal flow. The goal is to develop a method that is approximate but fast enough to be useful in real-time or near- real-time applications.

3 PROPOSED METHOD

The proposed method is based on the well-known observation that for *translational* motion of the camera, image motion everywhere is directed away from a singular point corresponding to the projection of the translation vector on the image plane. This point, also called the Focus of Expansion (FOE), corresponds to the intersection of the optic flow vectors depicting the image motion. The FOE can be determined by a simple analysis of the direction of the components of the optic flow field vectors, as described

below. In the current approach we focus on translational motion under gaze-stabilized viewing conditions.

3.1 Approach to Determine the FOE from Optic Flow

In a rigid environment the sign of the horizontal component of an optic flow vector (e.g. positive: rightward, negative: leftward) is determined by the location of the FOE. At a qualitative level, the distribution of the component signs in an image region is determined by the position of the FOE relative to the region. For example, if the FOE is in the middle of the region, motion vectors in one half of the region will have negative sign, and those in the other half will have positive sign. This observation suggests the possibility to use the distribution of the signs to determine the horizontal location of the FOE. In a similar fashion, by using the signs of the vertical components, the vertical coordinate of the FOE may be determined.

We are interested in examining the applicability of this simple approach to determine the FOE based on the normal flow field instead of the full optic flow field. In the proof of the following theorem we demonstrate that the use of the normal flow field in this manner is likely to yield usable FOE estimates under the previously prescribed conditions.

Theorem: *With probability greater than chance, the signs of the components of a normal flow vector agree with the signs of the components of the corresponding optic flow vector, providing the local intensity gradient direction is distributed uniformly*

Proof: Let the local gradient subtend an angle α, and the optic flow vector subtend an angle θ, with respect to the x-axis. Without loss of generality, we can assume that both α and θ span the range $[0, \pi/2]$, where the local intensity gradient and the optic flow vector lie along the unit vectors $[\cos(\alpha), \sin(\alpha)]$ and $[\cos(\theta), \sin(\theta)]$ respectively. Since, by definition, the normal flow vector corresponds to the component of the optic flow that is parallel to the local intensity gradient (i.e. perpendicular to the local edge) the normal flow direction is given by $\cos(\theta-\alpha)[\cos(\alpha), \sin(\alpha)]$.

For the x-component of this (normal) vector to have the same sign as the x-component of the optic flow vector,

$$\cos(\theta-\alpha)\cos(\alpha)\cos(\theta) > 0 \qquad (1)$$

We consider two possibilities:

$$\begin{array}{ll} 1. & \cos(\theta-\alpha)>0 \quad and \quad \cos(\alpha)\cos(\theta)>0 \\ 2. & \cos(\theta-\alpha)<0 \quad and \quad \cos(\alpha)\cos(\theta)<0 \end{array} \tag{2}$$

The first situation occurs if $|\theta-\alpha| < \pi/2$, and if α and θ are in the same vertical set of quadrants (first and fourth, or second and third). For a given θ, α has a range of π-θ for which an agreement of signs results. For the second situation, $|\pi + \theta - \alpha| < \pi/2$, and α and θ must be in the same horizontal set of quadrants (first and second, or third and fourth). This corresponds to a range of π-θ for α. Since the total possible range for α is 0 to 2π, assuming a uniform distribution for α, the probability that the signs of the x components agree is $(\pi - \theta)/\pi$. Using a similar argument, it can be easily shown that the probability that the y components agree is $(\pi/2+\theta)/\pi$. In passing, we note that for $\alpha = 0$ or π (vertical edges), the signs of the x components always agree!

In light of this theorem, we conclude that it is possible to determine FOE based on normal flow vectors. Below, we present results based on the application of this method to compute the FOE in real image sequences.

4 EXPERIMENTS

The procedure described above has been implemented on three different platforms: in the image processing package HIPS on a Sun SPARC IPX to determine FOE in off-line image sequences, in a robot control program on a Sun SPARC 10 for controlling the motion of a six degree-of-freedom (DOF) robot in real-time, and in the public-domain software NIH Image for a real-time collision warning system.

4.1 FOE Calculation

Following the theorem in Section 3.1, the focus of expansion (FOE) is computed based on the principle that flow vectors are oriented in specific directions relative to the FOE. Specifically, the horizontal component h_L of an optic flow vector L to the left of the FOE points leftward while the horizontal component h_R of an optic flow vector R to the right of the FOE points rightward, as shown in Figure 1. In a full optic flow field the horizontal location of the FOE, corresponding to the position about which a majority of horizontal components diverge, can be estimated using a simple counting method to tally the signs of the horizontal components centered on each image location. At the point where the divergence is maximized, the difference between the number of h_L components to the left of the FOE and

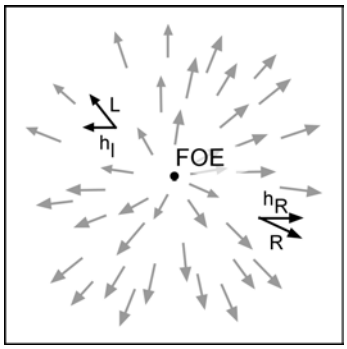

Figure 1. Schematic representation of two local optic flow components resulting from forward translation. For motion vectors located to the left (L) and right (R) of the Focus of Expansion (FOE) the horizontal components of motion point to the left and right respectively.

the number of h_R components to the right of the FOE will be minimized. Similarly we can estimate the vertical location of the FOE by identifying the position about which a majority of vertical components (v_U and v_D) diverge.

Here we extend this principle to the field of normal flow vectors. As outlined in the theorem in Section 3.1, we can reliably estimate the FOE from a normal flow vector field, providing that the intensity gradient (edge) distribution in the image is uniform (i.e., the probability that a certain orientation is seen in an image location is constant across orientations).

We implemented a version of the proposed method in HIPS to determine FOEs in standard image sequences. To calculate the normal flow vector fields, the images were first convolved with one-dimensional Gaussian derivative kernels to estimate the spatial derivatives of the intensity gradient. The standard deviation of the Gaussians was 0.75 pixels. The results of applying this kernel on an image are shown in Figure 2. The image, and its x- and y- spatial derivatives (I_x, I_y) are shown respectively from left to right. The temporal derivative (I_t) was computed as the pixel-wise difference between sequential pairs of image frames. Normal flow was then computed using the image flow constraint equation (Horn, 1985)

$$I_x u + I_y v + I_t = 0 \tag{3}$$

where I_x, I_y, and I_t correspond to the spatial (x,y) and temporal gradients in the local image intensity respectively and (u,v) correspond to the x- and y-components of motion.

The FOE coordinates were estimated from the fraction of rightward horizontal components and the fraction of downward vertical components of the normal flow. This method is admittedly crude because it does not consider

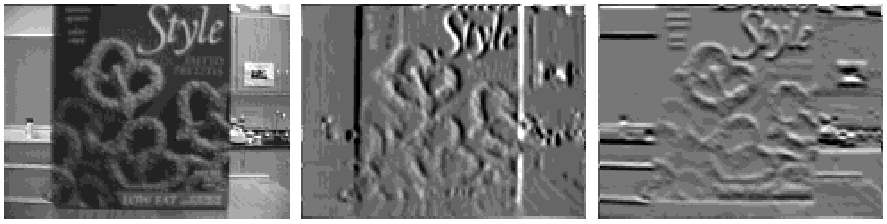

Figure 2. A sample image (left) together with the corresponding spatial gradients, I_x (middle) and I_y (right), formed by convolving the difference kernels with the image.

the detailed distribution of the signs (we used a more elaborate implementation in the collision detection application described later) and the temporal difference calculation does not incorporate temporal smoothing (which can improve the results for typical 3-D observer motion where transitions in the FOE are generally smooth). Using this straightforward procedure, we calculated the FOEs on four different standard image sequences, and found the FOEs to be qualitatively correct in each. The results are illustrated in Figure 3 where the FOE is marked as a dark dot within a white square (only typical results across the images sequences are shown).

Naturally, this method is limited because it is useful only for motion with pure translational velocity. The applicability of the method will be enhanced if we can demonstrate its utility in applications requiring rapid computation of the FOE such as control loops or fast collision detection. Accordingly, we describe below applications where we have used this fast approach to compute the FOE from the normal flow vector field.

4.2 Closed-Loop Control

The goal of this application was to control a camera so as to align its optical axis with the unknown direction of translation. The direction of translation (FOE) was computed using the method outlined above, and a standard visual servoing method (Espiau, et al., 1992) was used to calculate the rotational velocity control to accomplish the desired alignment. The overall control procedure, shown schematically in Figure 4 and explained in detail elsewhere (Bouthemy & Sundareswaran, 1993, Sundareswaran, et al., 1996, Sundareswaran, et al., 1994), resulted in correct alignment of the camera optical axis direction with the unknown direction of translation. A graph illustrating the convergence of these two directions in a typical experiment is shown in Figure 5.

Figure 3. Frames from standard image sequences showing the results of the FOE computations using the method proposed here. The FOE is shown in each case as a black dot embedded within a white square.

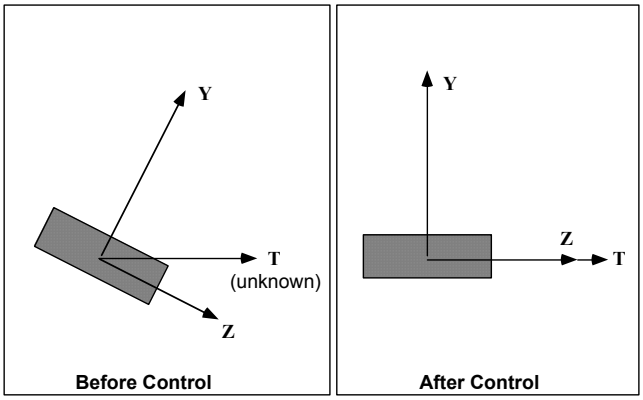

Figure 4. Schematic of the control system. Before control, the relative orientation between the camera optic axis (Z) and the translational direction is arbitrary. After control, they are aligned. A two-dimensional projection is shown for simplicity.

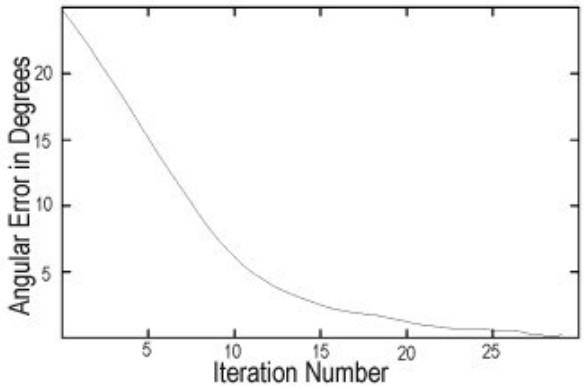

Figure 5. The graph shows the convergence in the angular difference between the camera optical axis direction and the direction of translation for an experiment performed with a camera mounted on a 6-DOF robot arm

4.3 Collision Detection

An important capability for mobile systems is to avoid collisions with static and mobile objects. Knowledge of self-motion is very useful to determine the *time to collision* (TTC) with static objects, and to segment independently moving objects. Using the method outlined above we calculated the FOE and estimated TTC by determining if the area surrounding the FOE was "zooming in" at a sufficiently fast rate to detect possible collision with that area.

The TTC can be estimated from the parameters of the first-order approximation to the optic flow field. The first-order (affine) approximation is given by

$$\begin{aligned} u &= a_1 + a_2 x + a_3 y \\ v &= a_4 + a_5 x + a_6 y \end{aligned} \tag{4}$$

where the parameters a_2 and a_6 encode the *divergence* of the flow. It can be shown that for surfaces that give rise to a flow that is nearly affine, the TTC is given by the expression

$$\tau = \frac{2}{a_2 + a_6} \tag{5}$$

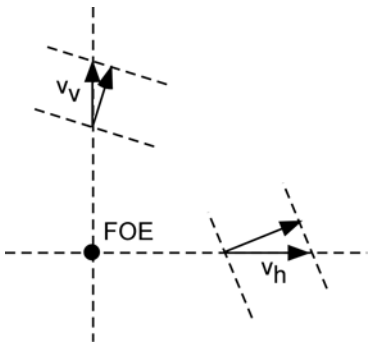

Figure 6. The affine parameters a_2 and a_6 can be estimated from points along the horizontal and vertical lines passing through the FOE as the distance normalized horizontal (v_v) and vertical (v_h) components respectively.

The parameters a_2 and a_6 can be computed from the normal flow over a 2-D image region. However, to simplify the computation, we consider only the horizontal line and the vertical line passing through the FOE. Figure 6 illustrates the simplification achieved by this choice of normal flow components.

Arbitrary local edge directions have been chosen to show the normal flow of two optic flow vectors, one along the vertical line and the other along the horizontal line passing through the FOE. Note that these optic flow vectors must lie along these lines because of the structure of translational flow fields. It can be easily shown that the magnitude of the horizontal optic flow vector is

$$v_h = \frac{I_t}{I_y} \tag{6}$$

and that of the vertical optic flow vector is

$$v_v = \frac{I_t}{I_x} \tag{7}$$

where (I) is the spatio-temporal intensity function. Using several measurements along the horizontal and vertical lines, we can calculate the desired affine parameters as follows

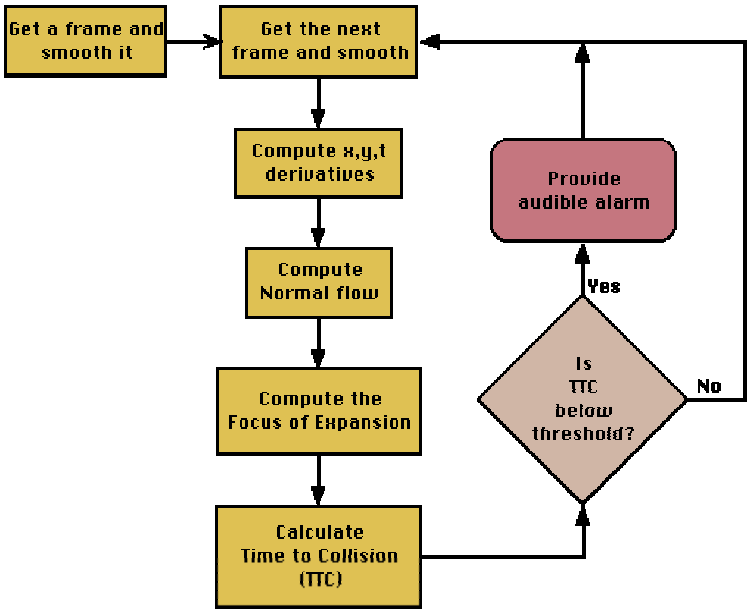

Figure 7. Flow chart of the collision warning method.

$$a_2 = \frac{1}{m}\sum_h \frac{v_h}{x_h}$$
$$a_6 = \frac{1}{n}\sum_v \frac{v_v}{y_v} \tag{8}$$

where x_h and y_v correspond to the transformed image coordinates of the horizontal and vertical normal flow components with the FOE as the origin, and *m* and *n* correspond to the number of horizontal and vertical motion components computed across the image.

We have implemented a demonstration in NIH Image, based on computation of the TTC on a PowerMacintosh 8500 with a camcorder connected to the built-in video port. The system generates warning beeps whenever an object approaches faster than a pre-set threshold to indicate a "low" time to collision value.

The difference between this system and a general motion sensor is that this system provides a warning only if there is an impending collision, but not if there is any other type of motion. The schematic in Figure 7 illustrates the operation of the system. Image sequences showing results of the collision

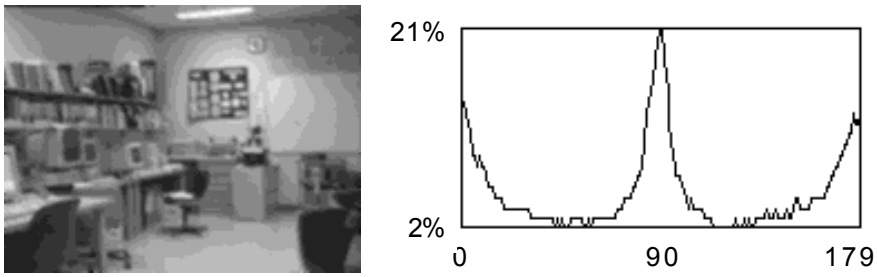

Figure 8. A sample frame from an indoor sequence and the corresponding distribution of edge orientations (across the whole sequence). The graph shows average percentage of pixels in each frame with a certain edge orientation (0 deg corresponds to horizontal edges, and 90 deg corresponds to vertical edges).

detection method can be accessed at http://www.bu.edu/bravi/research/MagicHat/MagicHat.html.

Incoming frames were smoothed using a built-in (NIH Image) weighted averaging filter, and spatial derivatives were computed using 3X3 difference filters. Temporal derivatives were computed by differencing (two) smoothed frames. The normal flow was computed as before using the optic flow constraint equation. The FOE was computed as the coordinate with the least number of "wrongly oriented" components on either side of it (e.g., for the horizontal coordinate of the FOE was chosen so that the sum of the number of rightward components to the left of it and the number of leftward components to the right of it was minimal).

The linear coefficients of the first-order approximation to the optic flow provide effective measures of the rate of expansion, and hence the time to collision. Thus the time to collision can be directly related to the coefficients of x and y in the first order expansion. The linear coefficients in a region around the FOE were estimated by considering the normal flow on the vertical and the horizontal lines passing through the FOE. The time to collision based on the estimated coefficients was thresholded, i.e., if it was below a threshold a warning beep was sounded.

To verify our assumption that the edge distribution is uniform, we implemented a method to histogram the orientation of moving edges in typical image sequences (using bins of 1 degree width), assuming that the edge distribution is a stationary process (probability distribution at a pixel location is the same as the spatial distribution). We observed empirically that for indoor image sequences the distribution of edge orientations was *not* uniform. We found a preponderance of vertical and horizontal edges, and a relatively uniform distribution across the other orientations (Figure 8).

This result, however, does not invalidate the method proposed here. Following the reasoning in the theorem in Section 3.1, it can be shown that for vertical edges, the sign of the x components of the optic flow and normal flow *always* agree; likewise for the horizontal edges, the sign of the y components *always* agree. Thus, the effect of the non-uniform edge distribution for indoor scenes only improves the performance of our implementation, as long as the camera pose is such that the vertical and horizontal edges of the scene are imaged as vertical or horizontal edges (i.e., near-zero camera tilt angle).

5 DISCUSSION

In this research we proposed the development of specific methods for performing tasks using visual information rather than building a representation based on visual processing. Toward this goal we have presented an approximate but fast method to locate the focus of expansion (FOE) based on the normal optic flow in an image sequence. Together with the use of a simple divergence estimator that does not require significant post-processing beyond the normal flow calculations, we have shown that the FOE in conventional image sequences can be reliably estimated in real-time and we have demonstrated its use in gaze control and collision detection.

The algorithm described here has several advantages that make it appealing as a method for extracting real-time FOE estimates for use in visually-guided tasks. First and foremost, the system can be implemented using off-the-shelf frame grabbers and cameras and the algorithm's performance can be easily scaled up to real-time levels using commercially available computer systems.

Second, no camera calibration information is required. While it is true that the inter-pixel distance is required for the TTC computation, it can be specified in arbitrary units (e.g., pixel count, as in our implementation) since the "collision sensitivity" or threshold of the system can be adjusted to compensate for the unknown scale factor.

Due to the simplicity and reliability of the computations, the approach can also be extended for use in low-light situations or infrared image sequences. Similarly, the approach can be easily adapted for tracking and motion encoding applications.

However, before the proposed method can be applied to more complex environments involving multiple motion sources it will be important to incorporate algorithms to segment the image motion and identify observer motion. This is a complex problem in itself and we do not propose a solution here. Nevertheless, we note from empirical observations that for the camera

alignment task, small object movement did not significantly affect the computation of the FOE and the subsequent convergence (Bouthemy & Sundareswaran, 1993).

In future work we will investigate these issues with the goal of generalizing the proposed algorithm to systematically more complex visual scenes. Together with the increase in computer power and portability, we expect that the ongoing development of robust and computationally efficient real-time methods will continue to facilitate application-driven research into autonomous robotic systems and mobility assistive devices for the visually impaired.

ACKNOWLEDGMENTS

This work was supported in part by NIH grant EY-2R01-07861-13 to LMV and NSF-SGER CDA-9528079 to LMV.

REFERENCES

Adelson, E.H., & Bergen, J.R. (1986). The extraction of spatio-temporal energy in human and machine vision. *Proceedings of the IEEE Workshop on Motion: Representation and Analysis* (pp. 151-155). Charleston, South Carolina.

Aloimonos, Y., & Duric, Z. (1992). Active egomotion estimation: a qualitative approach. *Second European Conference on Computer Vision* (pp. 497-510). Santa Margherita Ligure, Italy: Springer.

Aloimonos, Y., Rivlin, E., & Huang, L. (1993). Designing Visual Systems: Purposive Navigation. In: Y. Aloimonos (Ed.) *Active Vision* (pp. 47-102): Lawrence Erlbaum.

Alok, M., & Aditya, V. (1995). Real time vision system for collision detection. *J. Comput. Sci. Inform., Special Issue on "Robotics and Automation", 25 (1),* 174-208.

Anandan, P. (1989). A Computational Framework and an Algorithm for the Measurement of Visual Motion. *Int. J. Comput. Vision, 2*, 283-310.

Barron, J.L., Fleet, D.J., & Beauchemin, S.S. (1994). Performance of optical flow techniques. *Int. J. Comput. Vis*ion, 43-77.

Bouthemy, P., & Sundareswaran, V. (1993). Qualitative motion detection with a mobile and active camera. *Proceedings of the International Conference on Digital Signal Processing* (pp. 444-449). Cyprus.

Camus, T.A. (1995). Calculating time-to-contact using real-time quantized optical flow. (pp. 1-12). Tübingen, Germany: Max Planck Institute for Biological Cybernetics.

Coombs, D., Herman, M., Hong, T., & Nashman, M. (1998). Real-time obstacle avoidance using central flow divergence and peripheral flow. *IEEE Trans. Rob. Autom., 14 (1),* 49-59.

Espiau, B., Chaumette, F., & Rives, P. (1992). A new approach to visual servoing in robotics. *IEEE Trans. Rob. Autom., 8 (3),* 313-326.

Fermuller, C., & Aloimonos, Y. (1995). Direct perception of three-dimensional motion from patterns of visual motion. *Science, 270*, 1973-1976.

Heeger, D.J. (1988). Optical flow using spatiotemporal filters. *Int. J. Comput. Vision, 1*, 279-302.

Heeger, D.J., & Jepson, A.D. (1992). Subspace methods for recovering rigid motion I: Algorithm and implementation. *Int. J. Comput. Vision, 7 (2),* 95-117.

Herwig, C., Carmesin, H.-O., Hamker, F., & Wandtke, D. (1998). Real-time estimation of heading direction. *J. Real-Time Im., Special Issue on "Real Time Motion Analysis", 4 (1).*

Hildreth, E.C. (1984). *Computations Underlying the Measurement of Visual Motion.* (Cambridge, MA: MIT Press).

Horn, B. (1985). *Robot Vision.* (Cambridge MA: MIT Press).

Horn, B.K.P., & Schunck, B.G. (1981). Determining optical flow. *Artif. Intell., 17*, 185-203.

Horn, B.K.P., & Weldon, E.J. (1988). Direct methods for recovering motion. *Int. J. Comput. Vision, 2 (1),* 51-76.

Hummel, R., & Sundareswaran, V. (1993). Motion Parameter Estimation from Global Flow Field Data. *IEEE Trans. Pattern Anal. Mach. Intell., 15 (5),* 459-476.

Nagel, H.H., & Enkelmann, W. (1986). An estimation of smoothness constraints for the estimation of displacement vector fields from from image sequences. *IEEE Trans. Pattern Anal. Mach. Intell., PAMI-8*, 565-593.

Negahdaripour, S., & Horn, B.K.P. (1989). Direct method for locating the focus of expansion. *Comput. Vis. Graph. Im. Process., 46 (3),* 303-326.

Sinclair, D., Blake, A., & Murray, D. (1994). Robust estimating of egomotion from normal flow. *Int. J. Comput. Vision, 13*, 57-69.

Sundareswaran, V., Bouthemy, P., & Chaumette, F. (1996). Exploiting image motion for active vision in a visual servoing framework. *Int. J. Rob.. Res., 15 (6),* 629-645.

Sundareswaran, V., Chaumette, F., & Bouthemy, P. (1994). Visual servoing using image motion information. *Proceedings of the IAPR/IEEE Workshop on Visual Behavior* (pp. 102-106). Seattle.