Stereo and Color Analysis for Dynamic Obstacle Avoidance

Tamer F. Rabie and Demetri Terzopoulos

Department of Computer Science, University of Toronto
10 King's College Road, Toronto, Ontario, M5S 3G4
e-mail: {tamer|dt}@cs.toronto.edu

Abstract

We develop a vision system for highly mobile autonomous agents that is capable of dynamic obstacle avoidance. We demonstrate the robust performance of the system in artificial animals with directable, foveated eyes, situated in physics-based virtual worlds. Through active perception, each agent controls its eyes and body by continuously analyzing photorealistic binocular retinal image streams. The vision system computes stereo disparity and segments looming targets in the low-resolution visual periphery while controlling eye movements to track an object fixated in the high-resolution fovea. It matches segmented targets against mental models of colored objects of interest in order to decide whether the segmented objects are harmless or represent dangerous obstacles. The latter are localized, enabling the artificial animal to exercise the sensorimotor control necessary to avoid collision.

1 Introduction

Animals are active observers of their environment [11]. This fact has inspired a trend in computer vision popularly known as "active vision" [2, 3, 26]. The recently proposed *animat vision* paradigm offers a new approach to developing biomimetic active vision systems and experimenting with them [28]. Rather than allow the limitations of available robot hardware to hamper research, animat vision prescribes the use of virtual robots that take the form of realistic artificial animals, or animats, situated in physics-based virtual worlds. Animats are autonomous virtual agents possessing highly mobile, muscle-actuated bodies, as well as brains with motor, perception, behavior and learning centers. In the perception center of the animat's brain, computer vision algorithms continually analyze incoming perceptual information. Based on this analysis, the behavior center dispatches motor commands to the animat's body, thus forming a complete sensorimotor control system.

Biological creatures move through the world with little apparent effort. Many do so using eyes with a high-acuity fovea covering only a small fraction of a visual field whose resolution decreases monotonically towards the periphery. For example, a person can safely navigate a hallway while reading a book that occupies her foveal vision while avoiding potential threats identified through peripheral vision. In fact, a great deal of mobility can be supported by low resolution peripheral vision, freeing the small, high resolution visual area to attend to important matters during navigation [8]. Spatially nonuniform retinal imaging provides opportunities for increased computational efficiency through economization of photoreceptors and focus of attention, but it forces the visual system to solve problems that do not generally arise with a uniform field of view. A key problem is determining how to avoid danger when obstacles are detected in the low resolution periphery while focusing attention on an object of interest fixated in the high resolution fovea. In this paper we present a solution to this problem through the combined exploitation of color information and depth information from stereo disparity.

Figure 1: Artificial fishes swimming among aquatic plants in a physics-based virtual marine environment.

Building upon the animat vision paradigm, the stereo and color based motor control algorithms that we propose in this paper are implemented and evaluated within artificial fishes in a virtual marine world (Fig. 1). The fish animats are the result of research in the domain of artificial life (see [29] for the details). In the present work, the fish animat serves as an autonomous mobile robot inhabiting a photorealistic, dynamic environment. Our new navigation algorithms significantly enhance the prototype animat vision system implemented in prior work [28, 20, 27]. They support more robust vision-guided navigation, including obstacle recognition and avoidance. We briefly review the animat vision system in the next section before presenting, in the subsequent sections, our new work on integrating stereo disparity and color analysis for animat navigation and perception.

2 A Prototype Animat Vision System

The basic functionality of the animat vision system, which is described in detail in [28], starts with binocular perspective projection of the color 3D world onto the animat's 2D retinas. Retinal imaging is accomplished by photorealistic graphics rendering of the world from the animat's point of view. This projection respects occlusion relationships among objects. It forms spatially variant visual fields with high resolution foveas and progressively lower resolution peripheries. Based on an analysis of the incoming color retinal image stream, the visual center of the animat's brain supplies saccade control signals to its eyes to stabilize the visual fields during locomotion, to attend to interesting targets based on color, and to keep moving targets fixated. The artificial fish is thus able to approach and track other artificial fishes visually.

2.1 Eyes and Retinal Imaging

The artificial fish has binocular vision. The movements of each eye are controlled through two gaze angles (θ, ϕ) which specify the horizontal and vertical rotation of the eyeball, respectively. The angles are given with respect to the head coordinate frame, such that the eye is looking straight ahead when $\theta = \phi = 0°$.

Each eye is implemented as four coaxial virtual cameras to approximate the spatially nonuniform, foveal/peripheral imaging capabilities typical of biological eyes. Fig. 2(a) shows an example of the 64×64 images that are rendered by the coaxial cameras in each eye (rendering employs the OpenGL library and graphics pipeline on Silicon Graphics workstations). The level $l = 0$ camera has the widest field of view (about $120°$) and the lowest resolution. The resolution increases and the field of view decreases with increasing l. The highest resolution image at level $l = 3$ is the fovea and the other images form the visual periphery. Fig. 2(b) shows the 512×512 binocular retinal images composited from the coaxial images at the top of the figure. To reveal the retinal image structure in the figure, we have placed a white border around each magnified component image. Vision algorithms which process the four 64×64 component images are 16 times more efficient than those that process a uniform 512×512 retinal image.

2.2 Foveation by Color Object Detection

The brain of the artificial fish stores a set of color models of objects that are of interest to it. For instance, if the fish is a predator, it would possess models of prey fish. The mental models are stored as a list of 64×64 RGB color images.

To detect and localize any target that may be imaged in the low resolution periphery of its retinas, the animat vision system of the fish employs an improved version of a color indexing algorithm proposed by Swain [25].[1] Since each model object has a unique color histogram signature, it can be detected in the retinal image by histogram intersection and localized by histogram backprojection.

2.3 Saccadic Eye Movements

When a target is detected in the visual periphery, the eyes will saccade to the angular offset of the object to bring it within the fovea. With the object in the high resolution fovea, a more accurate foveation is obtained by a second pass of histogram backprojection. A second saccade typically centers the object accurately in both left and right foveas, thus achieving vergence. The saccades are performed by incrementing the gaze angles (θ, ϕ) in order to rotate the eyes to the required gaze direction.

2.4 Visual Field Stabilization using Optical Flow

It is necessary to stabilize the visual field of the artificial fish because its body undulates as it swims. Once a target is verged in both foveas, the stabilization process assumes the task of keeping the target foveated during locomotion.

Stabilization is achieved by computing the overall translational displacement (u, v) of intensities between the current foveal image and that from the previous time instant, and updating the gaze angles to compensate. The displacement is computed as a translational offset in the retinotopic coordinate system by a least squares minimization of the optical flow between successive image frames [13].

The optical flow stabilization method is robust only for small displacements between frames. Consequently, when the displacement of the target between frames is large enough that the method is likely to produce bad estimates, the foveation module is invoked to re-detect and re-foveate the target as described earlier. Each eye is controlled independently during foveation and stabilization of a target. Hence, the two retinal images must be correlated to keep them verged accurately on the target.

[1] Our improvements, which include iterative model histogram scaling and weighted histograms, make the technique much more robust against the large variations in scale that occur in our application. The details of the improved algorithm are presented in [28].

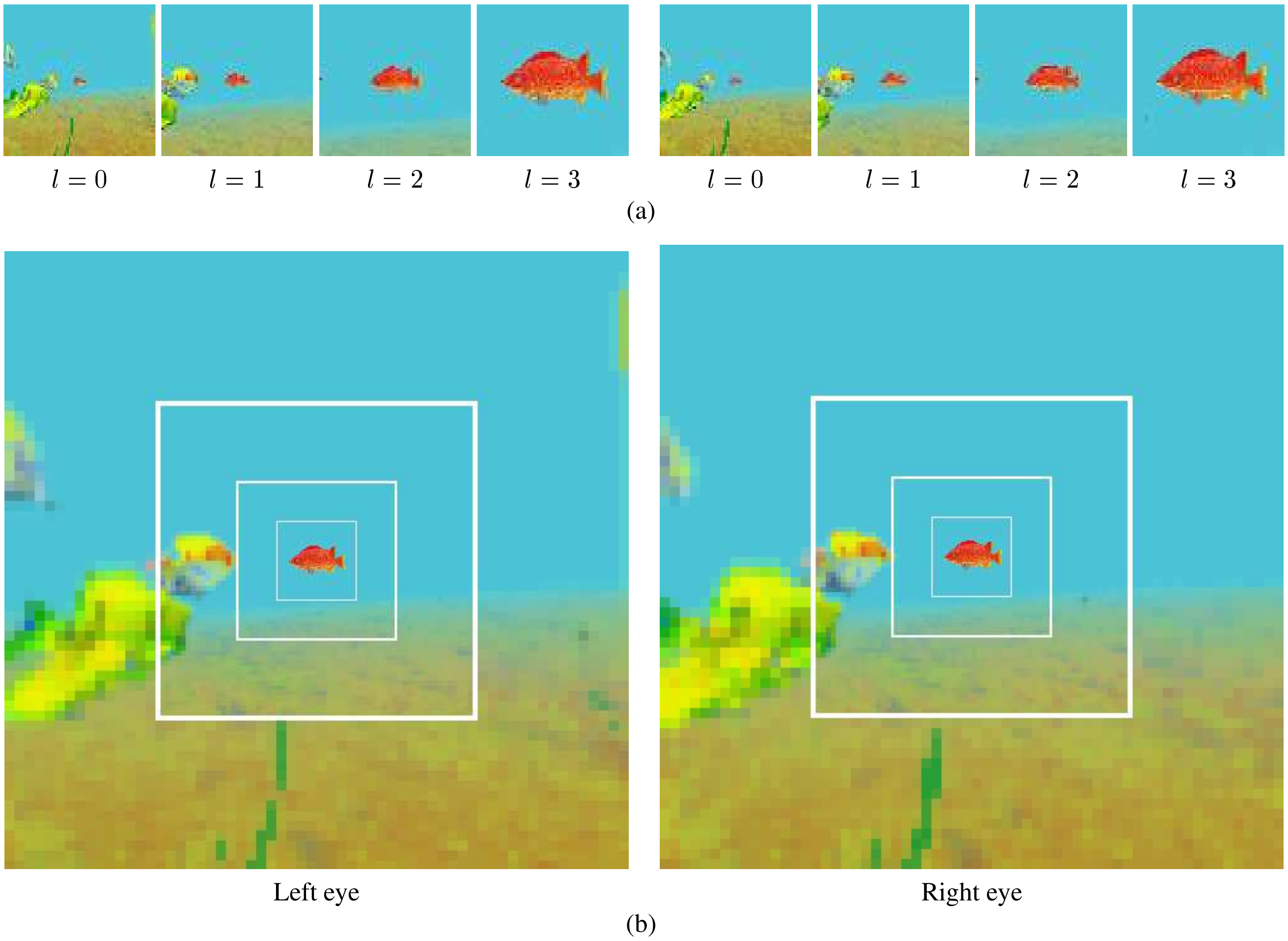


Figure 2: Binocular retinal imaging. (a) 4 component images; $l = 0, 1, 2$, are peripheral images; $l = 3$ is foveal image. (b) Composited retinal images (borders of composited component images are shown in white).

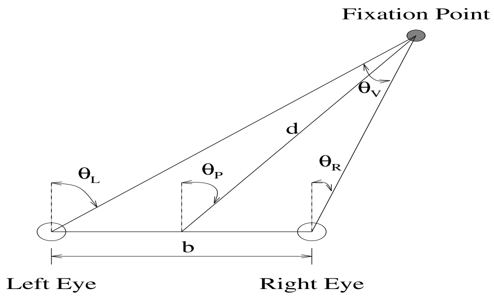


Figure 3: Gaze angles and range to target geometry.

2.5 Vision-Guided Navigation

The artificial fish can also employ the gaze direction (i.e., the gaze angles) while the eyes are fixated on a target to navigate towards the target. The θ angles are used to compute the left/right turn angle θ_P shown in Fig. 3, and the ϕ angles are similarly used to compute an up/down turn angle ϕ_P. The fish's turn motor controllers are invoked to execute a left/right turn—right-turn-MC for an above-threshold positive θ_P and left-turn-MC for negative θ_P—with $|\theta_P|$ as parameter. Up/down turn motor commands are issued to the fish's pectoral fins, with an above-threshold positive ϕ_P interpreted as "up" and negative as "down". The motor controllers are explained in [29].

The remainder of the paper presents our new work on integrating color and disparity analysis within the animat vision system for dynamic obstacle avoidance.

3 Disparity and Color for Obstacle Avoidance

Color and stereo algorithms have been discussed extensively in the literature in a variety of passive vision systems, but rarely have they been integrated for use in dynamic obstacle avoidance systems. Color and stereo cues have recently been integrated together with motion cues to implement a real-time passive stereo system that can detect and identify moving objects for application to surveillance and human-computer interaction [1]. Disparity and color cues have also been combined to improve the focus of attention and recognition capabilities of an active vision system [12].

Recent work involving autonomous mobile robot systems have used single image cues for obstacle detection and avoidance such as stereo disparity [6], optical flow [5], visual looming [15], peripheral optical flow [8], divergence of image flow and time-to-contact [7], and appearance based models of color and shape [21]. Shigang *et. al.* [23] have recently proposed a method for autonomous robot navigation along routes described by landmarks based on range and color information.

The following sections describe our dynamic obstacle recognition and avoidance algorithms. Exploiting stereo and color cues, the algorithms enable the animat to navigate through its virtual environment fixating and tracking a reference target in the fovea while avoiding obstacles that appear in its low resolution visual periphery.

3.1 Stereo Analysis

Classical stereo analysis deals with the correspondence problem with two basic techniques; area-based methods [18, 13] and feature-based methods [19, 13]. Both types of stereo algorithms have computational problems. For example, in feature-based stereo algorithms the intensity data is first converted to a set of features assumed to be a more stable image property than raw intensities. The matching stage operates only on these extracted image features, consequently, producing sparse disparity maps. In order to obtain dense disparity maps, one is forced to interpolate missing values. Furthermore, false matches are basic to all feature-based stereo algorithms. These problems can be reduced by introducing additional constraints derived from reasonable assumptions about the physical properties of object surfaces and by increasing the number of features considered in the matching process. In area-based stereo algorithms intensity values within small image patches of the left and right views are compared and the correlation between these patches is maximized. To assure stable performance, area-based stereo algorithms need suitably chosen correlation measures and a sufficiently large patch size, which is a computationally expensive process. Other methods extract local Fourier phases of left and right images and the phase difference at each location is used to estimate disparity [22, 17, 9].

Several approaches take into consideration available biological and neurophysiological data about the human visual system [19, 22, 16]. There is biological evidence that the pattern of light projected on the human retina is sampled and spatially filtered. Very early in cortical visual processing, receptive fields become oriented and are well approximated by linear spatial filters, with impulse response functions that are similar to partial derivatives of a Gaussian function [30].

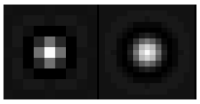
Our animat vision approach for estimating stereo disparity draws ideas from early visual processing in the primate cortex. We implement the receptive fields as steerable spatial filters that process the input images. The steerable filter responses at an image location form a *feature vector* that is used for solving the correspondence problem. The outputs of a steerable filter convolved with an image at multiple orientations provides very rich information about a local neighborhood around each pixel. Thus matching image patches from the left and right images of a stereo pair becomes simpler and the probability of a correct match increases as the length of the feature vector increases.

"Steerable filter" is a term used to describe a class of spatial filters in which a filter of arbitrary orientation is synthesized as a linear combination of a set of basis filters. Steerable filters, first developed by Freeman and Adelson [10], have been recently used for estimation of scene motion [14] and for object recognition [4] and stereopsis [16]. Simoncelli and Freeman have recently introduced a multiscale, multi-orientation steerable filter image decomposition framework called the Steerable Pyramid [24] which we use as a front-end for our stereo algorithm. It has the advantage of producing feature descriptions that are both translation- and rotation-invariant.

Our disparity estimation algorithm starts by decomposing the left and right images into steerable pyramid representations. The input images are initially low-pass filtered using a low-pass filter (L_0) with a radially symmetric frequency response. Each successive level of the pyramid is constructed from the previous level's low-pass band by subsampling it then convolving it with a bank of oriented basis filters (B_i) and a low-pass filter (L_1). Other orientations at each level are synthesized by taking linear combinations of the basis filtered images. The number of basis filters that are needed for steering the filter is $n+1$ for an $n_{th}-$order filter. We use third-order filters, thus requiring four basis filters oriented at $0°, 45°, 90°$, and $135°$ [10]. Fig. 4(a) shows these four spatial basis filters (B_i) which form a steerable basis set; any orientation of this filter can be written as a linear combination of the basis filters. Fig. 4(b) shows the two low-pass filters used to construct the pyramid. Typically, $L_0(w)$ is chosen to be $L_1(w/2)$ in the frequency domain [24]. Fig. 5 shows an example of a three-level steerable pyramid for a single orientation for an image acquired by the animat's right eye.

Feature vectors $\mathbf{f}_R(x, y, l)$ and $\mathbf{f}_L(x, y, l)$ are then constructed from the right and left pyramid responses for each pixel at each level of the pyramid by combining the responses of the multi-orientation steerable filters at each

a



b

Figure 4: (a) Four basis filters oriented at $0°, 45°, 90°$, and $135°$. (b) Left: L_1 low-pass filter, Right: L_0 low-pass filter.

pixel into a vector that provides a rich description of the intensities at that pixel in the image. To further enrich the description of each pixel, we make use of the (R, G, B) color signals from our color images, including them in the feature vector. This simple addition improves our matching process considerably by restricting the matching process to areas of similar color composition, which can be considered as a sort of color-feature constraint.

An initial disparity map is estimated at each individual level by matching left and right feature vectors by minimizing the mean square error (MSE) between left and right feature vectors. The MSE measure is computed over all the elements in the vector as follows;

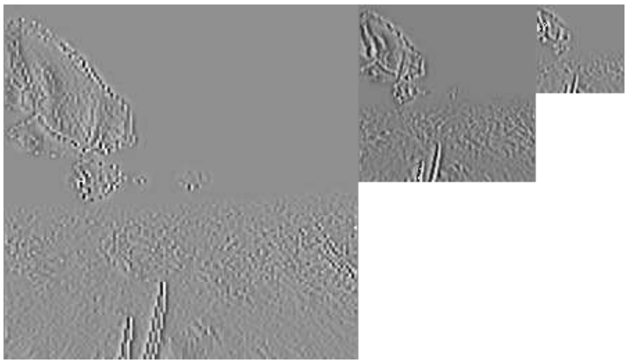
$$E_m = \frac{1}{S} \sum_{i \in S} [\mathbf{f}_R^i(x, y, l) - \mathbf{f}_L^i(x + d_x, y + d_y, l)]^2, \quad (1)$$

where S is the feature vector size. The MSE measure E_m is computed for a limited range of horizontal and vertical disparities $d_x(l) \in D_x(l)$ and $d_y(l) \in D_y(l)$ within a window of size $D_x(l) \times D_y(l)$ (typically, $D_x(0) = 20$, and $D_y(0) = 10$). The $(d_x(l), d_y(l))$ value that minimizes the MSE within this window is taken as the best initial disparity estimate at pixel (x, y, l) at pyramid level l. A boundary condition of zero disparity at image borders is applied. Also a zero disparity condition is applied to locations where no match is possible, such as across constant intensity areas. The disparity range used lies within $[-\frac{D(l)}{2}, \frac{D(l)}{2}]$. The disparity range differs from level to level and is given as,

$$D_x(l) = \frac{D_x(0)}{2^l}, \\ D_y(l) = \frac{D_y(0)}{2^l}. \quad (2)$$

A coarse-to-fine-flow-through strategy is then taken based on the assumption that for level l disparity estimates $|d(l)| > |\frac{D(l)}{4}|$ are more accurately estimated at the coarser level $l + 1$. Thus at coarse levels, large disparities are estimated presumably more accurately, and these flow through to the finer levels; small disparities that are estimated from the finer levels are assumed accurate since they cannot be estimated at coarser levels due to the loss of high frequency structure from the original coarse-level images.

(a)



(b)

Figure 5: (a) An image acquired by the animat's right eye, (b) A three-level Steerable Pyramid of the image in (a) shown for a single orientation.

Each disparity estimate $(d_x(l), d_y(l))$ at each level is median filtered at an appropriate scale before flow-through is performed (the window size used increases from coarse levels to fine levels – mainly 3×3 and 5×5 for 128×128 images). The full frame level is then median filtered to give the final disparity estimate. The median filtering step is required to correct for outlier disparity estimates that deviate from the correct expected estimate (a form of smoothness constraint on the estimates).

The stereo matching algorithm can be made more efficient by exploiting the epipolar geometry of the eyes of the artificial animal. The eye virtual cameras described in section 2.1 have identical focal lengths f_c. The eyes mounted in the animat's head may be aligned horizontally to within a scan line. To simplify the matching process we try to reduce the vertical disparity search range D_y as much as possible by restricting epipolar lines to one row. This is done by tying the vertical gaze angles together when acquiring stereo images; i.e., setting $\phi_R = \phi_L$. The vertical disparity search range D_y is nevertheless larger than one pixel due to non-uniform perspective distortions associated with the large field of view virtual cameras.

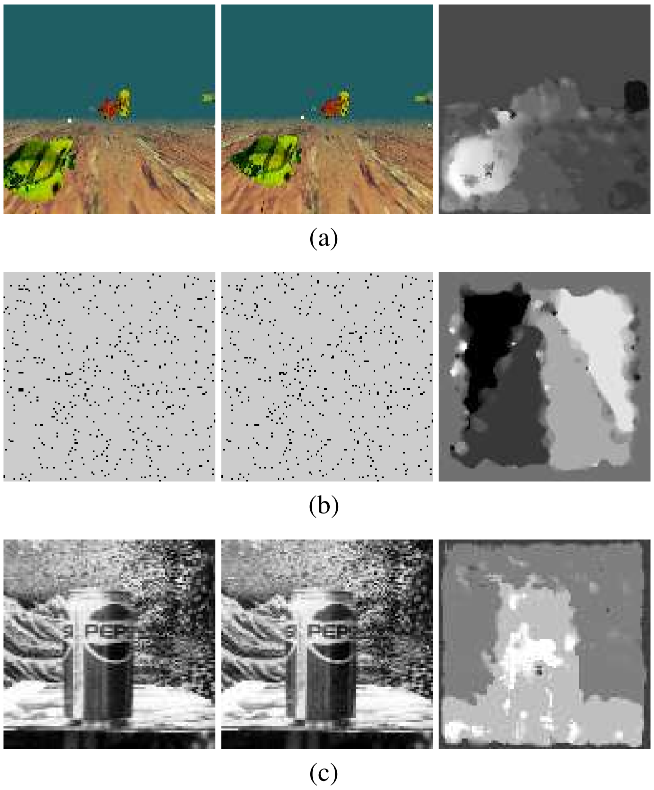


Figure 6: (a) Right and Left images acquired by the animat's eye and the estimated disparity map. (b) Stereo sparse random-dot-stereogram with 3% black dots and estimated disparity. (c) Pepsi sequence, left: frame 3, center: frame 0, right: estimated disparity.

Fig. 6 shows disparity maps estimated by the algorithm for real images, a random-dot stereogram, and retinal images acquired by our animat.

3.2 Color Obstacle Recognition and Localization

Next we develop a strategy to distinguish between dangerous obstacles and benign objects by combining the disparity cues estimated using the above algorithm with color cues available naturally from the acquired photorealistic images.

The animat continuously computes a disparity map from its stereo retinal input as it navigates through the virtual world. The estimated disparity map is used as a bottom-up cue to alert the animat of potential danger from objects that come too close. The disparity map is first segmented into potential obstacles. Then each segmented object is examined and matched against color mental models of designated dangerous obstacle objects. A match indicates that a candidate object is really an obstacle and is to be avoided, otherwise the object is considered harmless. Harmless objects include food particles and sea weeds.

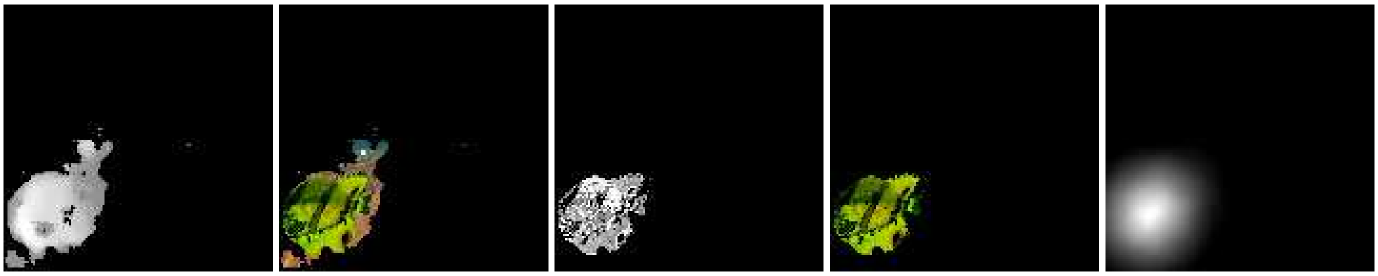
The disparity map $d(x, y)$ is segmented via thresholding. The appropriate disparity threshold is taken to be proportional to the disparity at the fixation point, which is the reference target that the animat is tracking foveally. This is given as, $dt = \mu d(0, 0)$; i.e., if the disparity of the localized object is at least μ times the estimated disparity of the reference target then this object is considered too close (typically $\mu = 1.0$). Values of $d(x, y) > dt$ are considered to belong to potential obstacles, while values of $d(x, y) < dt$ are set to the minimum disparity estimate. This segmentation step focuses the attention of the animat on potential obstacles while disregarding the rest of the peripheral view, thus simplifying the system as well as improving its robustness to false alarms.

The corresponding segmented pixels in the right eye image give the actual segmentation of the color objects. The color histogram of this segmentation is intersected with the color histogram of the mental models of stored obstacles, using the color methods described in [28]. A match indicates that this segment contains an obstacle; no match indicates a false alarm and the animat continues in its current path.

To localize a detected obstacle accurately, the exact region of support of this obstacle must be properly segmented out from the original segmentation obtained above. To tackle this non-trivial problem, we make use of Swain's color histogram backprojection methods [25]. Briefly, histogram backprojection gives large weights to pixel locations in the image whose color histogram closely resembles the color histogram of the model. This suggests, that we can use the backprojection itself to get an exact segment of the detected obstacle; pixel weights in the backprojection that are greater than an appropriate threshold are considered to belong to the obstacle. The threshold is determined empirically and for our case we used a value of 0.5 to separate the obstacle from outliers. Once the color region of support of the obstacle has been determined, the corresponding region in the disparity map gives the estimated disparities of the obstacle over the region. The updated disparity map is convolved with a circular disc of area equal to the area of the segmented obstacle's region of support. This will blur out any misclassified pixels in the segmentation while emphasizing the obstacle and facilitating its localization. The pixel location (x_c, y_c) of the peak in the blurred disparity map localizes the obstacle. Fig. 7 shows images of the various segmentation steps.

3.3 Obstacle Avoidance Strategy

The point of localization (x_c, y_c) obtained from the peak in the blurred disparity map is used to compute the steering angles the animat must use to steer clear of the obstacle. The angular location with respect to the right eye is



(a) (b) (c) (d) (e)

Figure 7: (a) Thresholded disparity map of fig. 6-(a). (b) Corresponding color segmentation of potential obstacles. (c) Backprojection map. (d) The exact region of support of the segmented obstacle. (e) The localization of the obstacle by blurring the corresponding segmentation of the disparity map.

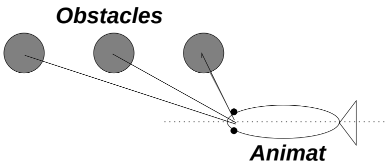


Figure 8: Relationship between close objects and large steering angles.

given as

$$\theta = \tan^{-1}(\frac{x_c}{f_c}),$$
$$\phi = \tan^{-1}(\frac{y_c}{f_c}). \quad (3)$$

The turn angles given to the animat's motor controller are, thus, proportional to $(-\theta, -\phi)$; i.e., in the opposite direction, to avoid collision while still fixating on a reference target to stabilize the visual world.

The merit of using $(-\theta, -\phi)$ for steering the animat is twofold: 1) simplicity of computing a steering vector and, 2) the fact that for close objects (θ, ϕ) is large, as is depicted in Fig. 8. Therefore, the turn maneuver will be large to avoid the obstacle quickly. The farther away the obstacle, the smaller the turn angles, hence steering will not be excessive.

Figure 9 shows frames from a top view of a sequence showing the animat navigating in its environment. The animat is fixating and tracking a target red fish while avoiding obstacles in the form of other fish obstructing its path. The figure shows two instances where the animat encounters an obstacle (frames 156 and 180). These are followed by frames showing how the animat has successfully avoided the obstacle by steering its body in the opposite direction as explained above. The white lines emanating from the eyes of the observer indicate the gaze direction.

4 Summary and Conclusions

We have presented computer vision research carried out within the animat vision paradigm, which employs lifelike artificial fishes inhabiting a physics-based, virtual marine world. We have successfully implemented a set of active vision algorithms for artificial fishes that integrate stereo and color analysis. These algorithms support robust vision-guided navigation and obstacle recognition and avoidance abilities, enabling the animat to better understand and interact with its dynamic virtual environment. Our work should also be relevant to the design of active vision systems for physical robotics.

Acknowledgements

The research described herein was supported by grants from the Natural Sciences and Engineering Research Council of Canada.

References

[1] H. Arakawa and M. Etoh. An integration algorithm for stereo, motion and color in real-time applications. *IEICE Transactions on Information and Systems*, E78-D(12):1615 – 1620, December 1995.

[2] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):996–1005, 1988.

[3] D. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[4] D.H. Ballard and L.E. Wixson. Object recognition using steerable filters at multiple scales. In *Proc. IEEE Workshop on Qualitative Vision*, pages 2 – 10, Los Alamitos, CA, June 1993.

[5] M. Campani, A. Giachetti, and V. Torre. Optic flow and autonomous navigation. *Perception*, 24:253–267, 1995.

[6] Y.C. Cho and H.S. Cho. A stereo vision-based obstacle detecting method for mobile robot navigation. *Robotica*, 12(3):203 – 216, May - June 1994.

[7] D. Coombs, M. Herman, T. Hong, and M. Nashman. Real-time obstacle avoidance using central flow divergence and peripheral flow. In *Proc. Fifth Inter. Conf. Computer Vision*

156 166 180 190

Figure 9: An overhead view of the animat as it pursues a red reference fish while detecting and avoiding other fish obstacles.

(ICCV'95), pages 276 – 283, MIT, Cambridge, MA, June 20 - 23 1995.

[8] D. Coombs and K. Roberts. Bee-bot: Using peripheral optical flow to avoid obstacles. *Proc. SPIE Intelligent Robots and Computer Vision XI*, 1825:714 – 721, 1992.

[9] D. Fleet, A. Jepson, and M. Jenkin. Phase-based disparity measurement. *CVGIP: Image Understanding*, 53:198 – 210, 1991.

[10] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE Trans. PAMI*, 13(9):891 – 906, September 1991.

[11] J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, 1979.

[12] W.E.L. Grimson, A.L. Ratan, G. Klanderman, and A. O'Donnell. An active visual attention system to play 'where's waldo'. In *Proc. of the Image Understanding Workshop*, volume 2, pages 1059 –1065, Monterey, CA, 13 - 16 Nov. 1994.

[13] B.K.P. Horn. *Robot Vision*. MIT Press, Cambridge, MA, 1986.

[14] C.L. Huang and Y.T. Chen. Motion estimation method using a 3D steerable filter. *Image and Vision Computing*, 13(1):21–32, 1995.

[15] K. Joarder and D. Raviv. A new method to calculate looming for autonomous obstacle avoidance. In *Proc. Computer Vision and Pattern Recognition Conf. (CVPR'94)*, pages 777 – 780, 1994.

[16] D.G. Jones and J. Malik. A computational framework for determining stereo correspondence from a set of linear spatial filters. In *Proc. Euro. Conf. Computer Vision (ECCV'92)*, pages 395 – 410, Portofino, Italy, 1992.

[17] K. Langley, T.J. Atherton, R.G. Wilson, and M.H.E. Larcombe. Vertical and horizontal disparities from phase. In *Proc. Euro. Conf. Computer Vision (ECCV'90)*, pages 315 – 325, 1990.

[18] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proc. Image Understanding Workshop*, pages 121–130, 1981.

[19] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proc. R. Soc. Lond. B.*, 204:301 – 328, 1979.

[20] T.F. Rabie and D. Terzopoulos. Motion and color analysis for animat perception. In *Proc. Thirteenth National Conf. on Artificial Intelligence (AAAI'96)*, pages 1090–1097, Portland, Oregon, August 4-8 1996.

[21] G. Salgian and D.H. Ballard. Visual routines for autonomous driving. In *Proc. Sixth Inter. Conf. Computer Vision (ICCV'98)*, Bombay, India, January 1998.

[22] T. Sanger. Stereo disparity computation using gabor filters. *Biological Cybernetics*, 59:405 – 418, 1988.

[23] L. Shigang, A. Ochi, and S. Tsuji. Route description by landmarks. In *Proc. of the Intelligent Vehicles '95 Symposium*, pages 454 – 459, Detroit, MI, 25 - 26 September 1995.

[24] E.P. Simoncelli and W.T. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *IEEE Inter. Conf. on Image Processing*, pages 444 – 447, Washington, DC, October 23-26 1995.

[25] M. Swain and D. Ballard. Color indexing. *Inter. J. Computer Vision*, 7:11 – 32, 1991.

[26] M.J. Swain and M.A. Stricker. Promising directions in active vision. *Inter. J. Computer Vision*, 11(2):109 – 126, 1993.

[27] D. Terzopoulos. Modeling living systems for computer vision. In *Proc. Int. Joint Conf. Artificial Intelligence (IJCAI'95)*, pages 1003–1013, 1995.

[28] D. Terzopoulos and T.F. Rabie. Animat vision: Active vision in artificial animals. *Videre: Journal of Computer Vision Research*, 1(1):2–19, September 1997.

[29] D. Terzopoulos, X. Tu, and R. Grzeszczuk. Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world. *Artificial Life*, 1(4):327–351, 1994.

[30] R.A. Young. Simulation of human retinal function with the Gaussian derivative model. In *Proc. Computer Vision and Pattern Recognition Conf. (CVPR'86)*, pages 564 – 569, 1986.