Automatic Object Tracking for an Unmanned Underwater Vehicle using Real-Time Image Filtering and Correlation

Richard L. Marks * Stephen M. Rock †
Stanford University Aerospace Robotics Laboratory
Durand Building
Stanford, California 94305

Michael J. Lee ‡
Monterey Bay Aquarium Research Institute
160 Central Avenue
Pacific Grove, California 93950

Abstract The focus of this research was automatic visual tracking of realistic underwater objects. Powerful real-time filtering and correlation hardware was used to generate full-image range and velocity maps at frame rates. These maps were in turn processed to provide object centroid position and rate. The object tracking system was tested experimentally on an underwater robotic vehicle, and qualitative results were obtained. Using this tracking system, the vehicle successfully tracked several different objects automatically. The feasibility of tracking realistic underwater objects in real-time was proven.

I. INTRODUCTION

The ability to automatically track objects is necessary for certain unmanned underwater vehicle missions. Automatic object tracking is a component needed for a variety of military and scientific applications [1]. One particular application of automatic tracking is the remote observation of fish by marine scientists. Because tracking such creatures requires a high bandwidth sensor, real-time vision systems were explored as a means of providing tracking data. Automatic visual tracking of mobile underwater objects (namely fish) is the particular problem that this research sought to address.

The underwater environment poses several challenges for real-time vision systems. Discrimination of objects from video images is difficult due to lighting and color variations, visual noise due to marine snow, and the wide spectrum of visual attributes of marine life. Many of these creatures are able to move rapidly, so tracking requires position data to be updated at a high rate with as little latency as possible. Attempting to achieve rapid updates produces a problem prevalent in many real-time vision systems of having too much data with too little time to process it.

II. BACKGROUND

Several researchers have addressed the problem of underwater optical sensing. Researchers at the University of New Hampshire studied the optical errors due to light attenuation, scattering, and absorption caused by water [2]. The University of Hawaii studied the use of passive optical sensing for obtaining position, orientation, and motion information of an underwater vehicle [3]. One application they addressed that could use this information is near-bottom stationkeeping [4, 5]. By controlling a vehicle with respect to the ocean bottom directly, problems such as instrument drift can be eliminated. MIT is currently developing a vision-system with similar capability which tracks features of a special target fixed in the environment [6].

The French laboratory INRIA has addressed visual servoing and mobile object tracking [7]. Particularly relevant work dealt with visually tracking an object and positioning a robot with respect to it. A mobile pattern of bright dots was tracked in three dimensions using a camera mounted on the end of a six d.o.f. robot arm [8]. Similar capability was demonstrated experimentally underwater [9] in a MBARI/Stanford ARL experiment. An underwater robotic vehicle with a camera mounted on a pan and tilt mechanism was used to automatically track a moving laser dot.

*Doctoral Candidate, Department of Aeronautics and Astronautics, rlmarks@sun-valley.Stanford.Edu

†Associate Professor, Department of Aeronautics and Astronautics, rock@sun-valley.Stanford.Edu

‡Senior Research Engineer, Monterey Bay Aquarium Research Institute, lemi@hp850.Mbari.Org

Tracking more realistic moving objects underwater has additional challenges not addressed by tracking surrogate objects. Since the object is not necessarily "bright", a different method is necessary to differentiate the object from the background and/or other objects. This must be done quickly enough so that the object can be tracked with reasonable performance.

In order to accomplish this, state-of-the-art video processing is needed. To deal with arbitrary object differentiation and the underwater visual noise/variation, a specialized type of vision processing can be used. This processing performs digital filtering and correlation of video data to calculate stereo disparities and optical flow.

III. IMAGE FILTERING AND CORRELATION

The availability of powerful vision processing was crucial for accomplishing the automatic tracking. The PRISM3 system [10] was the enabling technology which made realistic object tracking an attainable goal. The following section presents a short background of the theory behind the system and details some of its capabilities. For a more thorough description of the PRISM3 system, see [11] and [12].

A. Optical Flow

The PRISM3 vision system performs a specialized type of optical flow measurement. Optical flow can be summarized as a measurement of the rate of change in location of areas of video data. For example, if from one camera frame to the next, a section of the video image moves from A to B, that area is said to have "flowed" at a rate dictated by the change in location divided by the frame time. See Figure 1.

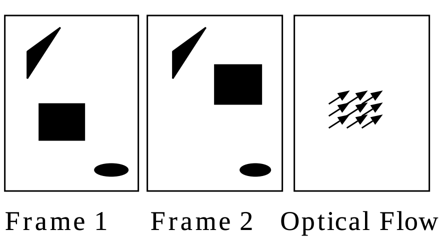

Figure 1: An example of localized optical flow.

B. Stereo Disparity

For a stereo pair of cameras, range can be determined by examining the change of location of video data from the left camera to the right. This measurement is known as the disparity. Objects far away have no disparity since they are at the same place on both images, whereas objects closer to the cameras appear at different locations and thus have a measurable disparity. See Figure 2. The similarities between the calculations required to compute optical flow and disparity enabled the same video processing equipment to be used for both.

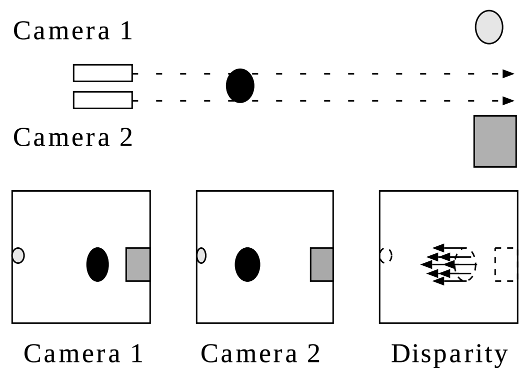

Figure 2: Example of stereo disparity.

C. Correlation

To calculate optical flow and disparity, areas of video images must be correlated. An area of one image is compared to many same-sized areas of another image. The correlation between two given areas is done by comparing every pixel in one area with the corresponding pixel in the other area. The number of pixels which have matching intensity are counted and used as a correlation measure. The location of the area in the second image which has largest correlation measure determines the optical flow or disparity. The disparity is the difference in the locations of the areas; the optical flow is the difference in the locations divided by the frame time. This process is repeated for a grid of areas over the first image to get measurements for the whole image.

D. Filtering

In the PRISM3 system, only pixels of matching intensity are counted in the correlation measure. This pixel correlation method is used because the video images used for correlation are first filtered to produce an image composed of only binary-valued pixels. The filtration can conceptually be broken up into three parts. The first part is a Gaussian filter. This filter smoothes the video image. The second part is a Laplacian filter which computes the concavity of the intensity of the image at each pixel location. The third part is a sign filter. Pixels where the concavity is positive are set to white, while the rest of the pixels are set to black.

Filtering the images before correlating has several benefits. Because the result is a binary image, the correlation step is much faster. Also, the filtering process reduces the impact that lighting variations and marine snow have on the correlation step. Because the width of the Gaussian filter can be adjusted, it is possible to remove frequencies of intensity variation above those needed to track the object.

E. Processing Capability

The PRISM3 processing system was purchased from the Teleos Research Corporation. With slight modification to the software, the PRISM3 can compute optical flow or disparity for 200 different 32-pixel by 32-pixel areas in 1/30 of a second. The location of the areas can be arbitrarily chosen anywhere on the screen. The width and height of the areas can vary between 8 and 127 pixels. The number, location, and size of the areas can be dynamically changed.

IV. OBJECT TRACKING

The following section describes the strategies and algorithms created to make use of the disparity and optical flow calculations provided by the PRISM3 system.

A. Area Size, Number, and Placement

1) Size and number: The total number of areas chosen for correlation is important, as is the size of each area. The basic tradeoff is speed vs. resolution. The total correlation time increases approximately linearly with both the number of areas and the individual area size. Area size also determines the accuracy of a correlation, the same way sample size does in a survey.

2) Placement: The placement of the areas is also important to the object tracking performance. Two different strategies are examined–dynamic placement and static placement.

For static area placement, the areas are initially arranged in a grid-like fashion over the entire image. This strategy is effective and simple to implement. However, it often wastes valuable time correlating uninteresting parts of the image unnecessary for object tracking.

Dynamic area placement is an alternative strategy that can also be used. Once an object is acquired (using a static grid over the whole image), an area grid of variable size is placed on the image. The size and origin of the grid is dynamically determined for each new image frame so that it fully covers the object and has one empty grid element on every side of the object. Assuming that the object cannot translate further than one grid element between frames, bordering the object on every side with an empty element insures that the grid will always fully cover the object.

Dynamic area placement requires fewer areas to track the object. Also, dynamic placement effectively screens out any visual interference away from the object. For example, consider the case of an object being tracked on the left side of the image. If another object enters the image on the right, it will not affect the tracking performance because the right part of the screen is not being processed.

Although dynamic placement offers several advantages, it is more complicated than static area placement. This complication increases the amount of code and lowers the number of measurements per unit time due to the added overhead of recalculating new area locations.

B. Disparity Map Filtering

The first object tracking algorithm is based solely upon disparity information from a static grid of areas. The assumptions made to enable successful tracking are that the object must span several grid elements, it must be significantly closer to the cameras than the background, and only one object can be in view. The object is tracked by distinguishing it from the background by its closer range.

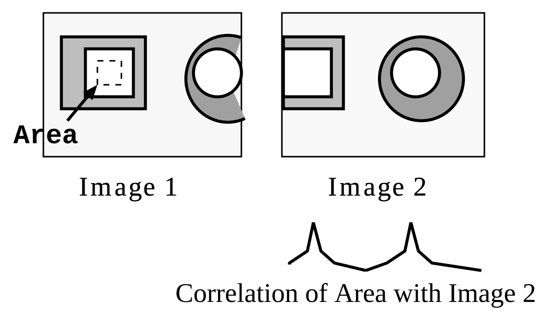

Figure 3: An example of a “false” correlation.

1) Neighbor filter: Unfortunately, areas in the image sometime correlate well although they actually correspond to different physical objects. An example of this occurs in Figure 3. The area from Image 1 is outlined by the dashed square. The correlation histogram is generated by successive correlations of this area with a row of areas from Image 2. The horizontal axis of the histogram corresponds to the abscissas of the origins of the areas from Image 2. Two correlation peaks occur in the histogram, each corresponding to different areas of Image 2. If the correlation peak corresponding to the area inside the circular region is slightly larger than the peak corresponding to the area inside the square region, it will be used for the disparity calculation. These “false” correlations occur frequently enough to be the major source of error in the disparity calculation.

To remove false correlation occurrences, the grid of disparity measurements can be filtered. If a measurement is surrounded by many similar measurements, it is likely that it is a valid measurement. If, however, a measurement is surrounded by many dissimilar measurements, it is probably spurious. More formally:

$d = disparity\ measurement\ of\ unknown\ validity$
$\phi(\mathbf{X}) = number\ of\ elements\ of\ set\ \mathbf{X}$
$\mathbf{S} = set\ of\ disparity\ meas.\ neighboring\ \mathbf{d}$
$\eta = \#\ of\ similar\ neighbors\ needed\ for\ validity$
$\delta = max\ disparity\ difference\ for\ similar\ meas.$
Define:

$$F(d) = \phi(\{c \mid c \in \mathbf{S}, |(d-c)| > \delta\} \tag{1}$$

Measurement d is valid if:

$$F(d) > \eta \tag{2}$$

Measurements which are considered invalid are set to a value of zero disparity (far away).

2) Centroid determination: To differentiate the object from the background, the object is assumed to be closer than the background. Therefore, a valid measurement $\mathbf{d}$ is only considered part of the object if $d \geq \gamma$, where γ is the minimum allowed disparity of the object. γ should be chosen to be slightly larger than the disparity of the background.

The object centroid coordinates are determined by separately averaging the ordinates and abscissas of the grid coordinates of each disparity measurement which corresponds to part of the object. Note that by averaging the coordinates of multiple measurements, the resolution of the grid location improves by a factor proportional to the number of measurements. Thus it is possible to achieve sub-grid accuracy for the centroid location.

3) Range determination: The overall disparity of the object can also be determined by averaging the measurements which correspond to part of the object. The overall disparity can then be converted to the object range. Assuming that the camera spacing is much smaller than the distance from the cameras to the object, the range can calculated using simple geometry:

$d = object\ disparity$
$l = camera\ spacing$
$k = pixel\ to\ radian\ conversion\ factor$
$r = object\ range$

$$r = \frac{l}{\tan(d * k)} \tag{3}$$

4) Background range: The background range can also be determined. Valid measurements with too small a disparity to be the object are assumed to be the background. The overall background disparity can be found by averaging these disparities. The range can again be calculated using (3), substituting the background disparity for the object disparity.

C. Optical Flow Map Filtering

The second object tracking algorithm is based solely upon velocity information from a grid of optical flow measurements. The assumptions are that the object spans several grid elements but takes up less than half the image, and only one moving object is in view. The object is tracked by distinguishing it from the background by its motion. If no motion relative to the background is detected, the object is assumed to have stayed in the same location as the previous frame. Note that this only enables azimuth and elevation tracking (no range), and the object must move to be acquired initially.

1) Neighbor filter: The optical flow measurements are susceptible to the same "false" correlations as the disparity measurements. A similar neighbor filter can be applied to the grid of velocity measurements as with the range measurements. Equations (1) and (2) can be used, but the magnitude of the velocity measurement is used instead of disparity to determine the validity of the measurements. Measurements which are considered invalid are specifically marked so they may be ignored by the background velocity and centroid determination algorithms.

2) Background velocity: In order to extract the optical flow measurements that correspond to the object, there must be some way to distinguish the object from the background. This is more difficult than in the disparity algorithm because the background velocity cannot be assumed to be small. However, if the object covers less than half the portion of the image being processed, it is likely that the most common optical flow measurements correspond to the background velocity.

3) Centroid determination: Optical flow measurements corresponding to the object are those which have a magnitude sufficiently different than the background velocity. Thus, the object centroid coordinates are determined by separately averaging the ordinates and abscissas of the grid coordinates of these valid measurements. If less than a threshold number of valid measurements have sufficiently different velocities than the background, the object is assumed to be moving at the background velocity, so the centroid is also assumed to be moving at the background velocity. In this case, the centroid is computed by adding the product of the background velocity and the time between frames to the old centroid location.

4) Object velocity: In addition to the location of the centroid of the object, the velocity of the object can be determined. By averaging the azimuth and elevation components of the optical flow measurements used for the centroid determination, the overall object velocity is computed. Note that this velocity may not be the same measurement as the velocity that would have been measured by differentiating the centroid location. The velocity measurement obtained by averaging the optical flow measurements is a more direct measurement of the object velocity which has better resolution and is less susceptible to noise. An accurate measurement of the object velocity can improve the tracking performance.

D. Object State Filter

Using either the optical flow or disparity measurements, information about the object state can be determined. Assuming there are limits to the movement capabilities of the object allows this information to be improved. If a maximum slew-rate for the object is assumed, the centroid location can be passed through a slew-rate saturation filter.

V. EXPERIMENTAL DEMONSTRATION

A stereo pair of black and white Pulnix CCD cameras were used to capture interlaced video frames at 30 Hz. A Datacube DIGICOLOR^{TM} board digitized the two video streams, and two special-purpose VME cards performed the image filtering and correlation (the hardware part of the PRISM3 system). A Motorola MVME 167 board running the VxWorks^{TM} real-time operating system processed the correlations and executed the tracking algorithms.

The cameras were mounted on the underwater test vehicle OTTER (Ocean Technology Testbed for Engineering Research). OTTER had three controlled degrees of freedom–x and y translation and azimuthal rotation. These were controlled through the use of four electric thrusters. An schematic of the OTTER vehicle is shown in Figure 4.

OTTER centered the object on the screen using simple proportional-derivative control on the error between desired and observed position of the object as computed by the vision system. When using the disparity map tracking algorithm, OTTER also remained a fixed distance away from the object.

A static area placement was used for the tracking experiments performed on OTTER. Both the disparity map and optical flow map filters were tested. The

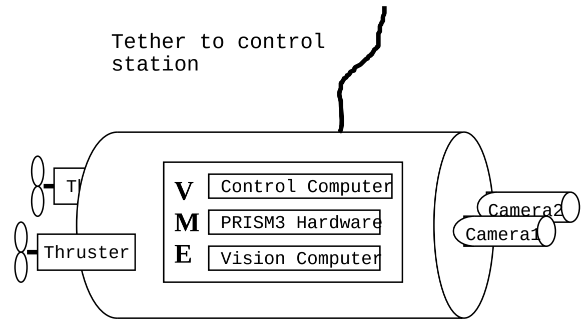

Figure 4: Schematic of the OTTER vehicle.

object centroid coordinates were passed through a slew-rate saturation filter. Several different objects were used to test the tracking system, the final object being a plastic replica of a turtle with a 20 cm diameter shell. Multiple slews were performed with each object. Because the tracking information was calculated in real time, it was possible to overlay cross-hairs of the vision system's estimated object centroid position on top of actual video taken by the video cameras. Video with overlaid object information was recorded for performance evaluation. In addition, time histories of the desired and observed positions of the object were available for determination of overall vehicle tracking performance.

VI. DISCUSSION OF RESULTS

A. Optical Flow Tracking

A variety of single objects were successfully tracked using a grid of optical flow measurements. However, the tracking performance was poor when the object velocity was close to the background velocity. After the background velocity was subtracted from grid of optical flow measurements, ideally the only nonzero velocities would be those corresponding to the object. This was not the case. Multiple measurements, not corresponding to the object, had nonzero velocities (due to noise and nonuniform velocities of the background). When the object velocity with respect to the background was large, the erroneous measurements slightly degraded the accuracy of the centroid calculation. The large number of centralized measurements corresponding to the object dominated the averaging calculation used to compute the centroid. However, when the object was moving sufficiently close to the background velocity, only the erroneous measurements were used to calculated the centroid. The centroid location calculated from these measurements tended to be random.

B. Disparity Tracking

The tracking performance achieved using a grid of disparity measurements was much better. The disparity tracking method was not susceptible to the same problem that occurred in the optical flow tracking method because the object disparity was always sufficiently different than the background disparity. Thus, the centroid calculations were made using only measurements corresponding to the object. These measurements tended to correspond to the interior of the object. The measurements along the perimeter of the object tended to be rejected by the neighbor filter. Measurements corresponding to appendages were also filtered out by the neighbor filter.

Object rotation affected the tracking performance due to the asymmetric nature of the objects which were tracked. As an object rotated, its viewable area varied. This caused the centroid calculation to move to different locations on the object, depending on its orientation. Thus, the centroid location of a rotating, non-translating object was not always fixed. Because of this, the vehicle continued

C. Vehicle Tracking

Despite the vision system's rapid object location update rate (up to 30 Hz), the vehicle was unable to track objects with large constant accelerations ($>\approx 1$ m/s^2). This was primarily due to the vehicle's limited control bandwidth. The velocity-squared drag inherent in the system bounded the performance that was achievable using a fixed-gain proportional-derivative controller. Although a more sophisticated vehicle controller could have been designed to improve the vehicle tracking performance, previous research has shown that high-speed, precise video tracking can be achieved best by mounting the cameras on a controlled pointing mechanism [9]. Future research will investigate the combination of the real-object tracking system with a high-speed pan and tilt apparatus.

VII. CONCLUSIONS

This research successfully demonstrated the feasibility of automatically tracking realistic underwater objects. A tracking system was designed which was able to track arbitrary objects in real time. Exploration has begun into how to combine optical flow measurements and disparity measurements to extend the tracking system capability. Research into multiple object tracking is also planned.

REFERENCES

[1] R. Blidberg, "Autonomous underwater vehicles. Current activities and research opportunities,"In *Rob. Aut. Sys.* Vol.7 No.2-3, pp. 139-150 (Aug. 1991).

[2] N. Kolagani, J.S. Fox, and D.R. Blidberg, "Photometric stereo using point light sources," In *Proc., IEEE Intl. Conf. Rob. Aut.*, Vol.2, pp. 1759-1764 (1992).

[3] S. Negahdaripour, A. Shokrollahi, and C.H. Yu, "Optical sensing for undersea robotic vehicles," In *Rob. Aut. Sys.,* Vol.7 No.2-3, pp. 151-163 (Aug. 1991).

[4] S. Negahdaripour, "Passive optical sensing for near-bottom stationkeeping," In *Proc. - Oceans '90*, pp. 82-87 (1990).

[5] J. Yuh, J.S. Fox, and R.Lakshmi, "Control and optical sensing in underwater robotic vehicles (URVs)," In *Proc. - Oceans '90*, pp. 88-93 (1990).

[6] H.L. Alexander, "Experiments in vision-based control of a neutrally buoyant free-flyer," In *Proc. SPIE,* Vol.1607, pp. 658-665, (1991).

[7] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," In *IEEE Trans. Rob. Aut., Vol.8 No.3*, pp. 313-326 (1992).

[8] F. Chaumette, P. Rives, and B. Espiau, "Positioning of a robot with respect to an object, tracking it and estimating its velocity by visual servoing," In *Proc., IEEE Intl. Conf. Rob. Aut.*, Vol.3, pp. 2248-2253 (1991).

[9] R. Marks, H. Wang, S. Rock, M. Lee, and R. Burton, "Combined camera and vehicle tracking of underwater objects," In *Proc. ROV '92*, pp. 325-332, (1992).

[10] H.K. Nishihara, "PRISM: a Practical Realtime Imaging Stereo Matcher," *Proc. SPIE*, pp. 134-142, (1983).

[11] H.K. Nishihara, "Practical realtime imaging stereo matcher," *Optical Engn.*, Vol. 23 No.5, pp. 536-545, Oct. 1984.

[12] H.K. Nishihara, "Minimal meaningful measurement tools," Technical Report TR-91-01, Teleos Research (1991).