Chapter 6

Detection and Tracking of Objects in Underwater Video

6.1 Introduction

In chapter 5 we demonstrated the application of the saliency-based attention system introduced in chapter 2 to solve learning and recognition of multiple objects in single scenes and objects in cluttered scenes. In this chapter we show how the same basic principles can be applied to the detection and tracking of objects in a multiple target tracking scenario. In particular, we are interested in detecting and tracking objects in underwater video used to estimate population statistics of marine animals. This task is challenging because of the low contrast and the large amount of clutter in the video. However, human annotators can learn this task in a matter of months. This led us to look at our model of selective visual attention in humans for insights into how we might be able to tackle this hard problem with biologically inspired algorithms.

This work is a collaboration with the Monterey Bay Aquarium Research Institute (MBARI). After extensive consultation with professional video annotators at MBARI, I designed and implemented the attention and tracking system, initially as part of the iLab Neuromorphic Vision C++ Toolkit. Karen Salamy set up the computer infrastructure at MBARI and conducted the evaluations in subsection 6.4.1. Danelle Cline set up processing on the Beowulf computer cluster at MBARI, and she conducted the evaluations in subsection 6.4.2. Rob Sherlock was the expert annotator for subsection 6.4.2. As the group manager at MBARI, Dr. Duane Edgington provided organizational support for the project throughout. He and I initiated the project during the 2002 Workshop for Neuromorphic Engineering in Telluride, Colorado, and the initial phase was supported by a collaborative research grant by the Institute for Neuromorphic Engineering.

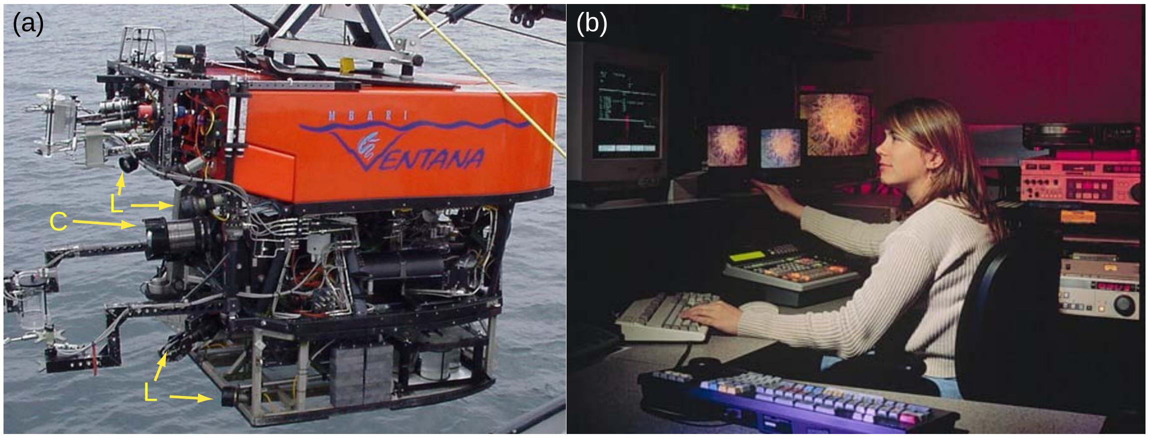

Figure 6.1: (a) ROV Ventana with camera (C) and lights (L). (b) Manual annotation of video tapes in the video lab on shore.

6.2 Motivation

Ocean-going remotely operated vehicles (ROVs, figure 6.1 a) increasingly replace the traditional tow net approach of assessing the kinds and numbers of animals in the oceanic water column (Clarke 2003). At MBARI, high-resolution video equipment on board the ROVs is used to obtain quantitative video transects (QVTs) through the ocean midwater, from 50 m to 1000 m depth. QVTs are superior to tow nets in assessing the spatial distribution of animals and in recording delicate gelatinous animals that are destroyed in nets. Unlike tow nets, which integrate data over the length of the tow, QVTs provide high-resolution data at the scale of the individual animals and their natural aggregation patterns for animals and other objects larger than about 2 cm in length (Robison 2000).

However, the current manual method of analyzing QVT video material is labor intensive and tedious. Highly trained scientists view the video tapes, annotate the animals, and enter the annotations into a data base (figure 6.1b). This method poses serious limitations to the volume of ROV data that can be analyzed, which in turn limits the length and depth increments of QVTs as well as the sampling frequency that are practical with ROVs (Edgington et al. 2003; Walther and Edgington 2004).

Being able to process large amounts of such video data automatically would lead to an order-of-magnitude shift in (i) lateral scale of QVTs from current 0.5 km to mesoscale levels (5 km); (ii) depth increment from current 100 m to the biologically significant 10 m scale; and (iii) sampling frequency from currently monthly to daily, which is the scale of dynamic biological processes. Such an increase in data resolution would enable modeling of the linkage between biological processes and physicochemical hydrography.

6.3 Algorithms

We have developed an automated system for detecting and tracking animals visible in ROV video (Walther et al. 2004a; Edgington et al. 2003). This task is difficult due to the low contrast of many of the marine animals, their sparseness in space and time, and debris ("marine snow") cluttering the scene, which shows up as ubiquitous high contrast clutter in the video.

Our system consists of a number of sub-components whose interactions are outlined in figure 6.2. The first step for all video frames is the removal of background. Next, the first frame and every pth frame thereafter (typically, $p = 5$) are processed with an attentional selection algorithm to detect salient objects. Detected objects that do not coincide with already tracked objects are used to initiate new tracks. Objects are tracked over subsequent frames, and their occurrence is verified in the proximity of the predicted location. Finally, detected objects are marked in the video frames.

6.3.1 Background Subtraction

Images captured from the video stream often contain artifacts such as lens glare, parts of the camera housing, parts of the ROV, or instrumentation (figure 6.3). Also, non-uniform lighting conditions cause luminance gradients that can be confusing. All of these effects share the characteristic that they are constant over medium or long periods of time, unlike the apparently fast moving objects in the water. Hence we can remove them by background subtraction (x, y, and t are assumed to be discrete):

$$I'(x,y,t) = \left[I(x,y,t) - \frac{1}{\Delta t_b} \sum_{t'=(t-\Delta t_b)}^{t-1} I(x,y,t') \right]_+ , \tag{6.1}$$

where I is the image intensity before and I' after background subtraction; $[\cdot]_+$ denotes rectification, i.e., setting all negative values to zero. This process is repeated separately for the R, G, and B channels of the color images.

The value of Δt_b should be larger than the typical dwell time of objects in the same position in the camera plane and shorter than the timescale of changes in the artifacts. In our transect videos, objects typically move fast. We found that $\Delta t_b = 0.33$ s (10 frames) works quite well, giving us enough flexibility to adjust to changes in the artifacts quickly (figure 6.4b).

6.3.2 Detection

We use the model of saliency-based bottom-up attention described in chapter 2 for the detection of new objects. Following background subtraction, input frames are decomposed into seven channels (intensity contrast, red/green, and blue/yellow double color opponencies, and the four canonical spatial orientations) at six spatial scales, yielding 42 "feature maps." After iterative spatial compe-

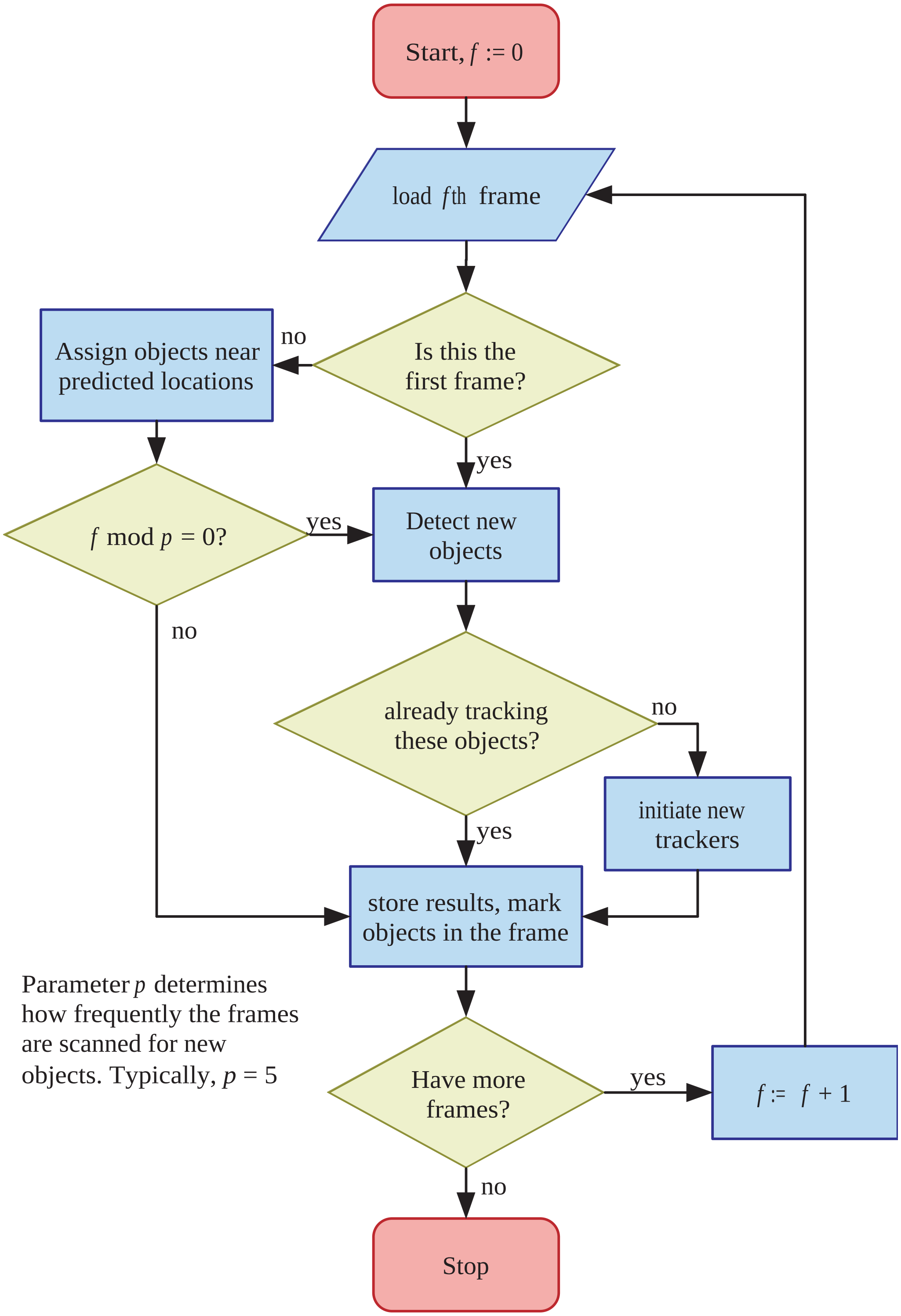
Start, f := 0
load fth frame
Is this the first frame?
no
Assign objects near predicted locations
yes
f mod p = 0?
yes
Detect new objects
no
already tracking these objects?
no
initiate new trackers
yes
store results, mark objects in the frame
Parameter p determines how frequently the frames are scanned for new objects. Typically, p = 5
Have more frames?
yes
f := f + 1
no
Stop

Figure 6.2: Interactions between the various modules of our system for detecting and tracking marine animals in underwater video.

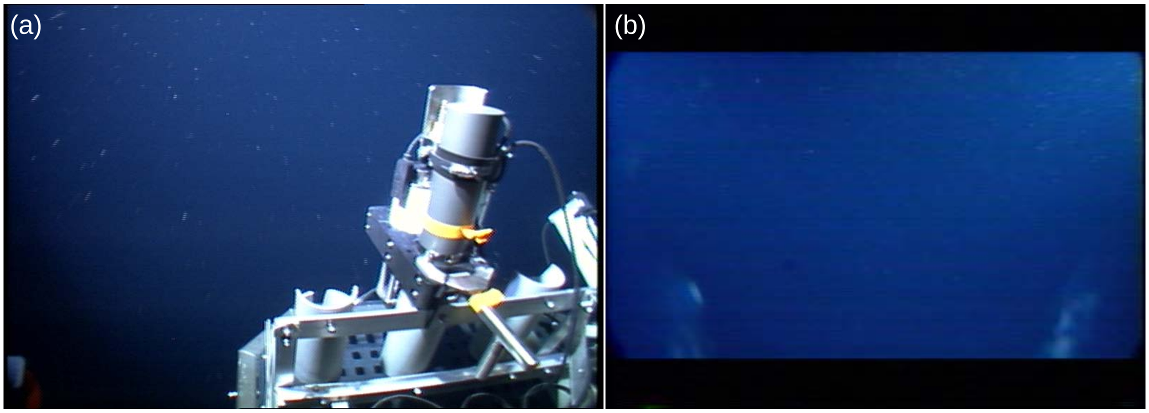

Figure 6.3: Example frames with (a) equipment in the field of view; (b) lens glare and parts of the camera housing obstructing the view.

tition for salience within each map, only a sparse number of locations remain active, and all maps are combined into a unique “saliency map” (figure 6.4c). The saliency map is scanned by the focus of attention in order of decreasing saliency through the interaction between a winner-take-all neural network (which selects the most salient location at any given time) and an inhibition-of-return mechanism (transiently suppressing the currently attended location from the saliency map). Objects are segmented at the salient locations found this way, and their centroids are used to initiate tracking (see subsection 6.3.3).

We found that oriented edges are the most important feature for detecting marine animals. In many cases, animals that are marked by human annotators have low contrast but are conspicuous due to their clearly elongated edges (figure 6.5a), whereas marine snow has higher contrast but lacks a prevalent orientation. In order to improve the performance of the attention system in detecting such faint yet clearly oriented edges, we use a normalization scheme for the orientation filters that is inspired by the lateral inhibition patterns of orientation-tuned neurons in visual cortex.

We compute oriented filter responses in a pyramid using steerable filters (Simoncelli and Freeman 1995; Manduchi et al. 1998) at four orientations. High-contrast “marine snow” particles that lack a preferred orientation often elicit a stronger filter response than faint string-like animals with a clear preferred orientation (figure 6.5c). To overcome this problem, we normalize the response of each of the oriented filters with the average of all of them:

$$O'_i(x,y) = \left[O_i(x,y) - \frac{1}{N} \sum_{j=1}^{N} O_j(x,y) \right]_+ , \tag{6.2}$$

where $O_i(x,y)$ denotes the response of the ith orientation filter ($1 \leq i \leq N$) at position (x,y), and $O'_i(x,y)$ is the normalized filter response (here, $N = 4$). Figure 6.6 shows a possible implementation of this kind of normalization in neurons, using an inhibitory interneuron, which represents the sum

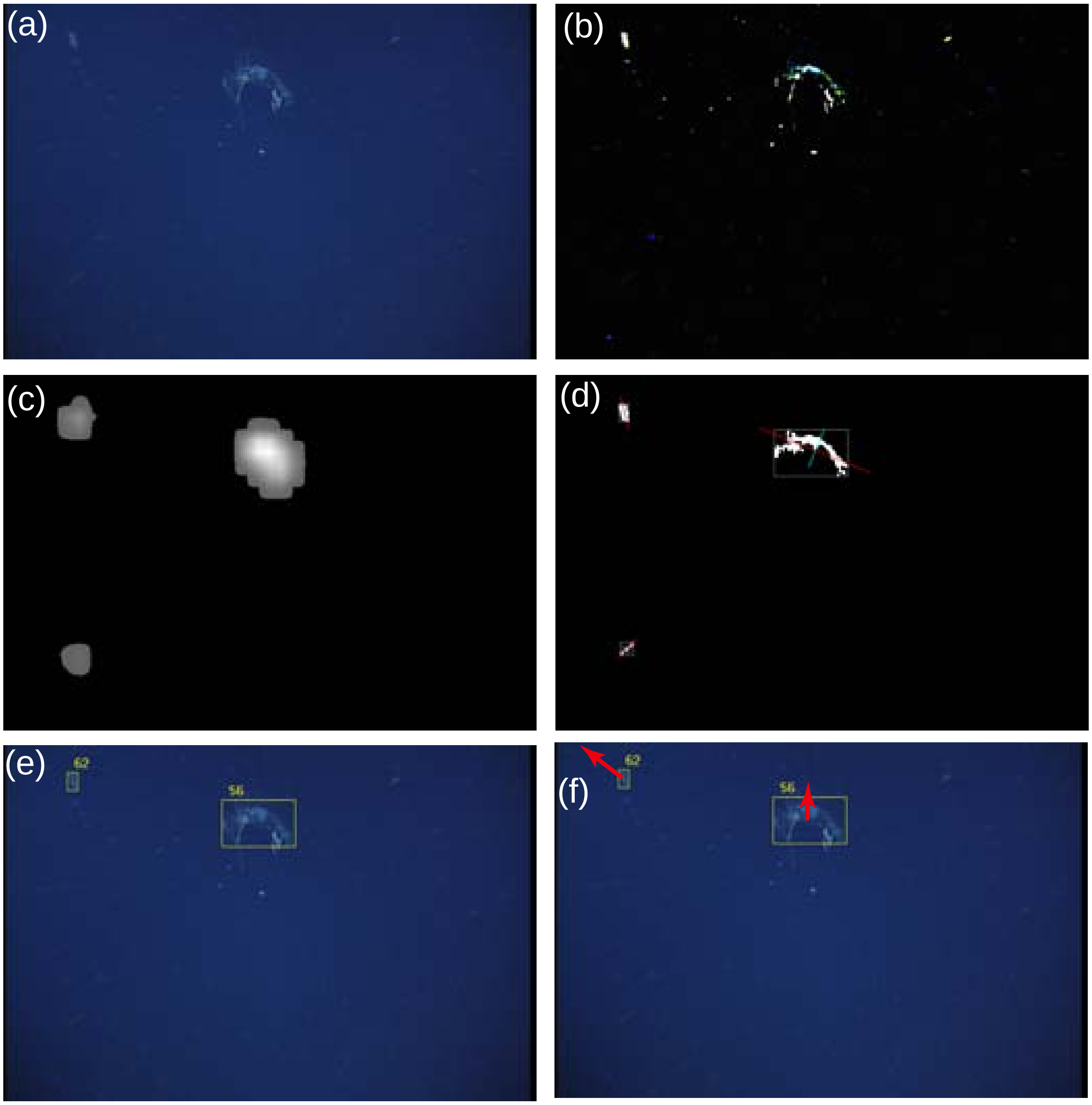

Figure 6.4: Processing steps for detecting objects in video frames. (a) original frame (720×480 pixels, 24 bits color depth); (b) after background subtraction according to eq. 6.1 (contrast enhanced for displaying purpose); (c) saliency map for the preprocessed frame (b); (d) detected objects with bounding box and major and minor axes marked; (e) the detected objects marked in the original frame and assigned to tracks; (f) direction of motion of the object obtained from eq. 6.11.

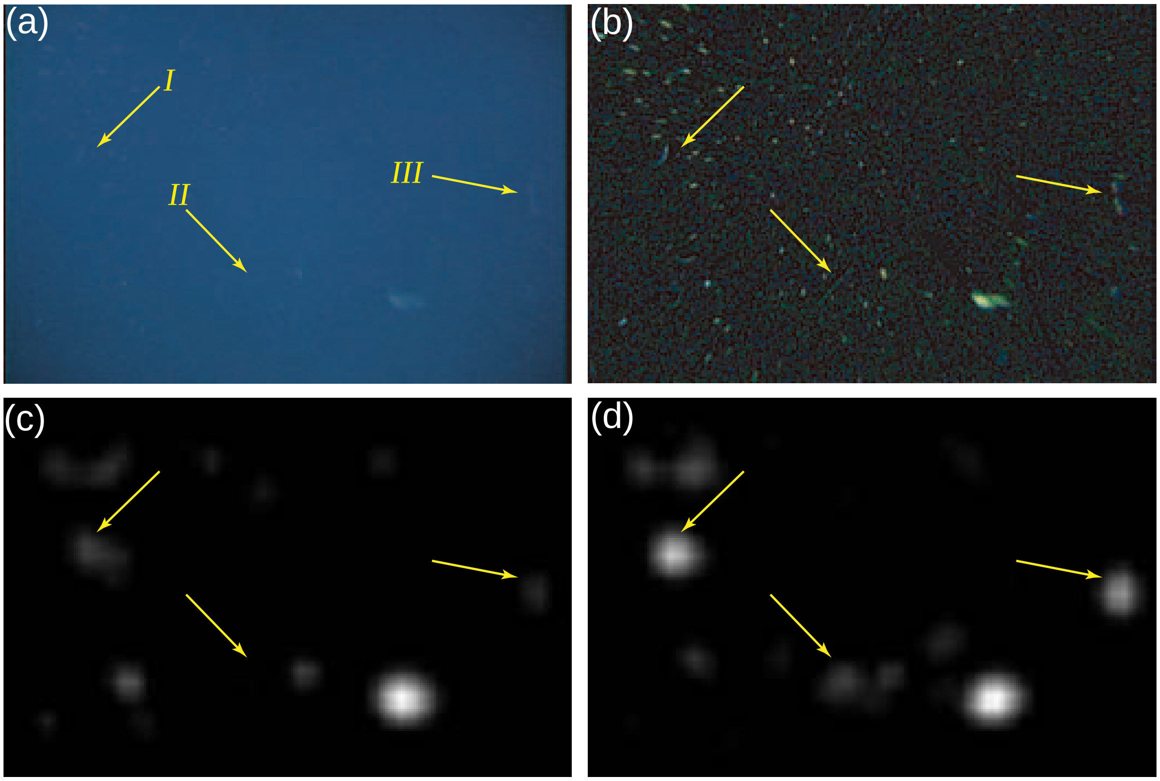

Figure 6.5: Example for the detection of faint elongated objects using across-orientation normalization. (a) original video frame with three faint, elongated objects marked; (b) the frame after background subtraction according to eq. 6.1 (contrast enhanced for illustration); (c) the orientation conspicuity map (sum of all orientation feature maps) without across-orientation normalization; (d) the same map with across-orientation normalization. Objects I and III have a very weak representation in the map *without* normalization (c), and object II is not represented at all. The activation to the right of the arrow tip belonging to object II in (c) is due to a marine snow particle next to the object and is not due to object II. Compare with (d), where object II as well as the marine snow particle create activation. In the map *with* normalization (d), all three objects have a representation that is sufficient for detection.

on the right-hand side of eq. 6.2.

Although this simple model of across-orientation normalization falls short of modeling the full array of long and short range interactions found in primary visual cortex (DeAngelis et al. 1992; Lee et al. 1999; Peters et al. 2005), normalization leads to a clear improvement in detecting faint elongated objects (figure 6.5 d).

6.3.3 Tracking

Once objects are detected, we extract their outline and track their centroids across the image plane using separate linear Kalman filters to estimate their x and y coordinates.

During QVTs the ROV is driven through the water column at a constant speed. While there are some animals that propel themselves at a speed that is comparable to or faster than the speed

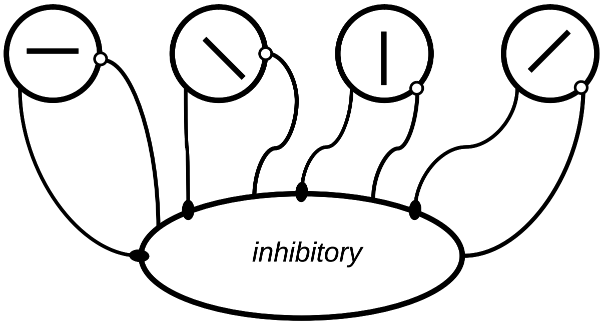

Figure 6.6: A schematic for a neural implementation of across-orientation normalization using an inhibitory interneuron. This circuit would have to be implemented at each image location for this normalization to function over the entire visual field.

of the ROV, most objects either float in the water passively or move on a much slower time scale than the ROV. Hence, we can approximate that the camera is moving at a constant speed through a group of stationary objects.

Figure 6.7 illustrates the geometry of the problem in the reference frame of the camera. In this reference frame, the object is moving at a constant speed in the x and z directions. The x coordinate of the projection onto the camera plane is

$$x'(t) = \frac{x(t) \cdot z_c}{z(t)} = \frac{v_x z_c \cdot t + c_x z_c}{v_z \cdot t + c_z}. \tag{6.3}$$

To a second order approximation, the dynamics of this system can be described by a model that assumes constant acceleration:

$$\frac{d\underline{x}}{dt} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \underline{x} \tag{6.4}$$

with

$$\underline{x} = \begin{pmatrix} x' \\ v \\ a \end{pmatrix}, \tag{6.5}$$

which results in a fundamental matrix that relates $\underline{x}(t)$ to $\underline{x}(t+\tau)$:

$$\Phi(\tau) = \begin{bmatrix} 1 & \tau & \frac{\tau^2}{2} \\ 0 & 1 & \tau \\ 0 & 0 & 1 \end{bmatrix}. \tag{6.6}$$

$\Phi(\tau)$ is used to define a linear Kalman filter (Kalman and Bucy 1961; Zarchan and Musoff 2000). The deviations of this simplified dynamics from the actual dynamics are interpreted as process noise

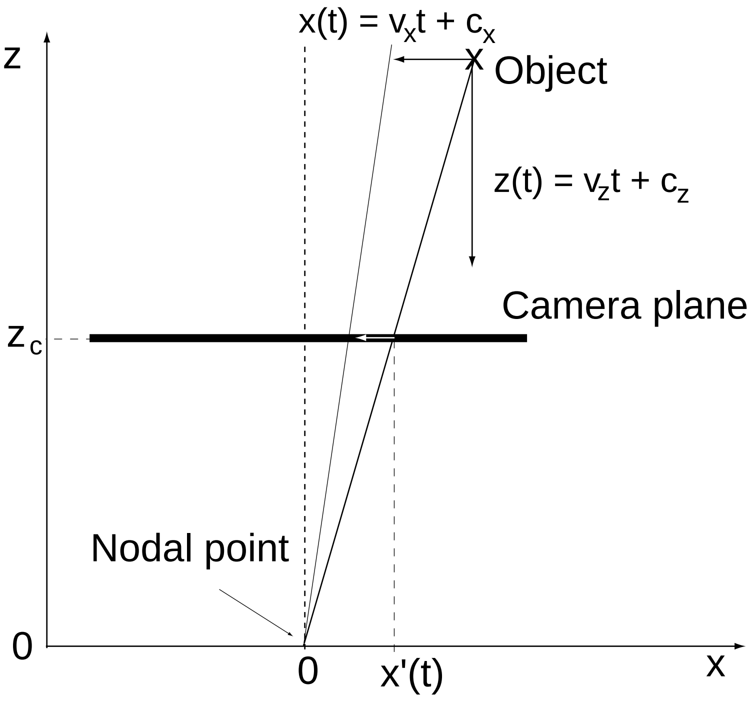

Figure 6.7: Geometry of the projection problem in the camera reference frame. The nodal point of the camera is at the origin, and the camera plane is at z_c. The object appears to be moving at a constant speed into the x and z direction as the camera moves toward the object. Eq. 6.3 describes how the projection of the object onto the camera plane moves in time.

Q. The resulting iterative equations for the Kalman filter are:

$$M_k = \Phi_k P_{k-1} \Phi_k^T + Q, \tag{6.7}$$

$$K_k = M_k H^T \left(H M_k H^T + R \right)^{-1}, \text{ and} \tag{6.8}$$

$$P_k = \left(I - K_k H \right) M_k, \tag{6.9}$$

where P_0 is initialized to a diagonal matrix with large values on the diagonal, $R = [\sigma_m^2]$ is the variance of the measurement noise, $H = [1\ 0\ 0]$ is the measurement matrix relating the measurement x^M to the state vector $\underline{x}$. Q is the process noise matrix:

$$Q(\tau) = \sigma_p^2 \begin{bmatrix} \frac{1}{20}\tau^5 & \frac{1}{8}\tau^4 & \frac{1}{6}\tau^3 \\ \frac{1}{8}\tau^4 & \frac{1}{3}\tau^3 & \frac{1}{2}\tau^2 \\ \frac{1}{6}\tau^3 & \frac{1}{2}\tau^2 & \tau \end{bmatrix}, \tag{6.10}$$

where σ_p^2 is the variance of the process noise. For our particular tracking problem we found $\sigma_m^2 = 0.01$ and $\sigma_p^2 = 10$ to be convenient values.

Once the Kalman matrix K_k is obtained from eq. 6.8, an estimation for $\underline{x}_k$ can be computed from the previous state estimate $\underline{\hat{x}}_{k-1}$ and the measurement x_k^M:

$$\underline{\hat{x}}_k = \Phi_k \underline{\hat{x}}_{k-1} + K_k(x_k^M - H\Phi_k \underline{\hat{x}}_{k-1}). \tag{6.11}$$

When no measurement is available, a prediction for $\underline{x}_k$ can be obtained by extrapolating from the previous estimate:

$$\underline{\hat{x}}_k = \Phi_k \underline{\hat{x}}_{k-1}. \tag{6.12}$$

For the initiation of the tracker we set $\underline{\hat{x}}_0 = [x_0^M \; 0 \; 0]^T$, where x_0^M is the coordinate of the object's centroid obtained from the saliency-based detection system described in the previous section.

We employ the same mechanism to track the y coordinate of the object in the camera plane. Whenever the x or y coordinate tracker runs out of the camera frame, we consider the track finished and the corresponding object lost. Re-entry of objects into the camera frame almost never occurs in our application. We require that an object is successfully tracked over at least five frames, otherwise we discard the measurements as noise.

In general, we are tracking multiple objects all at once. Normally, multi-target tracking raises the problem of assigning measurements to the correct tracks (Kirubarajan et al. 2001). Since the attention algorithm only selects the most salient objects, however, we obtain a sparse number of objects whose predicted locations are usually separated far enough to avoid ambiguities. If ambiguities occur, we resolve them using a measure that takes into account the Euclidean distance of the detected objects from the predictions of the trackers and the size ratio of the detected and tracked objects.

6.3.4 Implementation

We use two ROVs for deep sea exploration, the ROV Ventana and the ROV Tiburon (Newman and Stakes 1994; Mellinger et al. 1994). ROV Ventana (figure 6.1a), launched from R/V Point Lobos, uses a Sony HDC-750 HDTV (1035i30, 1920x1035 pixels) camera for video data acquisition, and the data are recorded on a DVW-A500 Digital BetaCam video tape recorder (VTR) on board the R/V Point Lobos. ROV Tiburon operates from R/V Western Flyer; it uses a Panasonic WVE550 3-chip CCD (625i50, 752x582 pixels) camera, and video is also recorded on a DVW-A500 Digital BetaCam VTR. On shore, a Matrox RT.X10 and a Pinnacle Targa 3000 Serial Digital Interface video editing card in a Pentium P4 1.7 GHz personal computer (PC) running the Windows 2000 operating system and Adobe Premier are used to capture the video as AVI or QuickTime movie files at a resolution of 720 x 480 pixels and 30 frames per second. The frames are then converted to Netpbm color images and processed with our custom software.

All software development is done in C++ under Linux. To be able to cope with the large amount

Table 6.1: Single frame analysis results.

	Image set 1	Image set 2
Date of the dive	06/10/2002	06/18/2002
ROV used for the dive	*Tiburon*	*Ventana*
Number of images obtained	456	1004
Images without animals	205	673
Images with detected animals	224	291
Images with missed animals	27	40

of video data that needs to be processed in a reasonable amount of time, we deployed a computer cluster with 8 Rack Saver rs1100 dual Xeon 2.4 GHz servers, configured as a 16 CPU, 1 Gigabit per second Ethernet Beowulf cluster. We currently process approximately three frames per second on each of the Xeon nodes at a resolution of 720 × 480 pixels.

6.4 Results

We present two groups of results – an assessment of the attentional selection algorithm for our purpose and a comparison of the automatic processing of three 10 minute video clips with expert annotations.

6.4.1 Single Frame Results

In order to assess the suitability of the saliency-based detection of animals in video frames in the early stage of our project, we analyzed a number of single video frames. We captured the images from a typical video stream at random. We analyzed two image sets – one with 456 images from video recorded by ROV Tiburon on June 10, 2002 and one with 1004 images from video recorded by ROV Ventana on June 18, 2002. Only some of the images in the sets contain animals. We used the attentional detection system described in subsection 6.3.2 to evaluate its performance on these images. We counted the number of images in which the most salient location, i.e., the location first attended to by the algorithm, coincides with an animal. The results are displayed in table 6.1.

In the images that did not contain animals the saliency mechanism identified other visual features as being the most salient ones, usually particles of marine snow. Originally, the system had no ability to identify frames that did not contain any objects of interest. We introduced this concept when we implemented tracking of objects (see subsection 6.3.3). For the majority of the images that did contain animals, the saliency program identified the animal (or one of the animals if more than one were present) as the most salient location in the image. In image set 1 the animals were identified as the most salient objects in 89% of all images that contained animals. In image set 2 this was the case for 88% of the images with animals. Ground truth was established by individual inspection of

Table 6.2: Results from processing four quantitative video transects.

Video Clip (10 min duration each)	**A**	**B**	**C**	**D**
Dive depth	400 m	50 m	1000 m	900 m
Number of animals annotated by person	122	57	29	29
Of those, found by the software	102 (84 %)	40 (70 %)	25 (86 %)	21 (72 %)
Missed by the software	20 (16 %)	17 (30 %)	4 (14 %)	8 (28 %)
Found by the software but missed by the person	5	2	3	4

the test images.

6.4.2 Video Processing

As a test of the video processing capabilities of our system, we processed four 10 min video segments that had previously been annotated by scientists. Table 6.2 shows the results. Our system missed on average 21 % of the annotated objects in the video clips. In several cases, our automated detection system detected animals that the annotators had missed. The program also detected several other objects that the scientists had not annotated. However, we did not consider these cases as false positives because a classification system capable of recognizing species would also be able to distinguish target animals from salient debris. The big advantage of the attention and tracking system is a massive reduction of data by 95–98 % for subsequent processing by a classification system.

Almost all misses by the software are of the sub-class Siphonophora, which are long string-like colonial animals that yield very low contrast in the video frames. Interestingly, most of the animals that were initially missed by the scientist were of the same sub-class.

Since our system can detect and track objects in the video fairly consistently, it can be used to summarize longer video segments by displaying a collection of small thumbnail video clips that show the most interesting parts of the original video. The individual frames for these thumbnail clips can be extracted automatically from the segmentation and tracking results. This mode is best described as a virtual camera following individual objects in the video. Using such automatic indexing, scientists would be able to grasp the most interesting parts of a particular video quickly without having to review the entire tape. Once these miniature clips are made part of the existing video annotation reference system (VARS 2005), they will help users identify potentially relevant tapes in the existing large collection of video data.

6.5 Discussion

We have presented a new method for processing video streams from ROVs automatically. This technology has a potentially significant impact on the daily work of video annotators by aiding their analysis of noteworthy events in video. After continued refinement we hope that the software will be able to perform a number of routine tasks fully automatically, such as “outlining” video, analyzing QVT video for the abundance of certain easily identifiable animals, and marking especially interesting episodes in the video that require the attention of the expert annotators.

Beyond its applications to ROV video, our method for automated underwater video analysis may potentially have a larger impact by enabling Autonomous Underwater Vehicles (AUVs) to collect and analyze quantitative video transects, with the potential to sample more frequently and at an ecologically significant finer spatial resolution and greater spatial range than is practical and economical for ROVs. We also see great benefit in automating portions of the analysis of video from fixed observatory cameras, where autonomous response to potential events (e.g., pan and zoom to events) and automated processing for science users of potentially very sparse video streams from hundreds of network cameras could be key to those cameras being practical scientific instruments.

In chapter 5 we showed that our attentional selection model can successfully cue object recognition systems for learning and recognition of multiple objects. In this chapter we have demonstrated the use of such an algorithm for multi-target tracking. In particular, the selective attention system detects the targets for track initiation completely automatically. Saliency-based filtering of targets even before resources are spent on tracking them saves computational resources, and it drastically reduces the complexity of the assignment problem in multi-target tracking.