Improved Estimation of Target Velocity Using Multiple Model Estimation and a Dynamic Bayesian Network for a Robotic Tracker of Ocean Animals

Aaron Plotnik[1] and Stephen Rock[2]

[1] Department of Aeronautics and Astronautics
Stanford University
aplotnik@stanford.edu
http://arl.stanford.edu/~aplotnik

[2] Department of Aeronautics and Astronautics
Stanford University
and
Monterey Bay Aquarium Research Institute
Moss Landing, CA
rock@stanford.edu
http://arl.stanford.edu

Abstract. A vision-based automatic tracking system for ocean animals in the midwater has been demonstrated in Monterey Bay, CA. Currently, the input to this system is a measurement of relative position of a target with respect to the tracking vehicle, from which relative velocities are estimated by differentiation. In this paper, the estimation of target velocities is extended to use knowledge of the modal nature of the motions of the tracked target and to incorporate the discrete output of an online classifier that categorizes the visually observable body motions of the animal. First, by using a multiple model estimator, a more expressive hybrid dynamical model is imposed on the target. Then, the estimator is augmented to input the discrete classification from the secondary vision algorithm by recasting the process and sensor models as a dynamic Bayesian network (DBN). By leveraging the information in the body motion classifications, the estimator is able to detect mode changes before the resulting changes in velocity are apparent and a significant improvement in velocity estimation is realized. This, in turn, generates the potential for improved closed-loop tracking performance.

1 Introduction

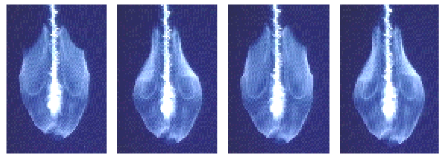
A vision-based automatic tracking system for ocean animals in the midwater has been developed and demonstrated under a program of joint research between the Stanford University Aerospace Robotics Lab and the Monterey Bay Aquarium Research Institute (MBARI) [1–4]. In field tests using MBARI's ROV *Ventana* in the Monterey Bay, this system has demonstrated fully autonomous closed-loop control of *Ventana* to track animals for periods up to 1.5 hours. This tracking system has been designed for both ROV and AUV deployments.

The current tracking system's control laws use velocities derived by differentiating the relative position of the target with respect to the vehicle as measured by the stereo vision sensing system. This system has been very effective in tracking many target specimens but performance can be poor when tracking very mobile and/or small targets. The performance of human pilots when doing this task, however, does not degrade nearly as much when tracking such targets. One difference between the logic currently embedded in this system and the way human pilots operate is that human pilots exploit their *a priori* knowledge of the strongly modal motion behaviors of the tracked animal and the visible body deformations associated with those motions. They do not rely solely on lead information determined through differentiation of relative position.

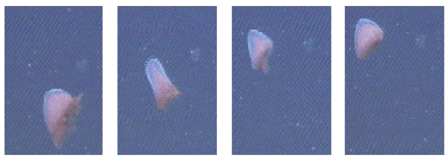
To improve the robustness of the tracker and expand its applicability to smaller and more mobile animals, this paper presents methods for incorporating into the estimator knowledge of the modal motions of the tracking target. Some information about the motion modes can be inferred from measured water-relative velocities. However, visual information of the type that the pilot uses is potentially available to the system, quantified as a classification of the visible body deformations exhibited by the animal [5,6]. This non-traditional knowledge-based lead information exploits the detection of mode switches as an early indicator of acceleration or deceleration, and hence provides improved velocity estimation (e.g., faster convergence).

In [7], an estimator was presented that derives water-relative velocities of the tracking vehicle and target by merging measurements of water-relative velocities with the stereo vision measurements using a Sigma Point Kalman Filter (SPKF) [8]. In this paper, the estimator is augmented with knowledge of the modal nature of the tracking specimen. Improved velocity tracking is demonstrated using a multiple model bootstrap filter[1] [9]. The estimator is then further augmented with discrete measurements from the online vision algorithm that classifies the type of body deformation patterns exhibited by the specimen [5]. This discrete information is fused by recasting the process and sensor models as a dynamic Bayesian network (DBN) [10] and by characterizing the discrete classification of body deformation patterns with a probability distribution conditioned on the propulsive motion mode of the specimen.

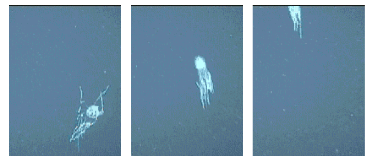
[1] The method referred to by this paper as "bootstrap" filtering is a Sequential Monte Carlo method of filtering that is often called other names including particle filtering, condensation, and Monte Carlo filtering.



(a)



(b)



(c)

Fig. 1. Examples of body motion behaviors: (a) the head of a *Praya* siphonophore exhibiting repetitive pulsing motions with period of about 0.9 sec. time-lapse at 0.43 sec intervals and stabilized, (b) a *Benthocodon* jellyfish making a single pulse swimming motion, time-lapse at 0.27 sec intervals, (c) a *Colobonema* jellyfish makes a sudden swimming motion while being tracked by the automatic system, evading the tracking system, time-lapse at 1 sec intervals. (Images courtesy of MBARI.)

Section 2 of this paper describes the motion modes exhibited by typical midwater ocean animals. In Section 3, a set of process models for the target dynamics within a multiple model bootstrap estimator are proposed to represent the propulsive modes of a typical tracking target. Section 4 details the DBN framework that is used to incorporate the body deformation classifications from the secondary vision algorithm into the bootstrap filter. Finally, Section 5 presents the resulting performance improvements attained by utilizing multiple models and the unique lead information in these visual cues in the new DBN-based filter.

2 A Mode Model of Motion Behavior

In this section, the motion behaviors exhibited by gelatinous animals are enumerated, and a body motion mode model derived from those behaviors is established.[2]

Gelatinous animals generally effect propulsive forces by deforming some part of their bodies (or the entire body) in a pulsing motion. Pumping water

[2] While the automatic tracking system is most commonly used to track gelatinous animals, other types of animal such as small squid and some (low swimming speed) fish species have been tracked. The visible cues related to their motions are also discernible by the vision algorithms of [5]. However, (1) does not apply to those classes of animal.

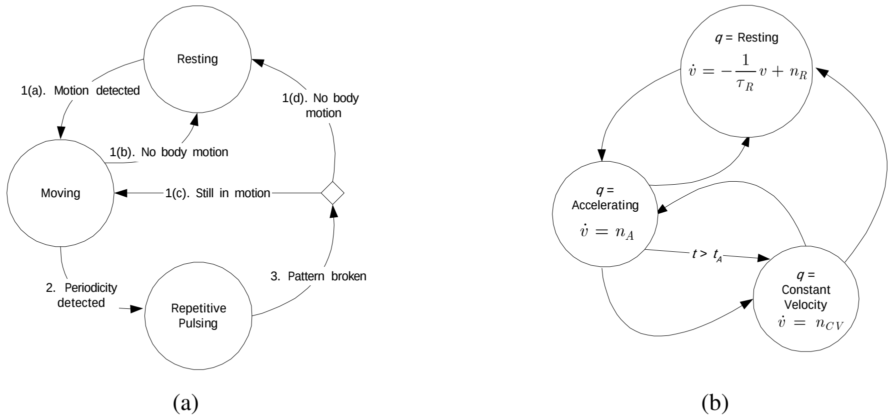


Fig. 2. (a) Gelatinous animal body motion from point of view of observer. (b) The hybrid dynamical system representation of the motion of the tracking target used by the multiple model bootstrap filter. The relationship between the body motions of (a) and the water-relative dynamic modes of (b) is assumed to be one-to-one. Thus, (1) when the body is not actively moving, no thrust is produced, (2) when the body begins moving, the animal accelerates, and (3) periodic motion of the body results an approximately constant velocity.

in and out of the bell portion of their bodies generates a thrust force on the surrounding water. Many species exhibit periodic bell pulsing behaviors in order to propel themselves through the water. Others exhibit these pulsing actions in non-periodic patterns. Fig. 1 shows some image sequences with examples of these motions. For medusa jellyfish, an expression for dynamic thrust given by Equation (1) shows thrust, T, related to water density, ρ, and generated by the time rate of change in the volume of the bell, V, and the velar area, A_v, as proposed by Daniel [11].

$$T = -\frac{\rho}{A_v}\frac{\delta V}{\delta t} abs\left(\frac{\delta V}{\delta t}\right) \tag{1}$$

Fig. 2(a) shows a finite automaton that expresses an observer's perception of body motion mode, including the criteria used to detect switching events. Algorithms for applying this finite automaton to a stream of video images of an object in real time were presented in [5] and [6]. The output of this vision algorithm is a classification of the motion type exhibited by the body of the observed animal at time t:

$$\sigma(t) \in \{Resting, Moving, RepetitivePulsing\} \tag{2}$$

For species whose active swimming modes are accomplished by moving or contracting significant portions of their bodies, the visible indication of those motions will lead the actual changes in velocity as the animal accelerates. In Section 5, this information is shown to assist the estimation of an animal's velocity by allowing the estimator to anticipate accelerations and decelerations before they manifest as significant changes in measured velocity.

3 Multiple Model Estimation in the Water Frame

In [7], estimation of water-relative velocities was accomplished by merging relative bearing measurements from the stereo camera pair with water-relative vehicle velocities measured by a Doppler Velocity Log (DVL). By applying the additional knowledge of the modal nature of the active motions by the target, velocity estimation can be improved.

The use of multiple models in target tracking estimation is a popular method to estimate target motion without direct knowledge of the accelerating inputs on the target, for instance [12]. Approaches such as the Interacting Multiple Model (IMM) estimator [12] or the multiple model bootstrap filter [9] have the effect of adapting the bandwidth of the estimator based on the most probable models of the model set. The model set is typically chosen to have appropriate bandwidth properties for different tracking situations and/or to incorporate specific knowledge of the modes present in a hybrid dynamical system [13]. For a multiple model estimator tracking an unknown maneuvering target, during non-maneuvering periods, a low bandwidth estimator (one with low process noise assumed) is preferable to mitigate the effects of noisy sensors. However, during maneuvers by a target, a higher bandwidth estimator is preferable to allow the state estimate to adapt quickly to the changing conditions. Multiple model estimators for target tracking are designed to adapt by favoring the most likely models based on the evidence provided by the measurements.

To apply these techniques to the underwater tracking system, a three-mode hybrid dynamical model is used, as shown in Fig. 2(b). Because detailed dynamic parameters are not known for a given specimen, only very generic kinematic models are used to model the motion. The $q = Resting$ dynamics include a small white noise acceleration term and a damping term representing the tendency for water-relative velocity to stay low in this mode. $q = Accelerating$ is represented as simply driven by large variance white noise acceleration. The $q = ConstantVelocity$ mode is associated with constant velocity dynamics with a different, more moderate white noise acceleration term. Switching between modes is assumed to be a Markov process for all unmarked transitions. To help capture the case of repetitive pulsing motions that accelerate from rest to a steady-state velocity, a timed transition is added from the $Accelerating$ and $ConstantVelocity$ modes. This timed transition also encodes the delay by the motion classification algorithm when classifying a motion as periodic, which typically requires the observation of 1 or 2 periods of motion. The full process and sensor models for the vehicle and the relative bearing (vision) and water relative velocity (DVL) sensors are presented in [7], and are not dependent on the discrete mode q of the target.

4 Incorporation of Visual Classifications of Body Motion via a Dynamic Bayesian Network Model

In this section, the modeling of a stochastic dynamic system as a dynamic Bayesian network is briefly defined and related to the familiar Kalman filter. Then the model of the tracking system's target is recast as a DBN, including an additional observation model to represent the motion classifier.

4.1 Bayesian and Dynamic Bayesian Networks

A Bayesian network (BN) is a graphical representation of a model of the probabilistic relationships and conditional independence of a set of variables [14]. The network in Fig. 3 is an example. An arrow connecting two nodes in the network (an edge) indicates that a conditional dependence exists for the child variable (the node pointed to) upon the parent variable. Based on the conditional dependencies defined by the network and the distributions associated with the variables and their parents, full joint distributions may be computed for any particular assignments to the variables in the network. That is, if each of N variables in the network are X_i,. . . ,X_N, then the joint probability that they are assigned to x_i,. . . ,x_N, respectively, is given by

$$p(x_i, \ldots, x_N) = \prod_{i=1}^{N} p(x_i | parents(X_i)) \tag{3}$$

A dynamic Bayesian network (DBN) is a Bayesian network describing the relationships between dynamic state variables that evolve over time. In this case, the network illustrates the dependencies between these variables at given instances in time. The Kalman filter (KF) [15] for linear Gaussian state-space systems is one example. The continuous-time portion of the DBN of Fig. 3 could represent the relationships between the states x_{k-1} and x_k and the observations y_k as modeled by a discrete-time KF (with no control

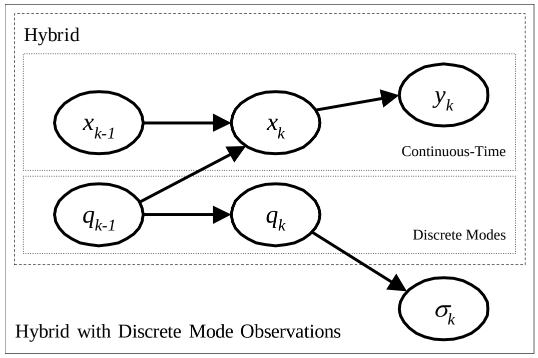


Fig. 3. A dynamic Bayesian network (DBN) representation of the process and sensor models of the multiple model estimator, including the output σ_k from the online body motion classifier, Equation (2).

inputs). Then the conditional probabilities represented by the edges in that DBN are the conditional Gaussian distributions of x_k given x_{k-1} defined by the state transition matrix and the process noise covariance, and of y_k given x_k based on the observation matrix and the observation noise covariance.

4.2 Estimation of the Target's Velocity with a DBN-based Bootstrap Filter

In [16], bootstrap filtering was applied to a set of hybrid dynamical systems by representing the systems being monitored as dynamic Bayesian networks. The conditional probability densities (CPDs) associated with the DBNs were of varying forms including Gaussian, simple conditional probability tables (CPTs) and softmax densities [17]. A similar approach to the problem of estimating the velocities of a tracked ocean animal is employed here, combining dissimilar probabilistic models for the evolution and observation of both continuous states (the velocities) and discrete variables (the propulsive mode of the animal, q, and the discrete observation variable, σ).

Fig. 3 illustrates the state evolution and observation models of the target, cast in the form of a DBN. Note the inclusion of the discrete motion classifier, whose output, σ_k, is assumed to be dependent only upon q_k. To use this DBN in a bootstrap filter, some CPD (that can be numerically sampled) relating the observation σ_k to the value of q_k is necessary. Based on the error rates expected when running the visual classifier [5], a simple CPT can be constructed and utilized to approximate this relationship.

The bootstrap filter, with the discrete classifier observation model added is evaluated as in [9] with two modifications based on the models used here. The first change is to the mode transition model that updates each sample for mode changes from step $k-1$ to k. In [9], the finite automaton is assumed to be entirely Markovian. However, the mode model as specified by Fig. 2(b), requires time-in-mode for a sample to be tracked. This is accomplished by augmenting the continuous state vector with this variable and integrating it throughout the sample's life, resetting it to zero upon any mode transition. For each sample in the filter, during the time update step, this time is checked before applying the Markovian mode update for any superseding switching criteria.

The second change from [9] is more significant, the modification of the measurement update step of the filter. In systems with only continuous variables being observed, the measurement update consists of calculating the probability of y_k, the continuous-time observation, given the continuous state of each sample, $x_k^{(i)}$. Here, the measurement update requires the computation of a joint probability of (y_k,σ_k) given (x_k, q_k) of the sample. Because the values of the (x_k, q_k) pair (and their parents x_{k-1} and q_{k-1}) have been instantiated to specific values, the probabilities of y_k and σ_k given x_k and q_k become independent. Therefore the joint probability is given by the product of the two separately conditioned probabilities:

$$p(y_k, \sigma_k | x_k, q_k) = p(y_k | x_k) p(\sigma_k | q_k) \tag{4}$$

5 Results

5.1 Test Data Description

To test the performance of the multiple model and DBN-based estimators, test data was generated using the baseline tracking control software to track a simulated moving target. This target was tracked through a sequence of propulsive modes, q, in the following order: resting, acceleration, steady swimming, resting. Noisy, distorted pixel measurements for each camera of the stereo pair as well as compass and angular rate readings were generated at the 10 Hz update rate of the tracking system. DVL velocities with additive zero-mean Gaussian noise at a standard deviation of 3 cm/s were generated at 5 Hz (the maximum update rate of the DVL deployed at MBARI), with angular rates of the vehicle coupled into the measurements based on the location of the DVL on the ROV *Ventana*.

The vector of values for σ, the noisy and imperfect mode classification, are generated based upon the expected error rates of the body motion classifier vision algorithm as applied to the "true" mode, q, of the target. This CPT, S, is given by the following:

q	$p(\sigma = Resting)$	$p(\sigma = Moving)$	$p(\sigma = RepPulse)$
$Rest$	0.90	0.08	0.02
$Accel$	0.15	0.80	0.05
$ConstVel$	0.03	0.09	0.88

5.2 Estimator Detailed Design

This data set was used to compare the performance of three estimators: (i) the UKF estimator of [7], (ii) a multiple model bootstrap filter as described in this paper but operating only on continuous state observations, and (iii) the multiple model bootstrap filter from (ii) supplemented by the classifier data, σ. For all models, the sensor noise standard deviations assumed were 2 pixels for each camera measurement (in a 160x120 image) and 3 cm/s on the DVL water velocities. Vehicle disturbance process noise standard deviation was set to 0.5 volts on all axes (on a scale with limits at +/- 5 volts for thruster command levels).

Several parameters specify the target models of Fig. 2(b) for the multiple model estimators. Target process noise terms were specified with standard deviations of 2 cm/s^2 ($Resting$, $ConstantVelocity$) and 10 cm/s^2 ($Accelerating$). The exponential decay (damping) term of the $Resting$ mode dynamics was specified with a time constant, τ_R, of 2 sec. The time-in-mode limit for the $Accelerating$ mode, t_A was set to 5 seconds.

For the UKF, which uses a single model design, the target model is a constant velocity model, with a single value for white noise acceleration standard deviation of 4 cm/s^2. This choice lies between the settings within the multiple model estimators for quiescent modes and the maneuvering mode, and is the result of the compromise required such that a single mode estimator will track adequately through more than one type of motion behavior.

The Markov mode switching probabilities are given by H below, where h_{ij} represents the probability of switching to mode i from mode j. The modes are indexed from 1 to 3 in the order presented in Equation 2.

$$H = \begin{bmatrix} 0.85 & 0.33 & 0.10 \\ 0.15 & 0.34 & 0.05 \\ 0.0 & 0.33 & 0.85 \end{bmatrix}$$

The direct use of the CPT probabilities from S in Equation 4 were found to make the estimator too sensitive to errors in the classifier output, σ, forcing the estimator to be too trusting in σ over evidence in the continuous measurements and the priors. To blunt this effect and achieve a better balance between the discrete classifier outputs and the prior belief states, a more uncertain version of S was used in the DBN-based estimator, given below as S_{DBN}:

q	$p(\sigma = Resting)$	$p(\sigma = Moving)$	$p(\sigma = RepPulse)$
$Rest$	0.75	0.18	0.07
$Accel$	0.20	0.67	0.13
$ConstVel$	0.10	0.18	0.72

5.3 Results Summary

A marked improvement in overall tracking quality is demonstrated by the augmented multiple model estimator (iii) over the other two estimators. The test trajectory was partitioned into phases by the motion of the target, and the estimator errors for each algorithm are tabulated in Table 1. The modal trajectory of the target consists of a sequence of $\{Resting$, $Accelerating$, $ConstantVelocity$, $Resting\}$, and for the purposes of judging the performance of the estimators, the transition from $ConstantVelocity$ to $Resting$ is broken down into two phases (deceleration and resting, where velocity is nearly zero).

The estimator that utilizes the output of the body motion mode classifier (iii) outperforms both of the other estimators in all motion phases. The performance improvement is particularly notable in the deceleration portion of the trajectory, where this estimator is able to anticipate the deceleration based on visual cue of the ceasing of body motions by the target (as recognized by the body motion classifier).

The mode probabilities calculated for each multiple model estimator are shown in Fig. 4, with (a) showing results from the estimator of (ii) which uses continuous measurements only, and (b) showing the results from estimator (iii). These results demonstrate that the uncertainties of the measurements and

Table 1. Average of 2-norm of error in target velocity estimates (cm/s), by target motion phase.

Estimator	$Rest1$	$Accel$	$ConstVel$	$Decel$	$Rest2$	Overall
(i) UKF	1.1	2.4	1.4	3.1	1.7	1.8
(ii) MM	0.7	4.1	4.5	1.2	1.3	2.5
(iii) MM-DBN	0.4	2.3	1.2	1.1	0.9	1.0

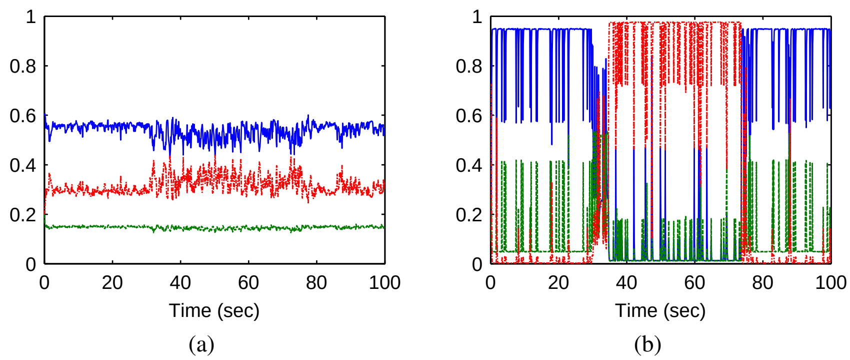
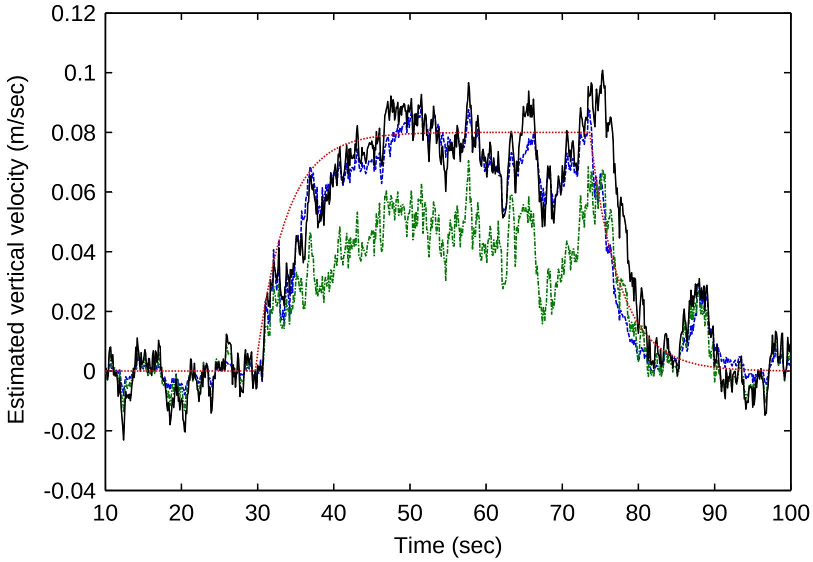


Fig. 4. Mode probabilities calculated by (a) multiple model bootstrap, and (b) bootstrap utilizing DBN model and discrete mode observations. Legend – $\hat{p}(q = Rest)$: blue solid, $\hat{p}(q = Accel)$: green dashed, $\hat{p}(q = ConstVel)$: red dash-dot.

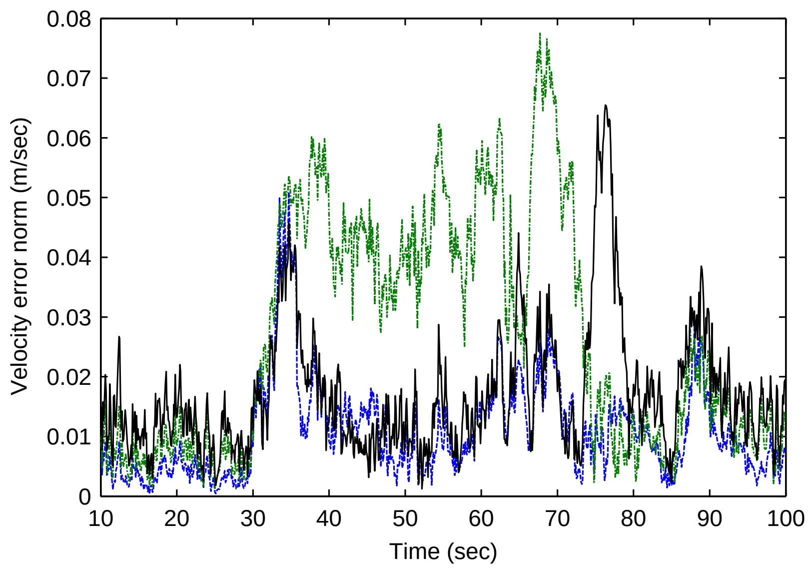
of the vehicle and target dynamics are too high to discern modal information successfully without the extra information from the online classifier. The noisiness in the mode probabilities in (b) are primarily in response to errors in σ. This response is momentary, countered by the evidence in the priors and continuous measurements, keeping overall state tracking errors due to classifier errors small. Velocity tracking results for the target's velocity in the vertical direction and the 2-norm of the overall velocity error are presented in Fig. 5 (a) and (b), respectively.

6 Conclusion

It has been shown that a significant advantage can be imparted to a multiple mode estimator of target velocity by incorporating the extra information from a vision-based body motion mode classifier. This information allows the estimator to discern the modes of the tracked specimen in spite of large uncertainties present in the measurements, the dynamics of the vehicle and target, and the disturbances encountered by the tracking vehicle in the underwater environment. This approach allows the on-line estimator to interpret the scene in a way that is modeled after the manner in which human pilots do. With these improved estimates of the tracking target's velocities, the performance of the tracking control system can be expected to improve, especially when tracking actively maneuvering targets.



(a)



(b)

Fig. 5. Velocity tracking results comparing performance with UKF, multiple model bootstrap, and bootstrap utilizing DBN model and discrete mode observations. (a) Tracking of velocity in vertical direction (m/s), (b) 2-norm of tracking error for target velocity (m/s). Legend – UKF: black, MM bootstrap: green dash-dot, DBN-based MM bootstrap: blue dashed, true velocity [(a) only]: red dotted.

References

1. J. Rife and S. M. Rock, "Field experiments in the control of a jellyfish tracking ROV," in *Proceedings of the IEEE OCEANS Conference*, pp. 2031–2038, 2002.
2. J. Rife, *Automated Robotic Tracking of Gelatinous Animals in the Deep Ocean*. PhD thesis, Stanford University, Stanford, California, December 2003.
3. J. Rife and S. M. Rock, "Design and validation of a robotic control law for observation of deep-ocean jellyfish," *IEEE Transactions on Robotics*, submitted.
4. A. M. Plotnik and S. M. Rock, "Relative position sensing and automatic control for observation in the midwater by an underwater vehicle," in *Proceedings of the Unmanned Untethered Submersible Technology Conference (UUST)*, (Durham, NH), AUSI, Aug 2005.
5. A. M. Plotnik and S. M. Rock, "Improving performance of a jelly-tracking underwater vehicle using recognition of animal motion modes," in *Proceedings of the Unmanned Untethered Submersible Technology Conference (UUST)*, (Durham, NH), AUSI, Aug 2003.
6. A. M. Plotnik and S. M. Rock, "Quantification of cyclic motion of marine animals from computer vision," in *Proceedings of the IEEE OCEANS Conference*, pp. 1575–1581, 2002.
7. A. M. Plotnik and S. M. Rock, "A multi-sensor approach to automatic tracking of midwater targets by an ROV," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, (San Francisco, CA), Aug 2005.
8. R. van der Merwe, E. Wan, and S. Julier, "Sigma-point Kalman filters for nonlinear estimation and sensor-fusion: Applications to integrated navigation," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, (Providence, RI), Aug 2004.
9. S. McGinnity and G. W. Irwin, *Sequential Monte Carlo Methods in Practice*, ch. 23. Springer, 2001.
10. T. Dean and K. Kanazawa, "A model for reasoning about persistence and causation," *Computational Intelligence*, vol. 5, no. 3, pp. 142–150, 1989.
11. T. Daniel, "Mechanics and energetics of medusan jet propulsion," *Canadian Journal of Zoology*, vol. 61, pp. 1406–1420, 1983.
12. Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley, 2001.
13. X. Koutsoukos, J. Kurien, and F. Zhao, "Estimation of distributed hybrid systems using particle filtering methods," in *Hybrid Systems: Computation and Control*, 2003.
14. J. Pearl and T. Verma, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.
15. R. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.
16. D. Koller and U. Lerner, *Sequential Monte Carlo Methods in Practice*, ch. 21. Springer, 2001.
17. D. Koller, U. Lerner, and D. Angelov, "A general algorithm for approximate inference and its application to hybrid bayes nets," in *Proceedings of the Fifteenth Annual Conference on Uncertainty in AI (UAI)*, pp. 324–333, 1999.