APPLICATION OF PREDICTION-BASED PARTICLE FILTERS FOR TELEOPERATIONS OVER THE INTERNET

Jae-young Lee, Shahram Payandeh, and Ljiljana Trajković
School of Engineering Science
Simon Fraser University
8888 University Drive, Burnaby, BC
V5A 1S6, Canada
jla155@sfu.ca, shahram@cs.sfu.ca, ljilja@cs.sfu.ca

ABSTRACT

In this paper, we present the prediction-based particle filter approach for processing motion and force data in teleoperation over the Internet. We first introduce the prediction-based particle filter algorithm, one of the Sequential Monte Carlo methods based on the recursive Bayesian prediction. The prediction algorithm is applied to dynamic models of the motion and force data flows in the state-space formulation. It is applied to the motion data transmitted to the slave controller and to the reflecting force data received at the master controller. Experiments are performed using the haptic device within a virtual 3D graphical environment. In each experiment, the motion and reflecting force data extracted from the haptic device are used to verify the prediction performance of the proposed method.

KEY WORDS

Teleoperation, Internet, Particle filter, Sequential Monte Carlo method, Bayesian prediction

1. Introduction

Internet-based teleoperation is an interactive application where a human user transmits motion data from a master controller while simultaneously receiving the reflecting force data from a slave controller stationed in a remote environment. Unlike most other Internet applications that require reliable data transmission, for stable operation interactive applications need to maintain a constant frequency of data transmission. Hence, in a teleoperation, both the time delay and the reliable data transmission should be considered.

The end-to-end Internet time delay consists of the propagation delay, transmission delay, processing delay, and queuing delay. Unlike the first three delay components, the queuing delay varies with time due to the Internet traffic conditions. The transport control protocol (TCP) and user datagram protocol (UDP) are two widely used Internet transport protocols. TCP, which provides reliable data transmission, often introduces large variable delay due to its retransmission scheme and congestion control. Thus, it has been suggested that UDP be employed for teleoperation even though it does not guarantee reliable data transmission and may lead to data losses [1].

Many approaches have been proposed to solve the time delay and data loss issues in teleoperation over the Internet. Various control systems approaches have been suggested, including the wave-variable transformation and its extensions [2], [3]. Prediction-based signal processing approaches that perform motion and force predictions have been also proposed [4], [5]. The Kalman filter method, which provides a recursive solution to the linear prediction and estimation, was proposed as a prediction-based approach [6]. The motion and reflecting force data are impaired by the presence of the Internet delay. Hence, these signal processing approaches are expected to compensate for the delays that vary over time. Nevertheless, the motion and reflecting force data are often difficult to predict if they involve nonlinear and non-Gaussian system characteristics. For example, hand movement patterns from a master controller can be highly nonlinear and the traditional Kalman filter may fail to provide accurate prediction. The reflecting force data may be even more difficult to predict since the data need to be sent at relatively high frequencies in order to ensure realistic and continuous force.

The particle filter algorithm, also known as the bootstrap filter or the Condensation filter, is a Sequential Monte Carlo (SMC) method that provides suboptimal solutions to the recursive Bayesian approach [7], [8]. Due to its robust prediction and estimation performances in nonlinear and non-Gaussian environments, the algorithm has been widely used in communications, image and speech signal processing, and robotics [9]-[12]. The particle filter method may be applied to any nonlinear dynamic model using a state-space framework, and, hence, it can be applied to the dynamic models of the motion and force data in the state-space formulation.

In this paper, we present the prediction-based particle filter algorithm to predict the motion and reflecting force data that suffer from the Internet delays that vary over time. In Section 2, we introduce the generic particle filter algorithm with the prediction-based formulation. In Section 3, we discuss the nonlinear state-space models of the motion and reflecting force data, and address the issues dealing with prediction of data using the particle filter algorithm. Experimental results with the implemented prediction-based particle filter algorithm are given in Section 4. We conclude with Section 5.

2. Prediction-Based Particle Filter Algorithm

A discrete-time dynamic system may be represented using a state-space model, where unknown states of the system are predicted or estimated based on available noisy observations. The particle filter method performs suboptimal prediction and estimation within the recursive Bayesian approach in case when the dynamic system is nonlinear and non-Gaussian. Using the particle filter method, the true posterior density in such nonlinear and non-Gaussian dynamic systems can be approximated by a simulation-based approach. The discrete time nonlinear state-space model can be expressed as:

$$\begin{aligned} x_{k+1} &= g_k(x_k, u_k) \\ \tilde{x}_k &= h_k(x_k, v_k) \end{aligned}, \tag{1}$$

where x_k and $\tilde{x}_k$ are respectively the state and the observation system at time k, g_k and h_k are nonlinear state and observation transition functions, and u_k and v_k are state and observation noise sequences, which may be non-Gaussian. In a state-space model, the prediction of the true state at time $k+1$ can be obtained based on the current state x_k and available observations $\tilde{x}_{1:k}$. Based on the recursive Bayesian approach, the optimal predictor of the true state at time $k+1$ can be expressed by the conditional means:

$$\hat{x}_{k+1|k} = \int x_{k+1} p(x_{k+1} \mid \tilde{x}_{1:k}) dx_k , \tag{2}$$

where $\hat{x}_{k+1|k}$ denotes the one-step-ahead prediction of the state x_{k+1} given available observations $\tilde{x}_{1:k}$. According to the Bayesian approach, the posterior density should be evaluated recursively solving two density functions [8]:

$$p(x_{k+1} \mid \tilde{x}_{1:k}) = \int p(x_{k+1} \mid x_k) p(x_k \mid \tilde{x}_{1:k}) dx_k \tag{3}$$

$$p(x_k \mid \tilde{x}_{1:k}) = \frac{p(\tilde{x}_k \mid x_k) p(x_k \mid \tilde{x}_{1:k-1})}{p(\tilde{x}_k \mid \tilde{x}_{1:k-1})} . \tag{4}$$

Equations (3) and (4) are respectively the prediction and update procedures for finding the optimal solution. They are not computationally tractable due to the integral forms. Hence, as a suboptimal solution, the particle filter method is used to approximate the posterior densities. Based on the prediction-based particle filter algorithm, (3) can be approximated as [13], [14]:

$$p(x_{k+1} \mid \tilde{x}_{1:k}) \approx \sum_{i=1}^{N_s} w_{k+1}^i \delta(x_{k+1} - x_{k+1}^i) , \tag{5}$$

where N_S is the number of particles, $\delta(.)$ is the Dirac delta function, and w_{k+1}^i is the importance weight that can be computed as:

$$w_{k+1}^i \propto w_k^i \frac{p(\tilde{x}_{k+1} \mid x_{k+!}^i) p(x_{k+1}^i \mid x_k^i)}{q(x_{k+1}^i \mid x_k^i, \tilde{x}_{k+1})} . \tag{6}$$

The importance density $q(.)$ may be chosen to be equal to the prior density in order to minimize the variance of the importance weights:

$$q(x_{k+1} \mid x_k^i, \tilde{x}_{k+1}) = p(x_{k+1} \mid x_k^i) . \tag{7}$$

Hence, the importance weight can be simplified as:

$$w_{k+1}^i \propto w_k^i \cdot p(\tilde{x}_k \mid x_k^i) . \tag{8}$$

Prior to performing the resampling step, the importance weights (8) should be normalized so that $\sum_i w_{k+1}^i = 1$.

The simplified illustration of the particle filter algorithm including the resampling step is shown in Figure 1. After N_S number of particles is randomly distributed in the first iteration k, the importance weights are computed for each particle in order to obtain the approximation of $p(x_k | \tilde{x}_{1:k-1})$. The resampling step is then performed to regenerate the predicted samples based on the weighted samples. In the resampling process, the particles with small weights are eliminated while the particles with high weights are concentrated. The large number of particles gives more accurate stochastic approximation, which in general provides the reasonable prediction performance. However, an efficient number of particles may be selected to avoid computational burden. The prediction-based particle filter algorithm is described in Table 1.

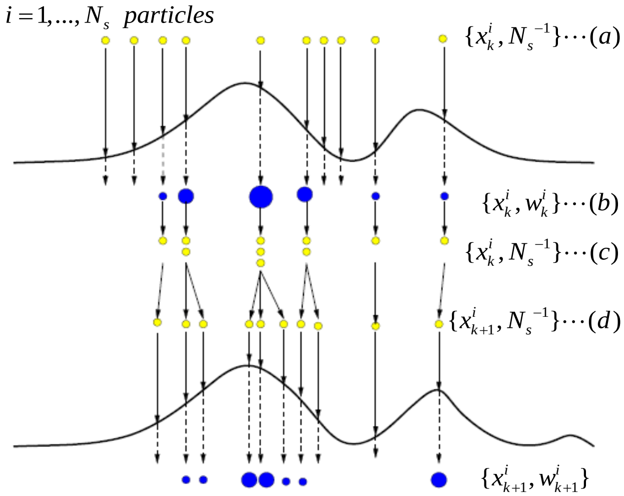

Figure 1. Graphical representation of the particle filter algorithm [8]. Starting from the initial state k (a), illustrated are the weighted measure (b), resampling (c), and prediction of next state $k+1$ (d).

Table 1
Prediction-based particle filter algorithm.

Step 1. Initialization

Draw the initial state randomly and define initial parameters.

Step 2. Prediction

Draw $x_{k+1}^i \sim p(x_{k+1} \mid x_k^i),\ 1 \le i \le N_s$

Step3. Update

Evaluate importance weights according to (8) and normalize the weights.

Step 4. Resampling

Multiply/suppress samples with high/low importance weights.

***Step 5.* Iteration**

Increase time step and go back to Step 2.

3. Prediction of Motion and Force Using the Particle Filter Algorithm

The motion data transmitted from the master controller may be impaired by the Internet delay. Hence, the receiving motion data at the slave controller need to be predicted in order to compensate for the delay. The predicted motion data, consisting of positions over time samples, may cause a contact with an object or a surrounding environment, which in turn generates reflecting force data. The force data that feeds into the master controller may also suffer from a similar Internet delay. Therefore, the reflecting force data also need to be predicted in order to compensate for the delay. The motion and reflecting force data sequences may be formulated as state-space models. The state-space formulations of the motion and reflecting force data in a teleoperation scenario are shown in Figure 2.

3.1 Motion Prediction

The motion data transmitted from the master controller consist of positions over time samples. As shown in Figure 2, the motion data, which may be nonlinear or non-Gaussian, are formulated as a state-space model. The true motion data are then predicted using the prediction-based particle filter algorithm based on available observations, which are impaired by the Internet delays.

For a single degree of freedom (DoF) teleoperation system, the true position x_k at time k is transmitted via the Internet and it is delayed by n time steps. The impaired observation received at the slave controller is denoted as $\tilde{x}_{k-n}$. In a state-space model, the prediction of the true position at time $k{-}n{+}1$ can be obtained based on the current state x_{k-n} and available observations $\tilde{x}_{1:k-n}$. Hence, as in (2), $\hat{x}_{k-n+1|k-n}$ represents the one-step-ahead prediction of the state x_{k-n+1} given available observations $\tilde{x}_{1:k-n}$. Using the prediction-based particle filter method, $\hat{x}_{k-n+1|k-n}$ can be computed by approximating the posterior density function $p(x_{k-n+1}|\tilde{x}_{1:k-n})$ based on (5)-(8) evaluated at time $k-n$.

In order to implement the particle filter algorithm to the motion data prediction problem, each step introduced in Section 2 should be executed. After the initialization step that randomly defines the initial state of the motion model, the prediction step is performed to obtain samples x_{k-n+1}^i from the prior density $p(x_{k-n+1}|x_{k-n}^i)$, where $1 \le i \le N_s$. In the update step, the new state x_{k-n+1} is assigned by using importance weights (8). Since the importance density $q(.)$ was chosen to be the prior density and the state and observation noise was assumed Gaussian, the evaluation of the importance weights can be simplified as:

$$w_{k-n+1}^i = e^{-\frac{1}{2\sigma^2}\left(\tilde{x}_{k-n} - x^i_{k-n}\right)}. \qquad (9)$$

Equation (9) gives the importance weights of the $i-th$ particle at time $k-n+1$ and needs to be normalized for the resampling. For the resampling step, the new set of states x_{k-n+1} is determined based on the importance weights. After cumulative distribution functions (CDFs) of the normalized weights are constructed, each element of the CDFs is compared with the uniformly distributed function to determine the low or high weights. The new set of states is then regenerated based on the high weighted samples.

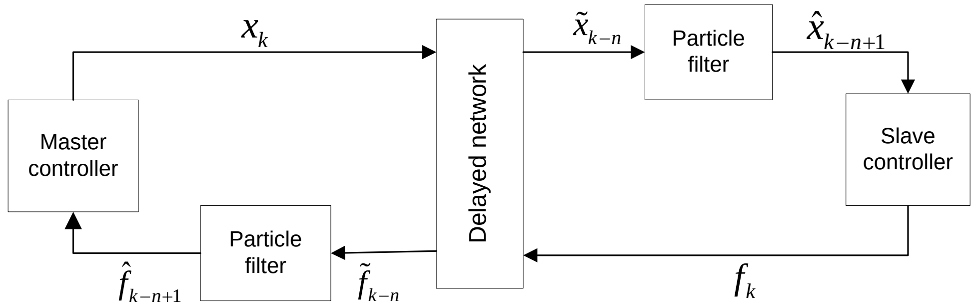

Figure 2. State-space formulations of motion and reflecting force data in the teleoperation scenario and the proposed prediction scheme using the particle filter method.

3.2 Force Prediction

Prediction of the reflecting force data can be achieved by a similar approach. The force data generated by a contact with an object or a surrounding environment are fed into the master controller. The transmission of the force data over the Internet can also be formulated as a nonlinear or non-Gaussian state-space model, as shown in Figure 2. The true force data are then predicted by the prediction-based particle filter algorithm given the available observations. Since the force data must be sampled at a relatively high frequency rate (above 1,000 Hz) in order to achieve realistic and continuous force, they are relatively difficult to predict.

Let the true force data generated from the slave controller at time k be f_k in a single DoF teleoperation system. This is the feedback force data delayed n time steps and transmitted over the Internet. The force data at master controller received through the Internet is viewed as the impaired observation and it can be expressed as $\tilde{f}_{k-n}$. Similar to the position prediction problem, the one-step-ahead prediction of the state f_{k-n+1} given available observations $\tilde{f}_{1:k-n}$ can be represented as $\hat{f}_{k-n+1|k-n}$. Based on the prediction-based particle filter approach, the posterior density function $p(f_{k-n+1}|\tilde{f}_{1:k-n})$ can be approximated by computing the importance weights based on (5)-(8). The implementation of the particle filter algorithm to the force data can be achieved by applying steps shown in Table 1. The dynamic model of the reflecting force data in the state-space framework and its prediction approach using the particle filter method are illustrated in Figure 2.

4. Experiments

In order to verify the proposed prediction-based particle filter method for a teleoperation scenario, we performed experiments using the PHANTOM Desktop haptic device. In conjunction with virtual 3D graphical environment, the haptic device provides positioning input while receiving feedback force by a 6-DoF manipulation. In this experimental scenario, the PHANTOM Desktop was used as a master controller where a human operator provides motion data. A 3D virtual teleoperator model was programmed as a slave controller using the C++ and OpenGL libraries. The virtual teleoperator based on the 4-DoF Selective Compliance Assembly Robot Arm (SCARA) model was designed. According to the movement from the master controller, the contact force that feeds into the master controller is generated when the tip of the SCARA model collides with objects. Note that the positions of the 4-DoF SCARA model are mapped to the master controller so that the PHANTOM Desktop is only capable of manipulating 4-DoF. The experimental scenario consisting of the PHANTOM Desktop haptic device interfaced with the 3D virtual teleoperator is shown in Figure 3.

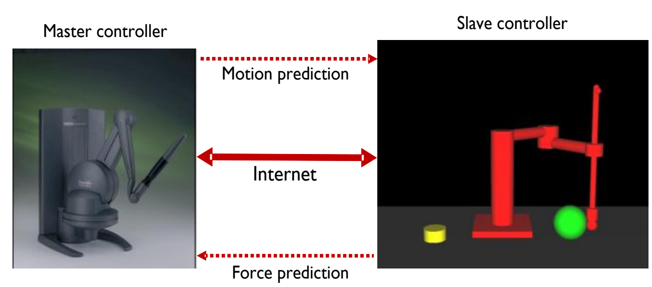

Figure 3. Prediction-based teleoperation experimental scenario: The PHANTOM Desktop haptic device and the virtual 3D graphical representation are used for master and slave controllers, respectively.

In the experimental scenario, we simulated the Internet time delay model between the master and slave controllers. TCP provides reliable data transmission between the controllers. However, due to the TCP retransmission and congestion control mechanism, large fluctuations of time delay may be introduced. Hence, it has been suggested that UDP may be used for the teleoperation because it introduces relatively low time delay variations. The experimental scenario provided the simulated Internet time delay model typically observed in the UDP transmission [1], [6]. The model that introduces fluctuation and jitter is shown in Figure 4. The maximum and average time delay was 132 ms and 63 ms, respectively, over a five-second interval. In this experiment, we assumed that the motion data transmitted to the slave controller and the force data received at the master controller experienced identical delay, shown in Figure 4.

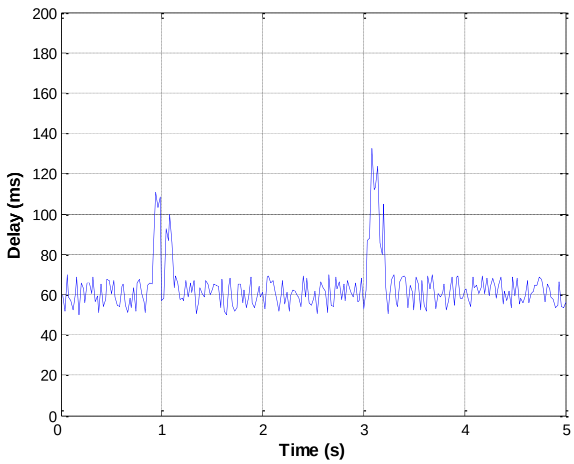

Figure 4. The Internet time delay model with variations.

To verify the prediction performance of the particle filter method, one-dimensional motion and reflecting force data at each stage were extracted over a five-second interval. The experimental scenario is based on the 3D virtual representation. The sampling rate of the motion data was 50 Hz to enable the human eye to perceive continuous motion. The sampling rate of the reflecting force data rendered by the PHANTOM Desktop haptic device was 1,000 Hz in order to maintain realistic and continuous force. It is advised that the motion data and force data should be sampled at no less than 30 Hz and 1,000 Hz, respectively.

The one-dimensional true motion and reflecting force data obtained from the master and slave controllers are shown in Figure 5. The observed motion and force data that are delayed based on the time delay model shown in Figure 4 are also shown in Figure 5. The delayed motion and force data were impaired by the time delay model due to the fluctuation of time delay.

The prediction-based particle filter method was implemented in order to predict the motion and reflecting force data. In both motion and force prediction cases, 100 to 500 particles were used. For simplicity, the state and observation noises were assumed to be Gaussian with zero means and unit variances. The predicted motion data and reflecting force data are shown in Figure 6.

In general, the large number of particles gives improved performances in the motion and force predictions. However, the large number of particles introduces computational complexity, which may introduce time delay. Since the teleoperation is an interactive application that requires the real-time operation, such computational requirements may adversely affect performing real-time operations. Hence, it may be necessary to efficiently select the number of particles. The mean square error (MSE) of the motion and reflecting force predictions based on the selected number of particles is shown in Table 2. The number of particles can be efficiently selected while maintaining the MSE performances.

Table 2
MSE vs. number of particles.

Number of particles	Motion error (mm)	Force error (Newton)
100	4.423	0.1416
200	4.366	0.1406
300	4.037	0.1373
400	4.027	0.1358
500	4.009	0.1357

5. Conclusion

In this paper, we considered the prediction of motion and force for teleoperation over the Internet by using the particle filter algorithm. The prediction-based particle filter was introduced and was applied to the motion and reflecting force predictions in a time-varying network such as the Internet. The experiments were performed using the PHANTOM Desktop haptic device in a virtual 3D graphical environment. The experiments showed that the prediction-based particle filter algorithm successfully performed the one-step-ahead predictions of the motion and force data.

This paper addressed one of the signal processing approaches to overcome the Internet delay in teleoperation scenarios. Signal processing solutions may also need to be combined with reliable control systems in order to improve bilateral teleoperation systems. More efficient particle filter algorithms also need to be designed to address the complexity and operation time issues by adaptively selecting the number of particles [15].

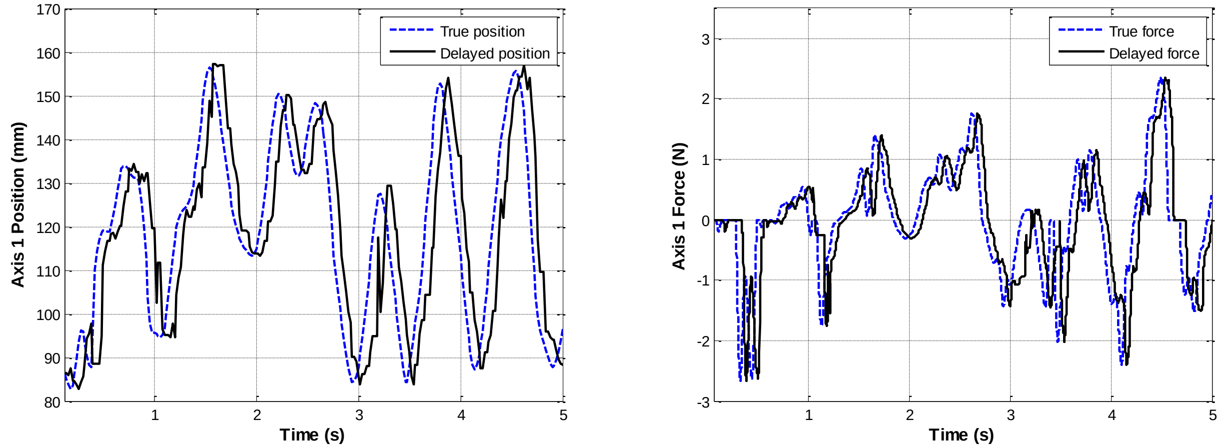

Figure 5. True and delayed motion data (left) and true and delayed feedback force data (right) at the master and slave controllers over a five-second interval.

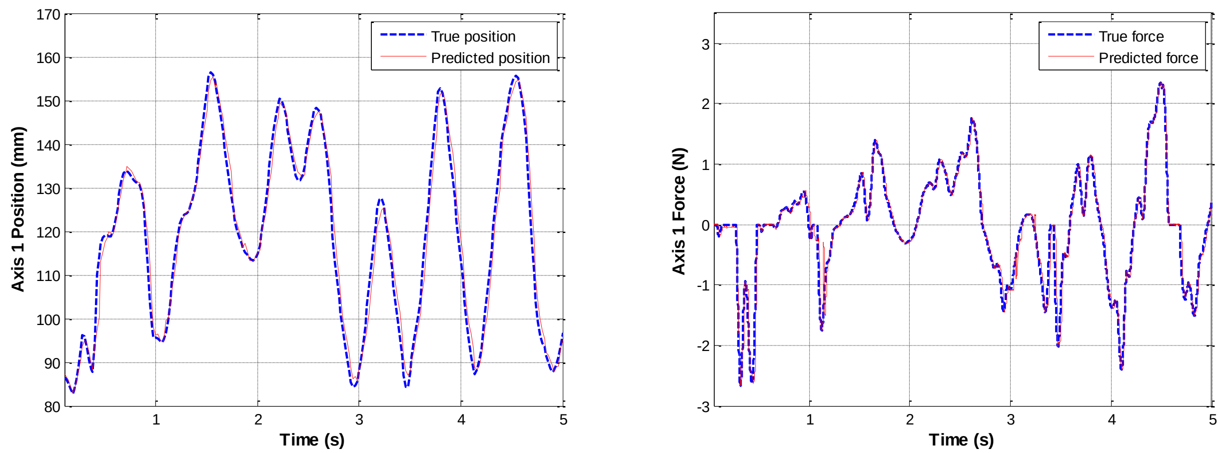

Figure 6. Predicted motion data (left) and feedback force data (right) at the slave and master controllers over a five-second interval.

References

[1] E. Kamrani, H. Momeni, and A. Sharafat, "Modeling Internet delay dynamics for teleoperation," *IEEE International Conference on Control Applications*, Toronto, ON, Canada, Aug. 2005, pp. 1528-1533.

[2] G. Niemeyer and J. J. E. Slotine, "Designing force reflecting teleoperators with large time delays to appear as virtual tools," *IEEE International Conference on Robotics and Automation*, Albuquerque, NM, USA, Apr. 1997, pp. 2212–2218.

[3] K. Kawashima, K. Tadano, G. Sankaranarayana, and B. Hannaford, "Bilateral teleoperation with time delay using modified wave variables," *IEEE International Conference on Intelligent Robots and Systems*, Nice, France, Sept. 2008, pp. 424-429.

[4] S. Clarke, G. Schillhuber, M. Zach, and H. Ulbrich, "Prediction-based methods for teleoperation across delayed networks," *Springer-Verlag, Multimedia Systems*, vol. 13, no. 4, Oct. 2007.

[5] S. Clarke, G. Schillhuber, M. Zach, and H. Ulbrich, "The effects of simulated inertia and force prediction on delayed telepresence," *Presence*, vol. 16, no.5, pp. 543-558, Oct. 2007.

[6] S. Munir and W. Book, "Internet-based teleoperation using wave variables with prediction," *IEEE Transactions on Mechatronics*, vol. 7, no. 2, pp. 124-133, June 2002.

[7] M. Sanjeev Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174-188, Feb. 2002.

[8] A. Doucet, N. de Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N, Freitas, and N. Gordon, Eds. New York: Springer-Verlag, 2001, pp. 4-14.

[9] Y. Lu and S. Payandeh, "Cooperative hybrid multi-camera tracking for people surveillance," *IEEE Canadian Journal of Electrical and Computer Engineering*, vol. 33, no. 3, pp. 145-152, 2008.

[10] R. Mottagi and S. Payandeh, "Coordination of multiple agents for probabilistic object tracking," *IEEE Canadian Conference on Computer and Robot Vision*, Victoria, BC, Canada, May 2005, pp. 162-167.

[11] R. Mottagi and S. Payandeh, "An overview of a probabilistic tracker for multiple cooperative tracking agents," *IEEE International Conference on Advanced Robotics*, Seattle, WA, USA, July 2005, pp. 888-894.

[12] D. Schulz, and D. Fox, "Bayesian color estimation for adaptive vision-based robot localization," *IEEE International* Conference *on Intelligent Robots and Systems*, Sendai, Japan, Sept. 2004, vol. 2, pp. 1884-1889.

[13] G. Kitagawa and S. Sato, "Monte Carlo smoothing and self-organising state-space model," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N, Freitas, and N. Gordon, Eds. New York: Springer-Verlag, 2001, pp. 177-195.

[14] F. Desbouvries and B. Ait-El-Fquih, "Direct versus prediction-based particle filter algorithm," *IEEE Workshop on Machine Learning for Signal Processing*, Cancun, Mexico, Oct. 2008, pp. 303-308.

[15] D. Fox, "Adapting the sample size in particle filters through KLD-sampling," *International Journal of Robotics Research*, vol. 22, no. 12, pp. 985-1003, 2003.