

Target Tracking and Data Fusion using Multiple Adaptive Foveal Sensors

Ya Xue and Darryl Morrell*
Department of Electrical Engineering
Arizona State University
Tempe AZ 85287-5706 USA
ya.xue@asu.edu, morrell@asu.edu

Abstract – *In this paper, multiple foveal sensors are used to track a target moving in three dimensions; the target tracker is implemented using particle filters. A foveal sensor’s field of view includes a high-sensitivity “foveal” region surrounded by a lower sensitivity region. The location and acuity of the foveal region are actively configured by the tracker using the estimated target position and spread of particles. Two tracker architectures are presented. In the joint resampling architecture, the observations from all sensors are used by a central control module to estimate the state; the central control module configures each sensor. In the individual resampling architecture, each sensor uses its own observations to configure itself and provide angular estimates to a central control module; the central control module estimates the target state using these estimates. The performance of both architectures is compared through Monte Carlo simulation.*

Keywords: Foveal sensor, particle filter, data fusion.

1 Introduction

One source of inspiration in the development of sensor systems is the study of biological sensory systems which have the ability to selectively gather information from the environment[1]. The human visual system is one such system. The density of optical cells in a human retina is not constant, as shown in Figure 1. The central high-density area is called the fovea and provides better spatial resolution than the surrounding low-density areas.

In this paper, we study the use of multiple foveal sensors to track a target moving in three dimensions. The structure of our foveal sensor model is inspired by the human visual system. A foveal sensor has a high-acuity foveal region surrounded by regions with much lower

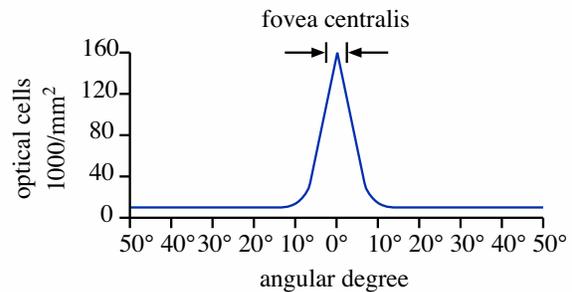


Figure 1: Density of optical cells in a human retina (from [2]).

acuity; the location and acuity of the foveal region are configured to collect observations of the target’s angular position.

We use particle filters to estimate the target position and velocity from the foveal sensor observations. We also investigate two system architectures to implement the tracker:

- A joint resampling scheme, in which data from all sensors is combined by a central control module to estimate the target state. Information from the central control module is used to configure each sensor.
- An individual resampling scheme, in which each sensor estimates the target state individually and uses this information to configure itself. Estimated target bearings from each sensor are used by the central control module to estimate the target state.

This work is an extension of previous work in which a single foveal sensor was used to track a target moving in one dimension[3].

*This work supported by AFOSR under award number F49620-00-1-0124.

This paper is structured as follows. In Section 2, we introduce the target dynamics and observation model and provide an overview of the particle filter. Section 3 introduces foveal sensors and presents an adaptive acuity algorithm for three-dimension tracking. The data fusion process of the distributed sensor system is discussed in Section 4. Simulation results are presented in Section 5, and conclusions are drawn and suggestions for further work are given in Section 6.

2 Particle Filter Formulation

The particle filter estimates the target state from observations obtained by foveal sensors and provides statistical information necessary to configure the foveal sensors. In earlier work with a fixed-acuity foveal sensor [12, 13], these functions were performed by a Kalman filter whose input data were subjected to significant statistical preprocessing. In prior work [3], we have shown that for the fixed-acuity sensor, the particle filter performs as well as the Kalman filter/preprocessor implementation. For the variable-acuity sensor used in [3] and in this paper, simulation results show that the particle filter performs significantly better than an Extended Kalman Filter (EKF); the particle filter can correctly compute the statistics resulting from the different acuities (gains) in the different regions of the foveal sensor.

2.1 Target and Observation Models

The target state at time k is a random vector X_k . The target is free to move in three dimensions; for each dimension, the target state consists of a position and a velocity. The target state evolution is modeled by the following linear system[4]:

$$X_{k+1} = AX_k + U_k, \quad (1)$$

where U_k is a vector Gaussian noise process with a mean of zero and a covariance matrix Σ_{U_k} . This dynamics model is Markov and can be represented by the conditional density $p(x_{k+1}|x_k)$.

At each discrete time k , a foveal sensor obtains a vector observation Y_k , which is a function of the target state X_k and observation noise W_k as described in Equations (5) and (6) in Section 3.2. The conditional density of Y_k given X_k is denoted by $p(y_k|x_k)$. We assume that W_k is Gaussian and additive, so that this conditional density is Gaussian. A sequence of state values is denoted $x_{1:k}$ and a sequence of observation values is denoted $y_{1:k}$.

Given the conditional densities $p(x_{k+1}|x_k)$ and $p(y_k|x_k)$, one can formulate a recursive Bayesian estimator of the target state sequence given the observation sequence. [5] provides a development of the recursion, as well as a description of optimal and sub-

optimal implementations for particular cases. These implementations include particle filters, the Kalman filter, extended Kalman filter, and grid-based filters; in this paper, we use particle filters to implement the target state estimator.

2.2 Particle Filters

With the increase in computer power over the last two decades, sequential Monte Carlo (MC) filters have become increasingly important. The sequential MC approach is known as particle filtering or bootstrap filtering. It implements an asymptotically optimal Bayesian filter to estimate the state of a system from a sequence of observations by using a set of random samples $\{x_k^{(i)}, i = 1, \dots, N_s\}$ with associated weights $\{\omega_k^{(i)}, i = 1, \dots, N_s\}$. The system dynamics and observation models can be non-linear, and the process and observation noise can be non-Gaussian. Particle filtering is described in detail in [5, 6, 7].

In this work, the particles (random samples) $\{x_{k-1}^{(i)}\}$ and associated weights $\{\omega_{k-1}^{(i)}\}$ approximate the posterior density $p(x_{k-1}|y_{1:k-1})$. The predicted density $p(x_k|y_{1:k-1})$ is approximated by sampling each particle using the state transition density $p(x_k^{(i)}|x_{k-1}^{(i)})$ as the proposal distribution. After the foveal sensor is configured and an observation y_k is obtained, the associated weights $\{\omega_k^{(i)}\}$ are updated using

$$\omega_k^{(i)} \propto \omega_{k-1}^{(i)} p(y_k|x_k^{(i)}). \quad (2)$$

The posterior density of X_k is approximated as

$$p(x_k|y_{1:k}) \approx \sum_{i=1}^{N_s} \omega_k^{(i)} \delta(x_k - x_k^{(i)}). \quad (3)$$

The system state estimate is the conditional mean of the posterior density, which is computed approximately using the particles and weights:

$$\hat{x}_k = E[X_k|y_{1:k}] \approx \sum_{i=1}^{N_s} \omega_k^{(i)} x_k^{(i)}. \quad (4)$$

This particle filter is simple to implement, but may suffer from *sample impoverishment* (when the resampled particles contain many repeated points), particularly when the observation noise variance is very small relative to the process noise variance. Sample impoverishment is a significant issue when configuring the acuity of the foveal sensor, since the width of the foveal region is set to include a given percentage of particles (as discussed in Section 3.3). We address this problem by adopting a Gaussian resampling method (fully described in [3]) that increases the diversity of particle

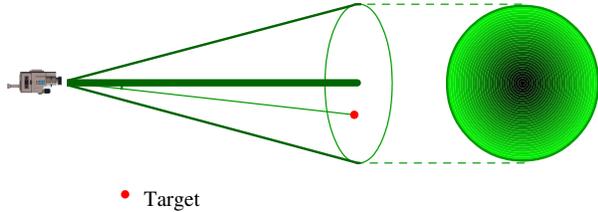


Figure 2: Angular foveal sensor with a high-sensitivity foveal region surrounded by a lower sensitivity region.

locations. In this method, the mean and covariance of the posterior density $p(x_k|y_{1:k})$ are computed from the particles and weights; the particles are then resampled from a Gaussian density with this mean and covariance. In practice, this resampling scheme has given improved performance compared to the resampling algorithm in [5].

The system model in (1) is linear with Gaussian noise, and the observations from the model defined in Section 3.2 depend only on the target position. Thus, the computational requirements of the particle filter are significantly reduced by using Rao-Blackwellization (see for example [17]).

3 Foveal Sensor

In this section, we provide some background on concepts and previous work related to foveal sensors, and then present the foveal sensor model used in our work.

3.1 Background

There have been significant efforts to emulate the structure and function of the human vision system in applications of automated tracking, robot vision, and surveillance. Most previous research on foveal sensors has concentrated on the development of image sensors with a foveal region and on tracking algorithms specifically tailored to these sensors. In 1990 Debusschere *et. al.* presented their work on a retinal CCD sensor for fast 2D shape recognition and tracking, which is the earliest designed using CMOS technology in the literature[8]. Wodnicki designed a foveated image sensor using a standard $1.2\mu\text{m}$ ASIC CMOS process in 1994[9]. Another foveated sensor using $0.7\mu\text{m}$ CMOS technology was designed to achieve a higher resolution at IMEC, Belgium in 1996[10]. For the application of advanced optical technology, Kuniyoshi developed an active stereo vision system with foveated wide angle lenses (special lenses with a space variant optical projection) in 1995[11].

Theoretical work investigating the performance of target tracking algorithms using a one-dimensional foveal sensor was initiated by Cochran *et. al.* [12, 13].

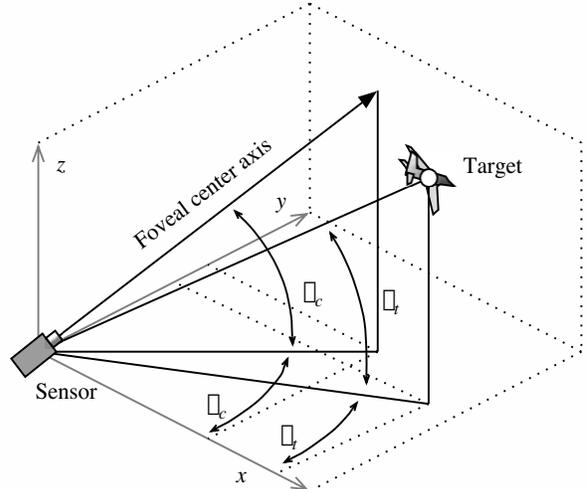


Figure 3: Coordinates and notation for the two-dimensional foveal sensor observations.

In this work, a target was constrained to move in one dimension. A foveal sensor provided measurements of position, which were used to estimate the target position and velocity. This work used a Kalman filter with various data pre-processing schemes to implement the tracker. This work was extended in [3] to a foveal sensor with adaptive acuity; a particle filter, rather than the Kalman-filter based estimator, was also used. The foveal sensor model developed in Section 3.2 is a two-dimensional extension of the one-dimensional model used in [3].

3.2 Mathematical Model

The two-dimensional foveal sensor measures the angles from the sensor axis to the target location. The foveal region of this sensor is illustrated in Figure 2. The foveal sensor has an conical foveal region. The foveal sensor can be configured by adjusting the orientation of the cone's axis. This foveal sensor provides highly accurate angle measurements when the angle between the axis and the target bearing is close to zero, and less accurate angle measurements when this angle is large. We illustrate the sensor accuracy in Figure 2 by the shading from dark to light, indicating decreasing visual acuity as the angle increases.

Let (θ_c, ϕ_c) denote the angular coordinates of the foveal center and (θ_t, ϕ_t) denote the coordinates of the target as shown in Figure 3. We define $\Delta\theta = \theta_t - \theta_c$ and $\Delta\phi = \phi_t - \phi_c$. We model the action of the foveal sensor as

$$Y_k = h(\Delta\theta, \Delta\phi) + W_k, \quad (5)$$

where Y_k is the vector of observed angles, h is a real-

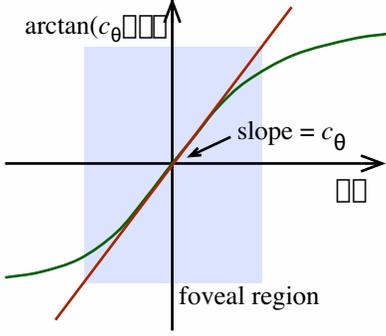


Figure 4: Observation function of the angular foveal sensor for $\Delta\theta$. This same observation function is used for $\Delta\phi$.

valued continuous function and W_k is a white Gaussian noise sequence that is independent of the target state and has mean zero and covariance Σ_{W_k} .

We choose the following functional form for h :

$$h(\Delta\theta, \Delta\phi) = \begin{bmatrix} \arctan(c_\theta \Delta\theta) \\ \arctan(c_\phi \Delta\phi) \end{bmatrix}. \quad (6)$$

Figure 4 plots a component of this function. This function models a foveal sensor, since it has a high gain (*ie.* derivative) when $\Delta\theta$ or $\Delta\phi$ are close to zero, and low gain when $\Delta\theta$ or $\Delta\phi$ are far from zero. The gain on each component of the observation is determined by the vector $c = [c_\theta \ c_\phi]$; the components of c can be adjusted independently. Large values of c_θ or c_ϕ result in a large gain at the cost of a small foveal region.

The foveal sensor is configured by choosing values for the four parameters θ_c , ϕ_c , c_θ , and c_ϕ . θ_c and ϕ_c determine the center of the foveal region; c_θ and c_ϕ determine the gain in the foveal region.

3.3 Adaptive Acuity Algorithm

In previous work[13, 12], the gain of the foveal sensor was fixed and only the center of the foveal region could be configured. In [3] and this paper, we investigate the performance obtainable by configuring both the center and gain of the foveal region. The foveal region is centered at the predicted angular position of the target. We now describe the algorithm used to adjust the gains.

The particle filter, by computing samples from an approximation to the posterior density, provides the information necessary to set the width of the foveal region to include likely target observations. In [3], the foveal region for the one-dimensional tracking problem was defined as the region where the slope of the observation function is above a threshold γ . The gain was

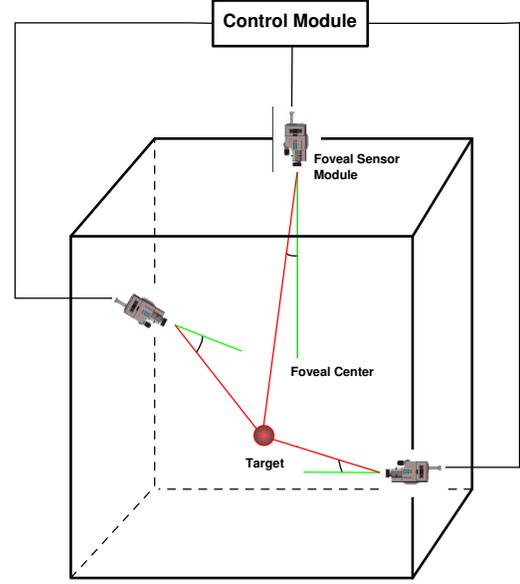


Figure 5: A sensor system composed of multiple foveal angle sensors and a control module.

chosen so that this region includes a fraction of $1 - \alpha$ of the particles, where α is a parameter that can be chosen between zero and one.

We extend this definition of the foveal region to the two-dimensional angular foveal sensor. Let $\Delta\theta_k^{(i)}$ and $\Delta\phi_k^{(i)}$ be the angles measured by the foveal sensor to the i th particle $x_k^{(i)}$. Let $F_\theta(\cdot)$ and $F_\phi(\cdot)$ denote the empirical cumulative distribution functions of $\Delta\theta_k^{(i)}$ and $\Delta\phi_k^{(i)}$, $i = 0, \dots, N_s$. Denote the $1 - \alpha$ percentile value of $\Delta\theta_k$ as $\Delta\theta_{k,1-\alpha}$ and of $\Delta\phi_k$ as $\Delta\phi_{k,1-\alpha}$; in other words,

$$\begin{aligned} F_\theta(\Delta\theta_{k,1-\alpha}) &= 1 - \alpha \\ F_\phi(\Delta\phi_{k,1-\alpha}) &= 1 - \alpha \end{aligned}. \quad (7)$$

Given the slope threshold γ for the foveal region, the instantaneous sensor gain $\hat{c}_k = [\hat{c}_{\theta,k} \ \hat{c}_{\phi,k}]^T$ is the real valued solution of [3]:

$$\begin{aligned} \gamma \Delta\theta_{k,1-\alpha} \hat{c}_{\theta,k}^2 - \hat{c}_{\theta,k} + \gamma &= 0 \\ \gamma \Delta\phi_{k,1-\alpha} \hat{c}_{\phi,k}^2 - \hat{c}_{\phi,k} + \gamma &= 0 \end{aligned}. \quad (8)$$

A decaying moving average filter is used to reduce the effect of significant errors in \hat{c}_k caused by occasional large measurement noise values. The instantaneous gain is computed using (8) then averaged with previous values:

$$c_k = \xi \hat{c}_k + (1 - \xi) c_{k-1},$$

where ξ is a weighting factor between one and zero.

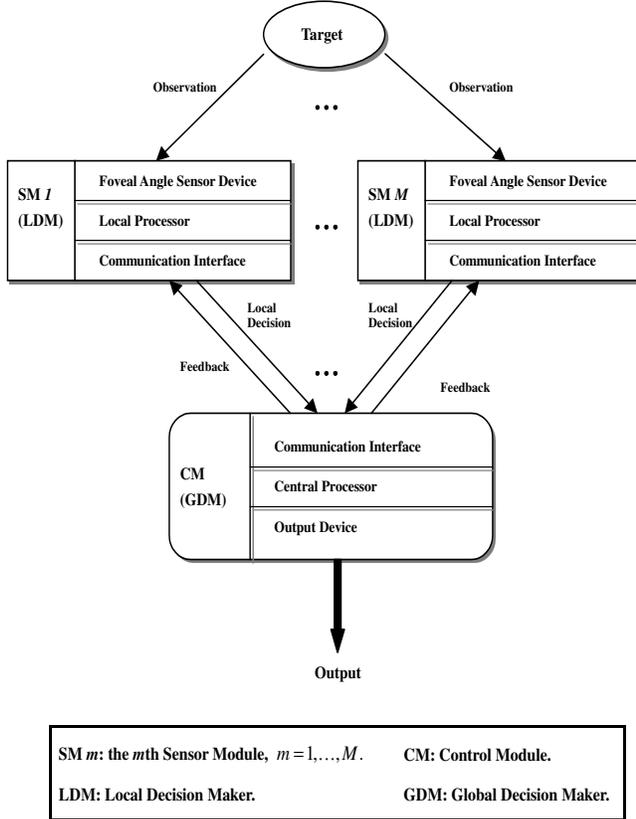


Figure 6: Distributed decision network architecture.

4 Data Fusion in a System of Distributed Foveal Sensors

The observations from a single angular sensor are typically insufficient to track a target moving in three dimensions. Thus, we consider a system of distributed foveal sensors similar to that shown in Figure 5; the number of sensors in the system is denoted M .

This multiple sensor system comprises a control module and multiple sensor modules, as shown in Figure 6. Each sensor module is composed of three parts: the foveal sensor, a local processor and a communication interface; these parts perform the functions of observing the target, making local sensor configuration and tracking decisions, and transmitting the decisions to the control module. The control module is a global decision maker which estimates the target location based on information supplied by the sensor modules; the control module also feeds back information to the sensors. The sensors may adapt their configurations based on this feedback information. This type of distributed detection network was categorized as a parallel fusion network topology with feedback in [14]. This architecture can also be seen as a self-improving fusion system architecture[15].

At each time epoch k :	
Sequential action	Done by
<ul style="list-style-type: none"> • Transmit the updated particles $\{x_k^{(i)}, i = 0, \dots, N_s\}$ to each sensor module; 	Control Module
<ul style="list-style-type: none"> • Using the particles, adjust the foveal gains $c_k^m, m = 1, \dots, M$, using the algorithm in Section 3.3; • Obtain the observations $Y_k^m = [y_{\theta,k}^m \ y_{\phi,k}^m]^T, m = 1, \dots, M$; • Calculate the local weights $\{\omega_k^{(i),m}, i = 0, \dots, N_s, m = 1, \dots, M\}$ by substituting Y_k^m into (2); • Transmit the local weights $\{\omega_k^{(i),m}, i = 0, \dots, N_s, m = 1, \dots, M\}$ to the control module; 	Sensor Module
<ul style="list-style-type: none"> • Calculate the global weights $\{\omega_k^{(i)} = \prod_{m=1}^M \omega_k^{(i),m}, i = 0, \dots, N_s\}$; • Estimate the location of the target by (4); • Resample. 	Control Module

Table 1: Operations performed by the control module and the sensor modules in one iteration of the joint resampling scheme.

Data fusion in such a distributed sensor system is a research area of significant interest. Frameworks for data fusion, mainly based on probabilistic models or least-squares techniques, are well known, *eg.* [14, 16]; however, specific algorithms must be designed for each specific system. In this work, two different approaches have been developed: a joint resampling scheme, and an individual resampling scheme. In the joint resampling scheme, data fusion is performed using particle weights from each sensor. In the individual resampling scheme, each sensor uses a particle filter to estimate the angular measurements, which are then transmitted to the control module.

In the joint resampling scheme, the control module implements a particle filter; the particles from this filter are communicated to each sensor, where they are

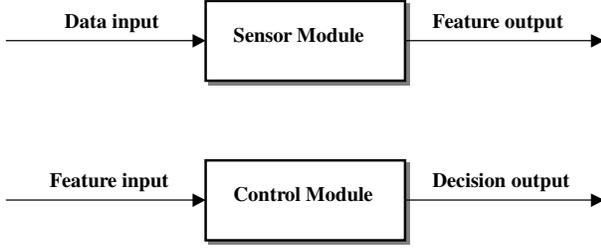


Figure 7: Fusion process I/O characteristic of joint resampling scheme.

used to configure the sensor location and gain as described in Section 3.3. After configuration, the sensor computes particle weights, which are then transmitted to the control module and used (jointly with all sensor weights) to update the particles in the particle filter. This scheme is described in Table 1. The drawback of this method is the heavy communication burden, and thus large bandwidth requirement, of transmitting samples and weights.

The individual resampling scheme is designed to reduce the amount of information that must be communicated between the control and sensor modules. In this scheme, each sensor individually implements a particle filter; each sensor is configured using the particles in its own filter. In addition, from these particles, each sensor module estimates the angles $(\Delta\hat{\theta}_k^m, \Delta\hat{\phi}_k^m)$, where m is the index of the sensor, which are then communicated to the control module. The control module finds as its estimate the point \hat{x}_k

$$\hat{x}_k = \arg \min_x \sum_{m=1}^M |d(x)_k^m|, \quad (9)$$

where $d(x)_k^m$ is the distance between the point x and the beam originating at the location of sensor m and pointing in the angular direction $(\theta_{c,k}^m + \Delta\theta_k^m, \phi_{c,k}^m + \Delta\phi_k^m)$. This scheme requires significantly less communication than the joint resampling scheme.

The I/O characteristics of the fusion process are different for these two resampling schemes. Figure 7 shows the I/O characteristics of the joint resampling scheme using the characterization developed in [15]; Figure 8 shows the I/O characteristics of the individual resampling scheme. In Figure 8, the fusion process in the sensor module is a combination of Data In - Feature Out (DAI-FEO) fusion mode and Feature In - Decision Out (FEI-DEO) fusion mode.

5 Simulation Results

The performance of this tracking system has been evaluated using Monte Carlo simulations. The dynam-

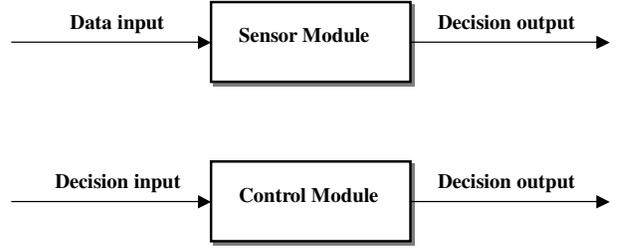


Figure 8: Fusion process I/O characteristic of individual resampling scheme.

ics matrix A used in the simulations is

$$A = \begin{bmatrix} 1 & \Lambda_k & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \Lambda_k & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \Lambda_k \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

This matrix models target motion in three dimensions with a time between observations of $\Lambda_k = t_{k+1} - t_k$. The covariance of the process noise U_k is

$$\Sigma_{U_k} = Q_k \begin{bmatrix} \frac{\Lambda_k^3}{3} & \frac{\Lambda_k^2}{2} & 0 & 0 & 0 & 0 \\ \frac{\Lambda_k^2}{2} & \Lambda_k & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\Lambda_k^3}{3} & \frac{\Lambda_k^2}{2} & 0 & 0 \\ 0 & 0 & \frac{\Lambda_k^2}{2} & \Lambda_k & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\Lambda_k^3}{3} & \frac{\Lambda_k^2}{2} \\ 0 & 0 & 0 & 0 & \frac{\Lambda_k^2}{2} & \Lambda_k \end{bmatrix} \quad (11)$$

where Q_k is the intensity of a continuous-time white noise process modeling the instantaneous target acceleration. More details about this model can be found in [4].

We used the simulation to investigate the performance differences between the joint and individual resampling schemes. We also investigated the tracker performance as a function of the number of foveal sensors.

5.1 Joint Versus Individual Resampling

The performance of the joint and individual resampling schemes was evaluated using Monte Carlo simulation and three foveal sensors. In all simulations, the time between observations was $\Lambda_k = 0.1$, the intensity of the process noise was $Q_k = 0.5$, and the covariance matrix of the observation noise was $\Sigma_{W_k} = \text{diag}(0.1, 0.1)$. Each simulation was run for 50 epochs (observations).

Three different sets of 100 Monte Carlo simulation runs were created using 200 particles in each run; in

	Joint Resampling	Individual Resampling
Set 1	0.0625	0.5338
Set 2	0.0596	0.1853
Set 3	0.0560	0.1210

Table 2: Average squared position estimation error for the three sets of simulation runs. In Set 1, the sensors were closest to the origin, and in Set 3, they were far away from the origin.

	Joint Resampling	Individual Resampling
$N_s = 200$	1400	5
$N_s = 1000$	7000	5

Table 3: Number of values per sensor that must be transmitted for the joint and individual resampling for 200 and 1000 particles.

each run, the sensor positions were chosen randomly. For the first set of Monte Carlo runs, sensor positions were close to the origin; sensor positions were farther away for the second set, and even farther away for the third set. The relative directions from the origin to each of the three sensors was the same in all three sets of Monte Carlo runs. The squared position error was averaged over the 50 epoch duration of each run and over the 100 simulation runs in each set.

Table 2 shows the average squared position error for the two resampling schemes for each set of simulations. In each of the three cases, the average squared estimate error for the joint resampling was significantly lower than the average error for the individual resampling. Also, the system performance improves as the sensors are moved farther from the origin; this is because moving the sensors farther from the origin increases their angular separation, which decreases the errors in estimated position caused by errors in the observations.

We also investigated the effect of increasing the number of particles in the particle filters from 200 to 1000; this improves the performance of the joint resampling scheme significantly but does not improve the performance of the individual resampling scheme appreciably.

The improved performance of the joint resampling scheme comes at the price of a much higher communication cost. For the joint scheme, at each time epoch, the control module sends the sensors the information for each particle (all six components of the state vector). After the observation is obtained, each sensor sends the local weight for each particle back to the con-

Number of Sensors	Information Quantity	Average Mean-Square Error
2	Insufficient	0.5753
3	Sufficient	0.3359
4	Redundant	0.2858
5	Redundant	0.3157

Table 4: Comparison of the average mean-square estimation error of the tracker as a function of the number of sensors in the network. Results are averaged over 1000 simulation runs.

trol module. Thus, the total amount of data transmitted between one sensor module and the control module is approximately $7N_s$ floating point numbers, where N_s is the number of particles used. For the individual resampling scheme, each sensor module estimates the angles $(\Delta\theta_k^m, \Delta\phi_k^m)$, then sends these estimates to the control module. The three dimensional location of each sensor must also be sent if it is unknown to the control module; thus the total amount of data transmitted between one sensor module and the control module is only five floating point values. The ratio of communication burdens of the individual and joint resampling scheme is thus approximately $1 : 1.4N_s$. Table 3 illustrates the difference in communication costs between the joint and individual resampling scheme for $N_s = 200$ particles and $N_s = 1000$ particles. Therefore, when N_s is large, it may be worth paying the performance penalty in estimate error to use the individual resampling scheme.

5.2 Number of Sensors Versus Performance

We investigated how the number of sensors affects the performance of the target tracker. Clearly, the performance of the sensor network depends on the location of the sensors as well as the number of sensors. Table 4 shows the performance of sensor networks composed of two to five foveal sensors for a particular sensor placement; the joint resampling scheme with 100 particles is used, and the squared error is averaged over 1000 simulation runs. Two sensors are insufficient to locate the target from a single set of observations; three sensors provide enough information to locate the target from a single set of observations, and four or five sensors provide redundant information. The results show that increasing the number of sensors improves the estimate quality except in the case of five sensors, where the addition of the fifth sensor does not improve the estimate quality.

6 Conclusions and Future Work

In this paper, we have presented a target tracking system based on multiple foveal sensors. We have presented a centralized and a decentralized system architecture to configure the foveal sensors and estimate the target state. The centralized architecture provides better performance at the cost of significant information communication.

We have identified several areas for further work:

- *Alternative system architectures:* We have investigated only two of the many possible system architectures. It is possible that other architectures will provide useful cost/performance trade offs. For example, it is likely that improved state estimates could be obtained from the individual resampling architecture by the addition of a particle filter at the control node that estimates the system state based on the estimated angular values supplied by the sensors.
- *Sensor selection strategy:* When sensing resources are redundant, the required communication bandwidth can be significantly reduced without degrading the system performance by selecting the most appropriate sub-set of sensors from among the sensors available to the system. Finding computationally tractable centralized and distributed algorithms for sensor selection is an area of ongoing research.

References

- [1] D. Sinno, *Attentive Management of Configurable Sensor Systems*, Ph.D. thesis, Arizona State University, May 2000.
- [2] U. Strohbeck, *A new approach in image data compression by multiple resolution frame-processing*, Ph.D. thesis, University of Northumbria at Newcastle upon Tyne / U.K., 1999.
- [3] Y. Xue, and D. Morrell, "Adaptive Foveal Sensor for Target Tracking", *36th Asilomar Conference on Signals, Systems, and Computers*, Monterey, CA, Nov. 2002, pp. 848–852.
- [4] F. Lewis, *Optimal Estimation with an Introduction to Stochastic Control Theory*, Wiley, 1986.
- [5] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for On-line Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Trans. on Sig. Proc.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
- [6] A. Doucet, S.J. Godsill, and C. Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering", *Statistics and Computing*, 10(3), 2000.
- [7] A. Doucet, N.D. Freitas, and N. Gordon *Sequential Monte Carlo Methods in Practice*, New York : Springer, 2001.
- [8] I. Debusschere, E. Bronckaers, C. Claeys, C. Kreider, J. Van der Spiegel, G. Sandini, P. Dario, F. Fantini, B. Bellutti, and G. Soncini, "A Retinal CCD Sensor for Fast 2D Shape Recognition and Tracking", *Sensors and Actuators*, pp. 456–460, 1990.
- [9] R. Wodnicki, G. W. Roberts, and Martin D. Levine, "A Foveated Image Sensor in Standard CMOS Technology", *Custom Integrated Circuits Conference*, Santa Clara, California, May 1995.
- [10] F. Pardo, J.A. Boluda, J.J. Pérez, S. Felici, B. Dierickx and D. Scheffer, "Response Properties of a Foveated Space-Variant CMOS Image Sensor", *1996 IEEE International Symposium on Circuits And Systems*, pp. 373–376, May 1996.
- [11] Y. Kuniyoshi, N. Kita, S. Rougeaux, and T. Suehiro, "Active stereo vision system with foveated wide angle lenses", *Proc. of 2nd Asian Conf. on Computer Vision*, Vol. I, pp. 359–363, 1995.
- [12] D. Cochran, and R. Martin, "Nonlinear Filtering Models of Attentive Vision" *IEEE International Symposium on Circuits and Systems* pp. 26–29, 1996.
- [13] L. Li, D. Cochran, and R. Martin, "Target Tracking with an Attentive Foveal Sensor", *34th Asilomar Conference on Signals, Systems and Computers*, pp. 182–185, 2000.
- [14] P. K. Varshney, *Distributed Detection and Data Fusion*, New York : Springer, 1997.
- [15] B.V. Dasarathy, "Intelligent fusion learning techniques in multi-source information fusion environments", *37th IEEE Conference on Decision and Control*, pp. 221–226, 1998.
- [16] J. Manyika and H. Durrant-Whyte, *Data Fusion and Sensor Management: A decentralized information theoretic approach*, Ellis Horwood, 1994.
- [17] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for Positioning, Navigation, and Tracking", *IEEE Trans. on Sig. Proc.*, vol. 50, no. 2, pp. 425–437, Feb. 2002.