

Dynamic Agent Classification and Tracking Using an Ad Hoc Mobile Acoustic Sensor Network

David Friedlander

*Applied Research Laboratory, The Pennsylvania State University, P.O. Box 30, State College, PA 16801-0030, USA
Email: dsf10@psu.edu*

Christopher Griffin

*Applied Research Laboratory, The Pennsylvania State University, P.O. Box 30, State College, PA 16801-0030, USA
Email: cgriffin@psu.edu*

Noah Jacobson

*Applied Research Laboratory, The Pennsylvania State University, P.O. Box 30, State College, PA 16801-0030, USA
Email: ncj102@psu.edu*

Shashi Phoha

*Applied Research Laboratory, The Pennsylvania State University, P.O. Box 30, State College, PA 16801-0030, USA
Email: sxp26@psu.edu*

Richard R. Brooks

*Applied Research Laboratory, The Pennsylvania State University, P.O. Box 30, State College, PA 16801-0030, USA
Email: rrb5@psu.edu*

Received 12 December 2001 and in revised form 5 October 2002

Autonomous networks of sensor platforms can be designed to interact in dynamic and noisy environments to determine the occurrence of specified transient events that define the dynamic process of interest. For example, a sensor network may be used for battlefield surveillance with the purpose of detecting, identifying, and tracking enemy activity. When the number of nodes is large, human oversight and control of low-level operations is not feasible. Coordination and self-organization of multiple autonomous nodes is necessary to maintain connectivity and sensor coverage and to combine information for better understanding the dynamics of the environment. Resource conservation requires adaptive clustering in the vicinity of the event. This paper presents methods for dynamic distributed signal processing using an ad hoc mobile network of microsensors to detect, identify, and track targets in noisy environments. They seamlessly integrate data from fixed and mobile platforms and dynamically organize platforms into clusters to process local data along the trajectory of the targets. Local analysis of sensor data is used to determine a set of target attribute values and classify the target. Sensor data from a field test in the Marine base at Twentynine Palms, Calif, was analyzed using the techniques described in this paper. The results were compared to “ground truth” data obtained from GPS receivers on the vehicles.

Keywords and phrases: sensor networks, distributed computing, target tracking, target identification, self-organizing systems.

1. INTRODUCTION

Distributed sensing systems combine observations from a large area network of sensors, creating the need for platform self-organization and the sharing of sensor information between platforms. It is difficult to integrate the data from each sensor into a single context for the entire network. Instead, groups of sensors in local areas collaborate to produce useful information to the end user.

Our objective is to create a distributed wireless network of sensors covering large areas to obtain an accurate representation of dynamic processes occurring within the region. Such networks are subject to severe bandwidth limitations and power constraints. Additionally, we need to integrate data from heterogeneous sensors.

Our goals are met through algorithms that determine the characteristics of the target from local sensor data. They dynamically cluster platforms into space-time neighborhoods

and exchange target information within neighborhoods to determine target class and track characteristics. This differs from other methods of decentralized detection such as [1, 2] where the dimensionality of the sensor data vectors is reduced to the distinct number of target attributes. Once organized into clusters, sensors can combine their local knowledge to construct a representation of the world around them. This information can be used to construct a history of the dynamic process as it occurs in the sensor field [3].

Our analysis is based on the concepts of a *space-time neighborhood*, a *dynamic window*, and an *event*. A space-time neighborhood centered on the space-time point (\bar{x}_0, t_0) is the set of space-time points

$$N(\bar{x}, t) \equiv \{(\bar{x}, t) : |\bar{x} - \bar{x}_0| \leq \Delta x, |t - t_0| \leq \Delta t\}. \quad (1)$$

The quantities Δx and Δt define the size of the neighborhood. The space-time window contains all the data that was measured within a distance Δx around \bar{x}_0 and within the time interval $t_0 \pm \Delta t$.

We can define a dynamic window around a moving point $\bar{g}(t)$ as

$$\omega(t) = \{(\bar{x}, t) : |\bar{x} - \bar{g}(t_0)| \leq \Delta x, |t - t_0| \leq \Delta t\}. \quad (2)$$

Ideally, if $\bar{g}(t)$ were the trajectory of the target, we would analyze time-series data from sensors in the window $N_e = \omega(t_e)$ to determine information about the target at time t_e .

The target trajectory $\bar{g}(t)$ is unknown. It is, in fact, what we want to determine. We therefore look at *closest-point-of-approach* (CPA) events that occur within a single space-time neighborhood. A CPA event e_{ij} is defined for platform i occurring at the CPA time t_j . The space-time coordinates of the event are $(\bar{x}_i(t_j), t_j)$, where $\bar{x}_i(t)$ is the trajectory of platform i .

We make the assumption that sensor energy increases as distance from the source decreases. This is a reasonable assumption for acoustic and seismic sensors. The CPA event is therefore assumed to occur when there is a peak in sensor energy. The amplitude of the event a_{ij} is defined as the amplitude of the corresponding peak. In order to filter out noise, reflection, or other spurious features, we count only peaks above a threshold and do not allow two events on a single platform within the same space-time window. If data from multiple sensors are available, they must be integrated to determine a single peak time for the event.

For an event e_{ij} , we analyze data from platforms in the neighborhood $N(\bar{x}_i(t_j), t_j)$. We define the set of platforms that contain events in this space-time neighborhood as the *cluster* of platforms associated with event e_{ij} . These definitions apply to both stationary and moving platforms and seamlessly integrate both types. They can be used to determine target velocity as long as the platform trajectories are known and the platform speed is small compared to the propagation speed of the energy field measured by the sensors. Platform locations can be determined by GPS and, for stationary platforms, additional accuracy can be achieved by integrating GPS signals over time.

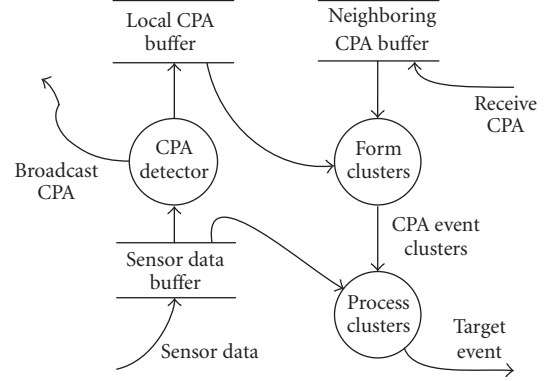


FIGURE 1: System overview.

The sets of parameters needed to identify targets are called *target events*. They include \bar{x}_i : the target position, t_i : the time, \bar{v}_i : the target velocity, and $\{a_1 \cdots a_n\}$: a set of target attributes for target classification, which can be determined from the sensor data in a region around the space-time point (\bar{x}_i, t_i) . A CPA event is detected by a platform when the target reaches its CPA to the platform. Each CPA will correspond a peak in the readings of our acoustic sensors. We have developed an algorithm that limits data processing to the platforms closest to the trajectory of the target rather than processing each CPA event. It evenly spreads the processing out over the space-time range of the target trajectory. All the platforms within the neighborhood of an event are assumed to be capable of communicating with each other.

The remainder of this paper is divided as follows. Section 2 discusses the algorithm for platform clustering. Section 3 discusses our velocity and position estimation algorithm. Section 4 discusses our approach to target identification. Section 5 provides both simulated and real-world experimental data that show that our approach produces promising results for velocity approximation and target recognition. Finally, Section 6 discusses our conclusions.

2. ALGORITHM FOR EVENT CLUSTERING

Nodes located within a given space-time window can form a cluster. Both the time and spatial extent of the window are currently held constant. The maximum possible spatial size of the window is constrained by the transmission range of the sensors. Each node contains a buffer for its own CPA events, and a buffer for CPA events transmitted by its neighbors. Figure 1 shows a simple diagram depicting the system running in parallel on each platform.

The *CPA detector* looks for peaks in sensor energy as described in Section 1. When it finds one, it stores the amplitude, time, and platform position in a buffer, and broadcasts the same information to its neighbors. When it receives neighboring CPA events, it stores them in another buffer. The *form clusters* routine looks at both CPA event buffers, and forms event clusters as shown in Figure 1. The *process*

```

For each local CPA event  $k_{ij} = k(x_i, t_j)$ 
  For each neighboring CPA event  $n_{kl} = n(x_l, t_k)$ 
    If  $n_{kl}$  is in the neighborhood  $N_{ij} = N(\bar{x}_i, t_j)$ 
      Add  $n_{kl}$  to the event set  $M$ 
  If the local peak amplitude  $a(k_{ij}) \geq a(n_{kl}) \forall n_{kl} \in M$ 
    Emit CPA event cluster  $F \equiv k_{ij} \cup M$ 

```

ALGORITHM 1: Form clusters pseudocode.

clusters routine determines the target position and velocity as described in Section 3 and the target attributes as described in Section 4.

3. VELOCITY AND POSITION ESTIMATION ALGORITHM

Models of human perception of motion may be based on the spatio-temporal distribution of energy detected through vision [4, 5]. Similarly, the network detects motion through the spatio-temporal distribution of sensor energy.

We extend techniques found in [6] and adapt them to find accurate vehicle velocity estimates from acoustic sensor signals. The definitions shown below are for time and two spatial dimensions $\bar{x} = (x, y)$; however, their extension to three spatial dimensions is straightforward.

The platform location data from the CPA event cluster can be organized into the following sets of observations:

$$\begin{aligned} & (x_0, 0), (x_1, t_1) \cdots (x_n, t_n), \\ & (y_0, 0), (y_1, t_1) \cdots (y_n, t_n), \end{aligned} \quad (3)$$

where (x_0, y_0) is the location of event k_{ij} (see Figure 1), which contains the largest amplitude CPA peak in the cluster. We redefine the times in the observations, so $t_0 = 0$ where t_0 is the time of CPA event k_{ij} .

We weighted the observations based on the CPA peak amplitudes on the assumption that CPA times are more accurate when the target passes closer to the sensor to give

$$\begin{aligned} & (x_0, t_0, w_0), (x_1, t_1, w_1) \cdots (x_n, t_n, w_n), \\ & (y_0, t_0, w_0), (y_1, t_1, w_1) \cdots (y_n, t_n, w_n), \end{aligned} \quad (4)$$

where w_i is the weight of the i th event in the cluster. This greatly improved the quality of the predicted velocities. We defined the spatial extent of the neighborhoods, so nodes do not span more than a few square meters and vehicle velocities are approximately linear [6]. Under these assumptions, we can apply least square linear regression to obtain the following equations [7]:

$$x(t) = v_x t + c_1, \quad y(t) = v_y t + c_2, \quad (5)$$

```

Input: Time-sorted event cluster  $F$  of CPA values.
Output: Estimated velocity components  $v_x$  and  $v_y$ .
While  $|F| \geq 5$  {
  Compute  $v_x$  and  $v_y$  using event cluster  $F$ ;
  Compute  $r_x$  and  $r_y$ ; the  $v_x$  and  $v_y$  velocity
    ; correlation coefficients for  $F$ 
  If  $r_x > R_x \parallel r_y > R_y$ 
    {
       $R_x = r_x$ ;
       $R_y = r_y$ ;
       $v_{x\_store} = v_x$ ;
       $v_{y\_store} = v_y$ ;
    }
  PopBack( $F$ );
};

```

ALGORITHM 2

where:

$$\begin{aligned} v_x &= \frac{(\sum_i t_i)(\sum_i x_i) - (\sum_i w_i)(\sum_i x_i t_i)}{(\sum_i t_i)^2 - (\sum_i w_i)(\sum_i t_i^2)}, \\ v_y &= \frac{(\sum_i t_i)(\sum_i y_i) - (\sum_i w_i)(\sum_i y_i t_i)}{(\sum_i t_i)^2 - (\sum_i w_i)(\sum_i t_i^2)}, \end{aligned} \quad (6)$$

and the position $\bar{x}(t_0) = (c_1, c_2)$. The space-time coordinates of the target for this event are $(\bar{x}(t_0), t_0)$.

This simple technique can be augmented to ensure that changes in the vehicle trajectory do not degrade the quality of the estimated track. The correlation coefficients for the velocities in each spatial dimension (r_x, r_y) can be used to identify large changes in vehicle direction and thus limit the CPA event cluster to include only those nodes that will best estimate local velocity. Assume that the observations are sorted as follows:

$$O_i < O_j \longrightarrow |t_i - t_0| < |t_j - t_0|, \quad (7)$$

where O_i is an observation containing a time, location, and weight and t_0 is the time of the event k_{ij} . The velocity elements are computed once with the entire event set. After this, the final elements of the list are removed and the velocity is recomputed. This process is repeated while at least five CPAs are present in the set and subsequently the event subset with the highest velocity correlation is used to determine velocity. Fewer than five CPA points could severely bias the computed velocity and thus render our approximation useless. Algorithm 2 summarizes our technique.

4. TARGET CLASSIFICATION

The sounds a vehicle produces are a combination of the acoustic features of its components: its acoustic ‘‘fingerprints.’’ We have developed an algorithm to identify the presence or absence of given features in a target vehicle traveling through a sensor network. Once the vehicle type is

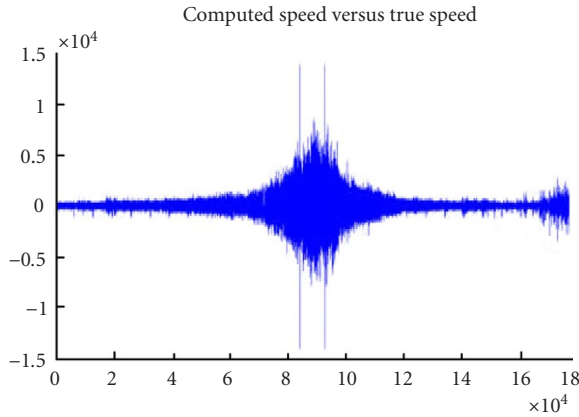


FIGURE 2: Time series window.

determined, it is combined with velocity and position data and broadcast over the network as a *target event*. This requires much less bandwidth than transmitting the original time series data.

The singular value decomposition (SVD) [8] is a matrix decomposition that can be used to find relationships within sets of data. When used to construct relationships between words and documents, this technique is called latent semantic analysis (LSA). There is significant evidence that LSA can be used to allow machines to learn words at a rate comparable to that of school children [9]. LSA accomplishes this by using SVD to infer relationships among members of a data set. We believe that this concept can be applied to vehicle identification.

Our identification algorithm combines Latent Semantic Analysis [9] with Principal Component Analysis [10, 11] to fuse semantic attributes and sensor data for target classification. There are two algorithms: *data processing* and *data classification*. CPA event data are divided into *training* and *test* sets. The training data are used with the data processing algorithm and the test data are used with the data classification algorithm to evaluate the accuracy of the method.

The training set is further divided into databases for each possible value of each target attribute being used in the classification. Target attribute values can be used to construct feature vectors for use in pattern classification. Alternatively, we can define “vehicle type” as a single attribute and identify the target directly.

A 4- to 5-second window is selected around the peak of each sample. All data outside the window is discarded. This ensures that noise bias is reduced. The two long vertical lines in Figure 2 show what the boundaries of the window would be on a typical sample.

The window corresponds to the period of time when a vehicle was closest to the platform. The data are divided into consecutive frames. A frame is 512 data points sampled at 5 kHz (0.5 seconds in length) and has a 12.5% overlap (0.07 second) with each of its neighbors. The power spectral density of each frame is found and stored as a column vector of 513 data points (grouped by originating sample) with data

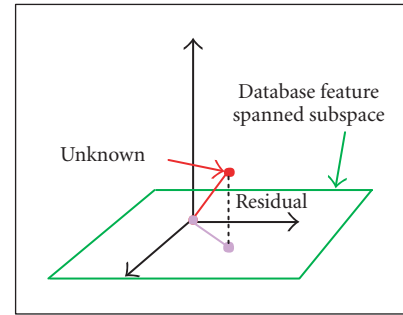


FIGURE 3: Isolating qualities in the feature space.

TABLE 1: Quality of estimation.

Computed versus true velocity	Percent
Percent within 1 m/s	81%
Percent within 2 m/s	91%
Percent within 5 degrees	64%
Percent within 11 degrees	80%
Percent within 17 degrees	86%

points corresponding to frequencies from 0 to 512 Hz.

Target identification combines techniques from [11] and makes use of an eigenvalue analysis to give an indication of the distance that an unknown sample vector is from the feature space of each database. This indication is called a residual. These residuals can be interpreted as “a measurement of the likelihood” that the frame being tested belongs to the class of vehicles represented by the database [11]. The databases are grouped by attribute and the residuals of each frame within each group are compared. The attribute value corresponding to the smallest total of the residuals within each group is assigned to the frame. Figure 3 illustrates this process.

5. EXPERIMENTAL RESULTS

We present two sets of results. Each demonstrates the quality of our techniques for estimating vehicle velocity in a distributed sensor field and identifying target characteristics. The result set comes from data collected at Twentynine Palms Marine Base during a field test and also from ideal data constructed in the lab for testing the velocity estimation algorithm.

5.1. Velocity estimation

We present a verification of our clustering and velocity estimation algorithms using data gathered at Twentynine Palms Marine base located in California. A sensor grid was tested there in August 2000.

We have analyzed the quality of our velocity estimation algorithm using our field data and these results appear in Table 1.

TABLE 2: Classification.

Actual vehicle	Classified numbers			Percent correctly classified
	AAV	DW	HV	
AAV	117	4	7	94%
DW	0	106	2	98%
HV	0	7	117	94%

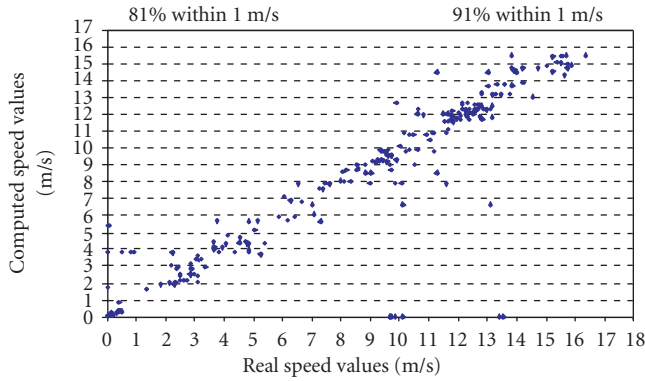


FIGURE 4: Computed speed versus true speed (field test).

Figures 4 and 5 show plots displaying the quality of the estimations.

We have also generated a simulated data set for testing our velocity algorithm. The data set was generated using a parabolic vehicle motion. Figure 6 shows activated sensors as the simulated vehicle passed through a dense grid of pseudorandomly distributed sensor platforms. Figure 7 displays the results of our algorithm for vehicle speed.

The calculated vehicle speeds yielded a correlation of 0.99 against a line of $y = 0.99x$, where y is the calculated speed and x is the simulated speed. The angle match is also extremely close.

5.2. Target identification verification

ARL evaluated its classification algorithms against the data collected during the field test. Data are shown for three types of military vehicles labeled AAV, DW, and HV. The CPA peaks were selected by hand rather than automatically detected by the software and there was only a single vehicle present in the network at a time. Environmental noise due to wind was significant. The data show that classification of military vehicles in the field can be accurate under noisy conditions, as shown in Table 2.

6. CONCLUSIONS

We have derived algorithms for target analysis that can identify target attributes using time-series data from platform sensors.

We have described an effective algorithm for computing target velocity. This velocity is critical for track formation

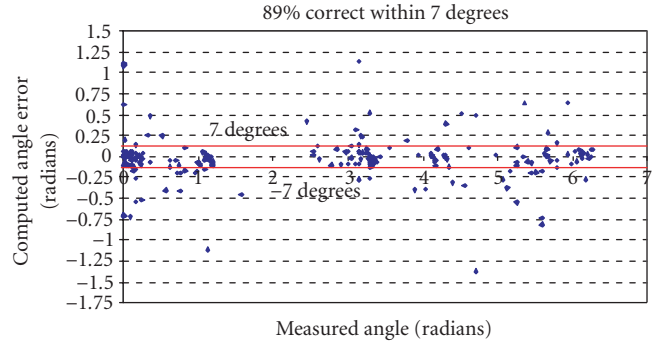


FIGURE 5: Computed angle versus true angle (field test).

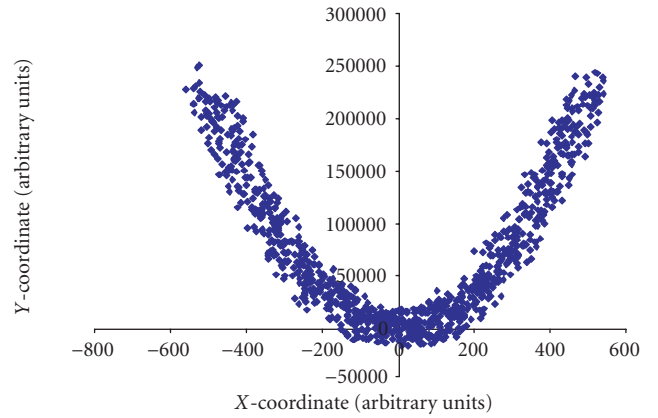


FIGURE 6: Simulated sensor node layout.

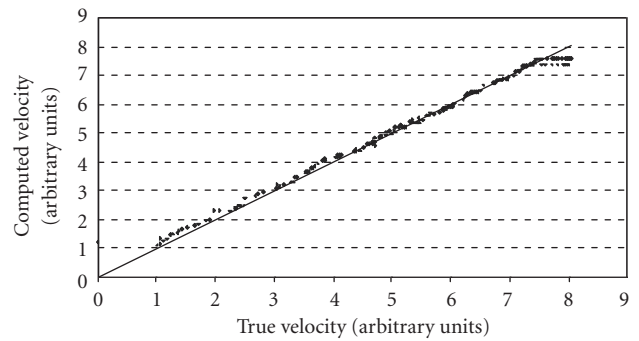


FIGURE 7: Computed speed versus true speed (simulation).

algorithms like those proposed in [3]. We have described an algorithm for accurate classification of military vehicles in the field.

We have also provided experimental verification of our procedures against field data using military vehicles and acoustic sensors. We have determined quantitative measures of the accuracy of the procedures.

Dense sensor networks over large areas contain massive amounts of computing power in total, but may be restricted

in bandwidth and power consumption at individual nodes. Forming dynamic clusters around events of interest allows processing multiple events in parallel over different local geographic areas. We have shown how networks can coordinate platforms around tracks and provide relevant processing with a minimum of bandwidth and power consumption related to interplatform communications. This procedure is scalable and takes full advantage of the parallelism in the network. The same algorithms run in parallel on each platform, making the procedure robust with respect to the loss of individual platforms. In addition, our method allows seamless integration of fixed and mobile heterogeneous platforms.

ACKNOWLEDGMENTS

This material is based upon work supported by the US Army Robert Morris Acquisition under Award No. DAAD19-01-1-0504. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the Army.

REFERENCES

- [1] B. Picinbono and M. P. Boyer, "A new approach of decentralized detection," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 1329–1332, 1991.
- [2] R. R. Tenney and N. R. Sandell Jr, "Detection with distributed sensors," *IEEE Trans. on Aerospace and Electronics Systems*, vol. 17, pp. 501–510, July 1981.
- [3] R. Brooks, C. Griffin, and D. S. Friedlander, "Self-organized distributed sensor network entity tracking," *International Journal of High Performance Computing Applications*, vol. 16, no. 3, pp. 207–219, 2002, Special Issue on Sensor Networks.
- [4] E. H. Adelson and J. R. Bergen, "Spatiotemporal energy models for the perception of motion," *Journal of the Optical Society of America {A}*, vol. 2, no. 2, pp. 284–299, 1985.
- [5] E. H. Adelson, "Mechanisms for motion perception," *Optics and Photonics News*, vol. 2, no. 8, pp. 24–30, 1991.
- [6] M. Hellebrant, R. Mathar, and M. Scheibenbogen, "Estimating position and velocity of mobiles in a cellular radio network," *IEEE Trans. Vehicular Technology*, vol. 46, no. 1, pp. 65–71, 1997.
- [7] W. H. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C*, Cambridge University Press, Cambridge, UK, 1992.
- [8] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, New York, NY, 1986.
- [9] T. K. Landauer and S. T. Dumais, "A solution to Platos problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge," *Psychological Review*, vol. 104, no. 2, pp. 211–240, 1997.
- [10] V. Bhatnagar, A. Shaw, and R. Williams, "Improved automatic target recognition using singular value decomposition," in *IEEE Trans. Acoustics, Speech, and Signal Processing*, Seattle, Wash, USA, 1998.
- [11] H. Wu, M. Siegel, and P. Khosla, "Vehicle sound signature recognition by frequency vector principal component analysis," in *IEEE Trans. Instrumentation and Measurement*, St. Paul, Minn, USA, May 1998.

David Friedlander is a Senior Research Engineer and Head of the Informatics Department of the Information Science and Technology Division of the Applied Research Laboratory at the Pennsylvania State University. His research includes formal languages, discrete-event control applied to command and control of military operations, and logistics for major industrial operations. He played a key role in developing and analyzing discrete-event control systems for the command and control of air campaigns. This includes the development of methods for analyzing the formal languages associated with finite state machines. He coauthored *The Scheduling of Rail at Union Pacific Railroad*, which won the Innovative Applications in Artificial Intelligence Award at the American Association for Artificial Intelligence in 1997. He researched methods for automating the development of Lexical Knowledge Bases. This included the use of latent semantic indexing (LSI) for automatically indexing an email corpus, and the use of hierarchical clustering of LSI indices for conceptual relationship discovery of the relationship between the intents of the email messages. He received the B.A. degree in physics and mathematics from New York University and received the M.A. degree in physics from Harvard University.



Christopher Griffin graduated with high distinction from the Pennsylvania State University in December of 2000 with a B.S. degree in mathematics. He is currently employed as an Assistant Research Engineer at the Pennsylvania State Applied Research Laboratory where his areas of research include high-level logical control, automated control systems, and systems modeling. Mr. Griffin is currently pursuing his master's degree in mathematics at the Pennsylvania State University.



Noah Jacobson is an undergraduate at the Pennsylvania State University, working towards majors in mathematics and computer engineering. He is doing research on acoustic sensor networks for vehicle tracking at the Information Science and Technology Division of Pennsylvania State Applied Research Laboratory. After receiving his B.S. degree, Mr. Jacobson is planning on to graduate school where he intends to earn a Ph.D. in computer vision.



Shashi Phoha is Professor of electrical engineering and Director of the Information Science and Technology Division of the Applied Research Laboratory at the Pennsylvania State University. She has led multi-organizational advanced research programs and laboratories in major US industrial and academic institutions. She pioneered the use of formal methods for the scientific analysis of distributed information for decision support, multistage coordination, and intelligent control of complex dynamic systems. She formulated the concept of information-based fault prognosis and maintenance planning over the National Information Infrastructure derived from online physics-based analysis of emerging damage. She has established



in situ analysis of correlated time-series data collected by a self-organizing sensor network of undersea robotic vehicles. She is the Principal Investigator for the Surveillance Sensor Networks MURI funded by DARPA, and the Project Director of the Complex Systems Failures MURI funded by the ARO. Dr. Phoha received her M.S. degree in 1973 from Cornell University and Ph.D. degree in 1976 from Michigan State. She is an Associate Editor of IEEE Transaction on Systems, Man, and Cybernetics. Dr. Phoha chaired the Springer-Verlag Technical Advisory Board for the *Dictionary of Internet Security*, published in May 2002.

Richard R. Brooks is the Head of the Distributed Systems Department of the Applied Research Laboratory, the Pennsylvania State University. His areas of research expertise include sensor networks, critical infrastructure protection, mobile code, and emergent behaviors. He has his B.A. degree in mathematical sciences from the Johns Hopkins University, and performed graduate studies in computer science and operations research at the Conservatoire National des Arts et Métiers in Paris, France. Dr. Brooks received his Ph.D. degree in computer science from Louisiana State University in 1996. His work experience includes being Manager of Systems and Applications Programming for Radio Free Europe/Radio Liberty in Munich, Germany. The consulting tasks Dr. Brooks has performed include the implementation of a stock trading network for the French stock exchange authority, and the expansion of the World Bank's internal computer network to Africa and the former Soviet Union.

