Robust Fusion of Multiple Microphone and Geophone Arrays in a Ground Sensor Network

Dr. D. Lindgren, Mr. H. Habberstad, Dr. M. Holmberg, and Dr. A. Lauberts

Swedish Defence Research Agency
SE-581 11 Linkoping
SWEDEN

email: { davlin, hanhab, mah, andris }@foi.se

ABSTRACT

A sensor network with arrays of microphones and geophones has been deployed along a track in the country. Together with a beamforming technique, the sensor arrays produce bearing estimates to passing vehicles. The bearing estimates are sent to, and fused by, a Kalman tracking filter. We report how we implemented this system with low-complexity distributed-type hardware and software components, and also indicate the level of performance to be expected using a small number of intelligent network nodes. We also study a large data base with acoustic recordings collected in realistic environments with the purpose to classify vehicles traversing the network. Support vector machine classifiers, taking as input spectral features picked up from the emitted audio, are fused in time and space to identify the vehicle model.

1.0 INTRODUCTION

The vision of a wireless ground sensor network is that of a network built up with cheap and tiny network nodes deployed with little preparation and no manual calibration. This network works unattended for months and delivers reliable situation awareness services to a local or remote operator. Each node senses the environment by means of, for instance, microphones, geophones, magnetometers, and/or passive IR detectors, and furthermore collaborates with its neighbor nodes to compare, fuse and refine sensor data with the purpose to detect, track and identify targets. To that end, the envisioned network topology is *scalable* in a sense that allows the number of nodes grow wildly to supply coverage of a large area or a long border. This implies a *distributed* or at least a *decentralized* network topology that enables some target activity to be a local business that involves only that fraction of all network nodes that actually senses the target in question, see Figure 1.

In our opinion, this is just a vision, but it is not science fiction. Although not suitable for a distributed sensor network, the modern cell phone is a proof of that we today can build portable devices with low energy consumption and means to communicate and do advanced signal processing. Furthermore, the Internet is an example of a huge scalable distributed network, although the internet protocol (IP) is not energy-efficient enough for our purposes. Bluetooth is an example of a self-configuring ad-hoc network, although it only allows a restricted number of network nodes. A target can be tracked using fusion methods and identified using modern pattern recognition methods that recognize patterns in, for instance, the emitted sound. Admittedly, there still are important obstacles to overcome to reach our vision, but today we also see many promising ways to advance the network and fusion technologies. No doubt, the force in control of the envisioned sensor network will be able to quickly gain a crucial awareness of ground activities in conflict areas.

Lindgren, D.; Habberstad, H.; Holmberg, M.; Lauberts, A. (2006) Robust Fusion of Multiple Microphone and Geophone Arrays in a Ground Sensor Network. In *Battlefield Acoustic Sensing for ISR Applications* (pp. 12-1 – 12-16). Meeting Proceedings RTO-MP-SET-107, Paper 12. Neuilly-sur-Seine, France: RTO. Available from: http://www.rto.nato.int/abstracts.asp.

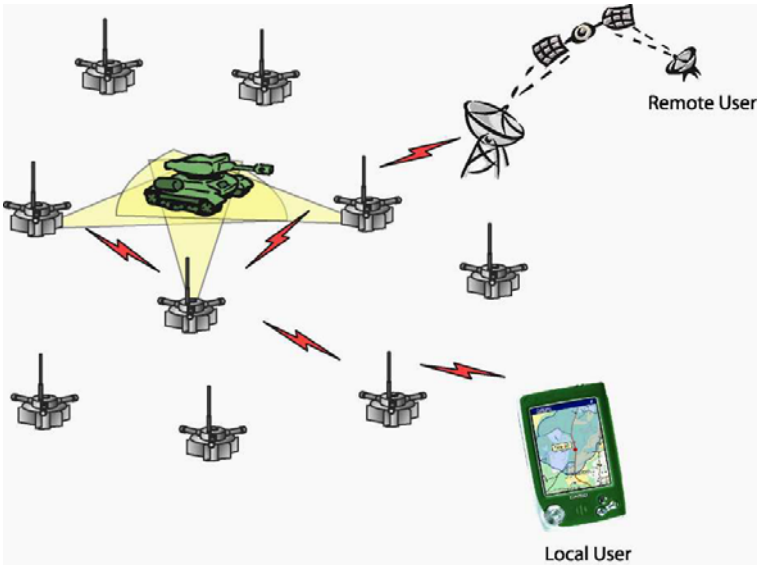

Figure 1: Ground sensor network. Sensors close to a target collaborate in tracking and classification, and the network propagates the information to a local or remote user/operator.

1.1 The Interactive Adaptive Ground Sensor Networks Project

The evolution of the wireless sensor network spans over many scientific and technological domains, and it is indeed difficult to account for more than just a fraction of them within the boundaries of any single research project. Within the Interactive Adaptive Ground Sensor Networks Project (IAM) at the Swedish Defence Research Agency (FOI) we have particularly focused on fusion methods for classification and tracking and also user interaction with a scalable, distributed sensor network. A field trial to collect sensor data for numerical studies off-line was conducted early in the project using a wired sensor network where each network node was connected to an array of either microphones or geophones. A set of military vehicles ranging from tanks to trucks served as targets. This paper reports results from data fusion experiments conducted on these data, mainly tracking of military vehicles, see Section 3.0. Results on the interaction part of the project have already been reported in [1], and will not be treated here.

For studies of automated vehicle classification (model identification), the Swedish Defence Material Administration (FMV) has supplied us with a data base containing over 900 recordings of vehicles passing a set of microphones and geophones in a wired sensor network. Some results from these studies are given in Section 4.0, where the *acoustic* part of the data has been used. We still work on the seismic part, for which we find it difficult to show general results, since a classifier that works well at one site might not work well on another. In other words, it is difficult to obtain a model that is useful on a wide variety of ground conditions.

Besides, in an ongoing project somewhat apart from IAM and in collaboration with SAAB Systems, we use *wireless* network nodes, a fact that of course adds aspects of communication and routing, but also puts to real test abilities of the fusion methods to cope with missing and randomly delayed data. Yet, it is a little early to draw any formal conclusions from that project. The network concept has been reported in [2].

1.2 Related Work

As mentioned, the information flow in the network has been discussed in [1]. We have also reported results on vehicle model classification in [3], and used similar methods on hydroacoustic data for ship classification in [4].

A text that fairly well overviews the components of a wireless sensor network, ranging from sensor tasks to network protocols, is [5]. Another comprehensive text focused on the mathematical aspects of data fusion, and with many references to further reading, is [6].

On subjects related to *target localization* in sensor networks, there are numerous interesting and rather recent contributions, see for instance [7], [8], [9], [10], [11], and [12]. In [13], the sound source of low-flying airborne targets are imaged by a high resolution array of 32 microphones.

An often cited text on *target tracking* is [14]. Some late and interesting contributions on bearings-only tracking are [15, pp. 149-169], [16], and [17]. Particularly on problems related to distributed tracking filters, we like to mention [18] and [19].

Signal classification in both the frequency and time domain is well covered in the literature, see for instance [20], although the best way to handle non-stationarities and to do efficient fusion in practical situations well deserves more treatment. Methods similar to the ones described here can also be seen in, for instance, [21], where features from autoregressive models and spectral estimates are used by support vector machines, Gaussian, and k-Nearest-Neighbor classifiers. Ground vehicles are classified as either tracked or wheeled using microphones and geophones. Particularly interesting is also [22], where the subjects of lightweight detection and classification appropriate for low-power wireless network nodes are discussed.

2.0 The IAM Sensor Network

The sensor network has been deployed hidden in the woods along a 500 m long dirt road in the back country. Figure 2 is a rendering of the environment, done by a testbed system (MOSART) that we develop and use to simulate and replay scenarios. The terrain is rough, and the sensors are placed at different distances from the road and at different heights. Some are placed with line of sight to the road, others all obscured by trees. The ground conditions are also characterized as rough with a mixture of marshland, rocks and tight vegetation. A total of 10 sensor network nodes have been used in the experiments, see the orthographic photograph in Figure 4. The node positions and orientations have been determined mainly by using this photograph, a tape measure, and a compass. Each node includes a horizontal-plane circular array of three 120° interspaced sensors, either geophones or microphones. The use of *an array* of sensors makes it possible to determine the bearing to the emitting vehicle. No height information can be extracted from the used horizontal-plane arrays.

All network nodes amplify and send their raw analog signals to a wired recording central that samples the signals at 48 kHz and stores the uncompressed time series on hard disk drives. The recording central is a PXI 1000B with PXI 4472 I/O-cards, both from National Instruments. A configuration with separate power supplies ensures the node overhearing is kept below -100 dB.

One geophone (of three) is depicted in Figure 3 (a). The metal part of the geophone is stuck into the ground on the circumference of a circle with radius 60 cm. The geophone is a SM6/U-B from INPUT/OUTPUT Inc. in a PE-3 casing, sensitive mainly for seismic shear-waves, that is, vertical vibrations transverse the direction of propagation.

Figure 3 (b) depicts a complete microphone array mounted on an ordinary microphone stand. The microphone array has the radius 13 cm and is typically positioned 1 m above the ground. The microphone elements are of the type KE-4-211-2 from Sennheiser, with diameter 4.75 mm, omnidirectional sensitivity, and 36 dB rated self noise.

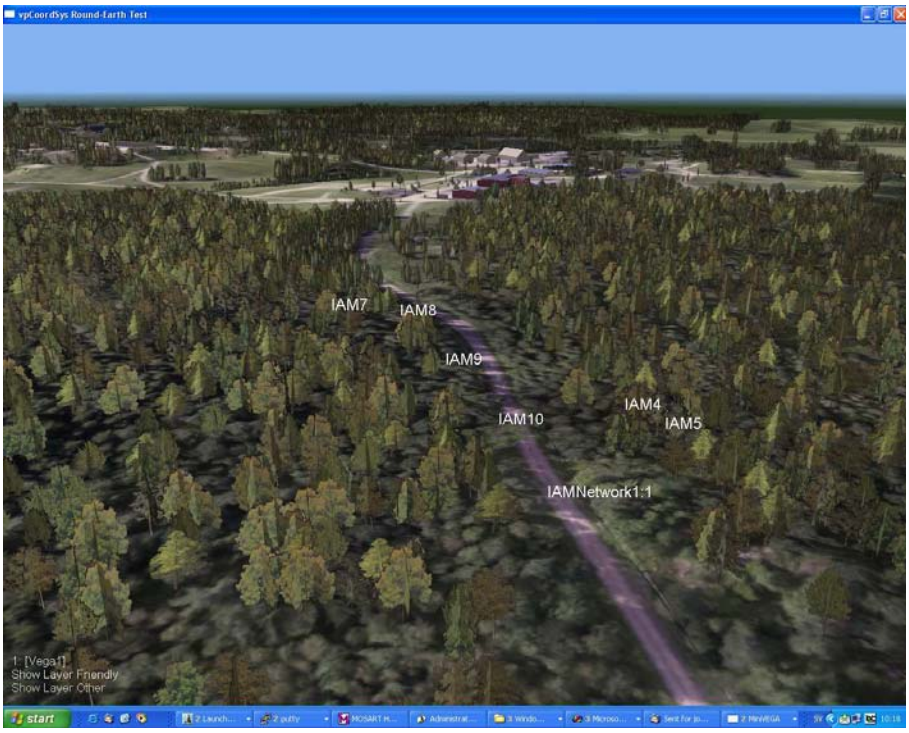

Figure 2: A view of the test site as rendered by the FOI testbed for battlefield simulations, MOSART. The test vehicles traveled on the road, while 10 arrays with microphones and geophones hidden off the road in the woods transmitted their signals to a recorder. The recordings where later analyzed off line and also replayed in the depicted MOSART environment.

(a) **(b)**

Figure 3: Sensors used in the IAM project. In (a) the geophone, put into the ground in arrays of three. In (b) an acoustically well-designed microphone array mounted on an ordinary microphone stand. The discus contains three microphone elements.

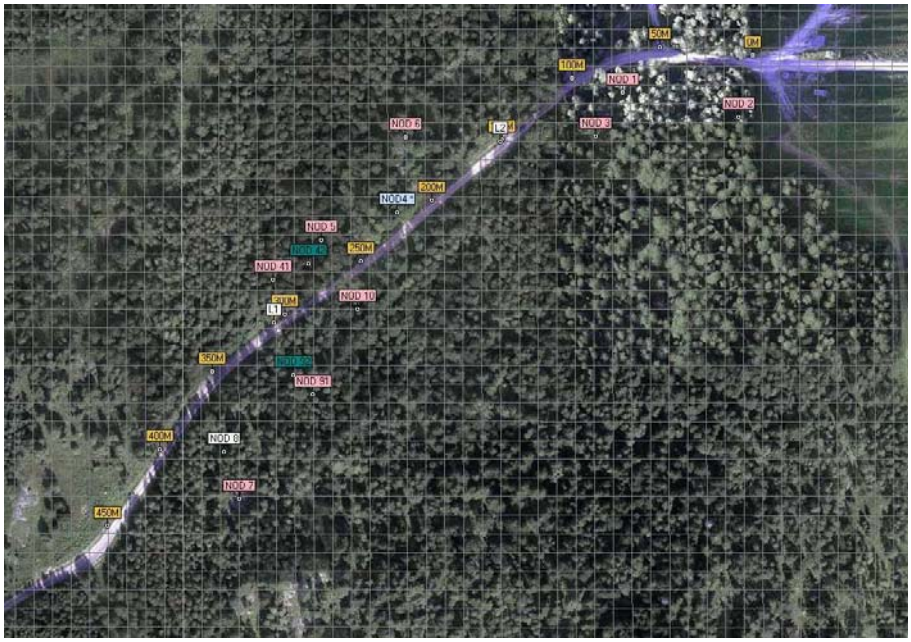

Figure 4: Overview of road and sensor placements. NOD(3,5,6,8,10) are microphone nodes, and NOD(1,2,4x,7,9x) are geophone nodes. L1 and L2 are laser triggers used for reference measurements. The grid resolution is 10 meter.

3.0 Tracking

By using one sensor array, the bearing to a target can be estimated. In turn, by fusing the bearings received from a collection of sensor arrays that sense the same target, a *tracking filter* can estimate the target position and speed. Below we describe how these steps have been implemented by programs that are fast and simple, although not shown optimal in accuracy.

3.1 Bearing Estimation

As mentioned, each sensor array has three sensors interspaced 120° in a circular arrangement. The microphone array radius is $r = 13$ cm and the geophone array radius is $r = 60$ cm. The direction of an emitter, acoustic or seismic, is determined by comparing the phases of the sensor signals, using the fact that the incident wave propagates with finite speed, and thus reaches the sensors at slightly different time instances.

We assume a plane acoustic wave propagating from the emitting vehicle along a straight line to the sensor array. This is a fair assumption only when the distance to the target is far greater than the size of the emitter and the array radius (the far-field assumption). Furthermore, we assume in this discussion that there is only one dominating emitter, which is to say, only one vehicle is in the vicinity of any sensor array when the direction is determined. The sensor signals are analyzed in time windows of 0.1 s, assuming the that the bearing change attributable to vehicle movement is insignificant during this time interval.

Say the emitter is about north-north-east vis-à-vis the sensor array according to Figure 5. Then the received wave front will reach the microphone closest to the emitter first–s_1 in the figure. After that, it will take $t_{12} = d_{12}/c$ s for the wave front to reach s_2, where c is the acoustic wave propagation speed in air for the microphone array, or the seismic wave propagation speed in ground for the geophone array. After another $t_{23} = d_{23}/c$ s it reaches s_3.

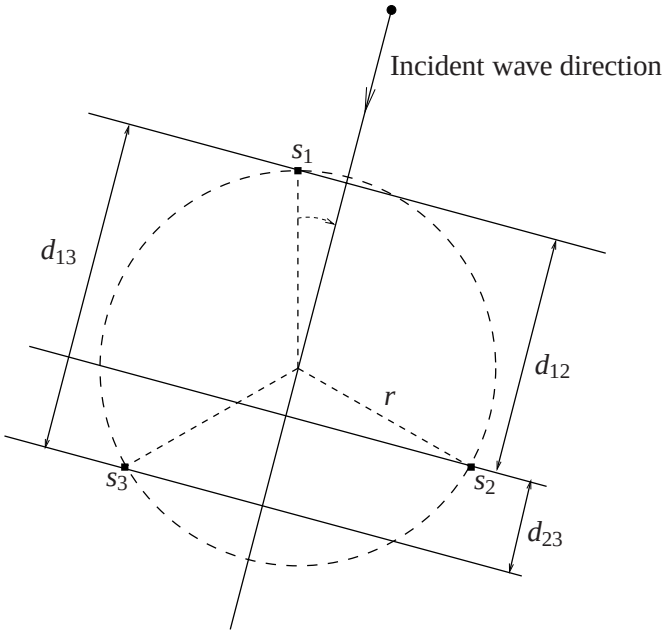

Figure 5: The circular sensor array with three sensors s_1, s_2, and s_3. Plane wave propagation is assumed. The acoustic or seismic wave from an emitter north-east vis-à-vis the sensor array will reach s_1 first, then s_2, and last s_3. The direction θ can be determined by estimating the time it took the wave to propagate d_{12}, d_{13}, and d_{23}.

Indeed, by knowing the distance $d_{23} = t_{23}/c$, trigonometry gives the sought bearing angle θ. In theory, the t_{23} could be found as the delay τ that maximizes the dependence between $s_2(t-\tau)$ and $s_3(t)$. In the discrete time domain (sampled signals), we estimate t_{23} by maximizing the *Pearson product-moment correlation coefficient* (PMCC) between time windows delayed integer amounts of sampling intervals. PMCC is a measure of how well a linear equation describes the relation between sampled signals. With the propagation speed c, the sampling interval T, and the maximizing integer time delay k_{23} ($\approx t_{23}/T$) at hand, the angle θ is given by a trigonometric exercise that ends up with

$$\theta = \sin^{-1}\left[\frac{d_{23}}{2r\cos\frac{\pi}{6}}\right] = \sin^{-1}\left[k_{23}\cdot\frac{cT}{r\sqrt{3}}\right]. \quad (1)$$

Here, the wave reached the sensors in the particular order s_1, s_2, and s_3. For the five other possible orderings the expressions for θ are simlar.

The merit of the described algorithm is that it is fast and has a light-weight implementation. The accuracy is constrained by the sampling frequency, since only integer time delays are considered. High accuracy thus requires high sampling rate (over sampling), which puts extra demand on hardware.

A second drawback is that the wave propagation speed c is required to be known beforehand. In air, it is temperature dependent, but fairly constant and within the range 320-350 m/s, so setting $c = 340$ m/s gives fair accuracy in general. The seismic propagation speed, however, varies between 1500 and 8000 m/s, depending on the media. To that end, seismic waves split into a longitudinal and a transverse component, where the latter has a propagation speed only 60-70% of the former. The used geophone is however sensitive mainly to the transverse component. To deal with the ground uncertainty, we calibrated the geophones and determined the ground speed by analyzing how the sensors responded to sledgehammer stimuli conducted at certain spots.

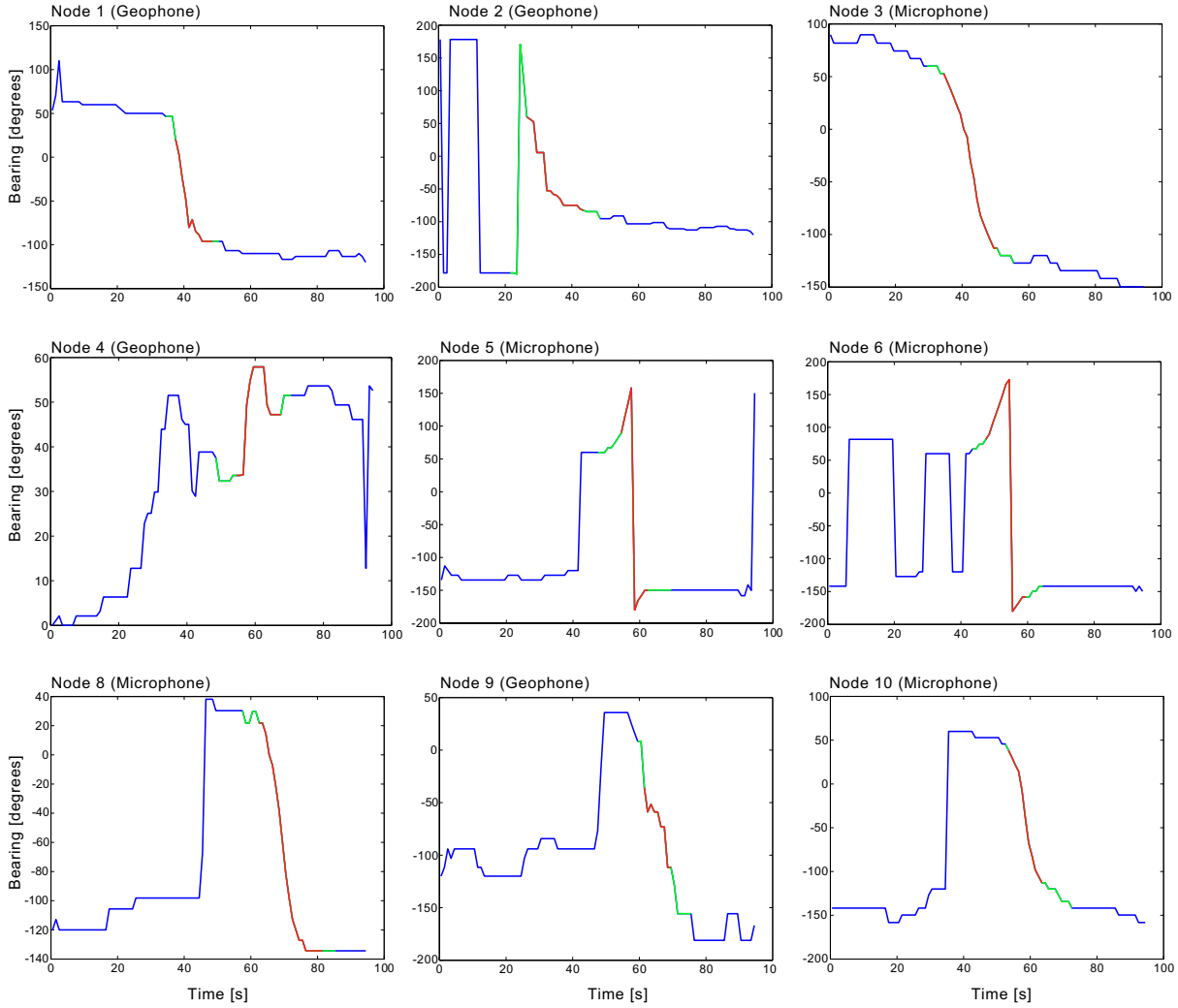

Figure 6: Bearing estimates of a vehicle passing the network from north to south with the speed 35 km/h (22 miles/h), see Figure 4. North is 0 degrees. On color media, red indicates a signal power exceeding 0.05 times the maximum power in the run. Node 7 has been excluded here, since it gave no useful estimates.

Of course, when a slightly more complex algorithm can be accepted, we could, in addition to (1), account for the whole set of equations induced by *all three* time delays t_{12}, t_{13}, and t_{23}. Then, one of these equations could be used to solve for c, and prior knowledge of the wave propagation speed is no longer crucial.

3.2 Numerical Example

Figure 6 shows the bearing estimates for the different nodes during the passage from north to south of a BTR70 at 35 km/h (22 miles/h). BTR70 is a 12 ton, 8-wheeled military vehicle. The bearing estimate is plotted against the time. Node 7 (geophone) gave no sensible estimate, so that plot is not provided here. On color media, the curves are color-coded with the power, based on a calibration. Red indicates the power exceeds 0.05 times the maximum calibration power.

For the acoustics, the analysis used 0.1 s windows with a sample rate reduced to 12 kHz. In Figure 7 we have made a simple analysis of the impact of using integer time steps in the time delay calculation with these settings. Apparently, the uncertainty attributable to quantified delays is about ±4.5°. This is however only one of many sources for uncertainty.

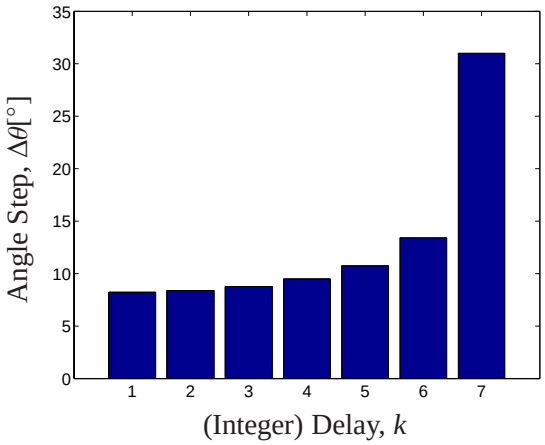

Figure 7: The angle steps (resolution) at different integer sample time delays. Delay $k = 3$ means, according to (1), an angle of $22.2°$, and $k = 4$ an angle of $\theta = 30.2°$. For any of the six possible orderings of s_1, s_2, and s_3, it is only necessary to consider a $360°/6 = 60°$ interval. This restricts the delays to $k \in \{-3, -2, \ldots, 3\}$, so the angle resolution is, according to the box plot, no larger than $\pm 9°/2$.

3.3 Tracking Filter

The task of the tracking filter is to estimate the state of the target, that is, the position and speed vector. The tracking filter starts with some initial state estimate when the target enters the network, and then updates this estimate recursively when bearings are received from sensor arrays within acoustic or seismic range of the target.

State estimation with the bearings-only measurements is a nonlinear problem in the sense that the target bearing angle transforms nonlinearly to the target position (sinuses and cosines). Since an implementation with low complexity is preferred, we settle with a suboptimal linear algorithm that builds on the assumption that the location uncertainty induced by inaccuracies in the bearing estimate has the shape of an ellipse (a Gaussian), rather than a sector. Again for low complexity, a *linear* vehicle model has been used, which basically is a Newtonian model of a mass affected by finite forces. With ellipse-shaped or Gaussian uncertainties and linear dynamics, the (linear) Kalman tracker is in a sense the optimal choice, see [23]. Particularly we use the (dual) information form of the Kalman filter, in which transformed information sent from nodes having information enters the filter additively in an appealing manner. In contrast to the primal Kalman filter, the information form is also appropriate for bearings-only estimation, since it allows the uncertainty along the bearing direction to be infinite.

Regardless filter type, the state can be estimated in a (de)centralized or in a distributed manner. In the centralized approach, there is only one state estimator per target, to which every node in vicinity of the target sends its bearing and perhaps also the corresponding bearing variance. In the distributed approach, each node in vicinity of the target has its own state estimator, and shares its state and perhaps also the state covariance with its neighbor nodes. The distributed approach is more complex and requires more network bandwidth, but gains in robustness since many nodes share the state. We have yet not dealt with the degeneration followed from estimators naïvely interchanging their information back and forth. For a discussion in depth on distributed Kalman filtering, see [18] and [19].

3.4 Numerical Example

Figure 8 shows the (naïve) distributed Kalman filter track for the same passage of BTR70 as in the previous example. A network node only sends its bearing estimate to the tracker when the signal power exceeds a

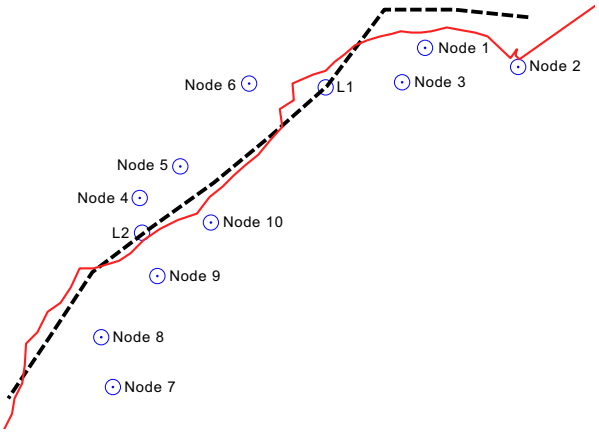

Figure 8: Example of a Kalman vehicle tracking. The dashed line is the true track, compare with the road in Figure 4. The vehicle is a BTR70 driving from north to south with the speed 35 km/h (22 miles/h).

fixed threshold set 0.05 times the maximum calibration power (the red curve part in Figure 6). Furthermore, the tracker disregard received bearing estimates that are considered outliers, that is, bearing estimates that not at all fit the track.

The tracker is initialized by the node that first produces a distinct bearing with sufficient signal power. The initial position is set according to this bearing, and with respect to a node/target distance beforehand known to be the limit for a useful bearing estimate. This initial state is usually very uncertain.

In conclusion, the tracking accuracy depicted in Figure 8 is probably sufficient for surveillance purposes. This has been achieved using 10 sensor arrays to survey a 500 m track through the woods. In open terrain, it will probably suffice with fewer sensors.

4.0 Vehicle Classification by Audio

Audio recordings of vehicles passing a sensor network do, not surprisingly, show a systematic difference depending on the type of passing vehicle, say a car and a motorcycle. Most people can by experience and/or little training recognize the difference and rather accurately distinguish between cars and motorcycles merely by listening to the sound of the recordings. However, natural variations such as different speeds and gears, Doppler shift, and so on, variations that the human brain easily generalize, can actually make it rather difficult to construct an automated classifier that does an equally god job as a human when it comes to distinguishing between different vehicles. To make an automated classifier work well, it usually needs to be "trained" by a large set of recordings with known vehicle types—training sets that encompass a major part of all variations within each vehicle class. When a new recording in some sense resembles one of the recordings in the training set, the *library*, there is a good chance the classifier makes the right choice. These libraries, however, need to be excessively large and become very expensive unless the classifier is given ability to extrapolate or generalize. By *ability to generalize* we mean that the classifier should make a satisfactory classification although the match with some of the recordings in the library is not perfect. This is really the classical problem within the field of pattern recognition and machine learning: to learn something general from a limited set of observations, see, for instance, [24].

By using audio recordings from a *sensor network* with a whole set of microphones placed at different locations, more information/observations is given, and thus, the sensor network should give better classifier performance compared to a single microphone. The sensor network, however, also gives an increased richness

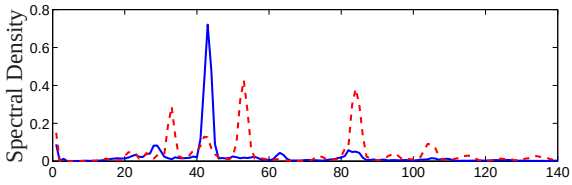

Figure 9: Example of normalized periodograms from two different vehicles.

of variation in the observations, and the demand on the classifier's ability to robustly generalize increases in the same degree. The numerical data used in this section are collected at many different locations and includes recordings both in the summer and in the winter, see Table 1. This rich data set gives opportunity to study the generalization ability of a classifier by collecting a library at one location and validating it on another.

4.1 Spectral Features

Inspecting the frequency spectrum of a signal often gives a "better picture" than the inspection of the signal directly. This is particularly so when there are periodic components in the signal, for instance, from a spinning engine. These periodic components appears as distinct peaks in the spectrum. The spectral representation also makes it easy to focus on frequency bands known to contain relevant information, *feature selection*. We use the *discrete Fourier transform* (DFT) to extract spectral features from the microphone signals. The idea is, of course, that there are significant systematic differences in the spectral feature sets due to different vehicles, and that these differences can be learned by an automated classification program.

Signal classification in the frequency domain is well covered in the literature, see for instance [20], although the best way to handle the non-stationarity in the vehicle audio appears to be an open question.

4.1.1 Discrete Fourier Transform and Periodogram

Denote by $x_i(t)$, $t = 1, 2, \ldots, n$ the sampled (audio) signal from *one* microphone under a fixed sampling frequency and within a time window with length n samples. The index i is the recording ID and recordings with different IDs can, of course, be due to different vehicles, but also just repeated recordings on the same vehicle. The standard discrete Fourier transform (DFT) offers one way to estimate the signal power of $\{x_i(t)\}_1^n$ at the (scaled) frequency f as

$$U_i(f) = \frac{1}{n}\sum_{t=1}^{n} x_i(t)e^{-2j\pi tf/n}, \quad f = 0, 1, \ldots, n-1, \tag{2}$$

where $j = \sqrt{-1}$, $U_i(f)$ is complex-valued, see for instance [25, pp. 122]. k frequencies $f_1, f_2, \ldots, f_k$ known to yield good classifier performance are selected once for all, and the absolute value of these selected $U_i(f)$ form the feature vector or *periodogram* X_i,

$$X_i = \begin{bmatrix} |U_i(f_1)| & |U_i(f_2)| \cdots |U_i(f_k)| \end{bmatrix}^T. \tag{3}$$

The phase is believed to contain less valuable information for the classifier and is disregarded. Figure 9 illustrates the low frequency part of the periodograms from two different vehicles. The periodograms are normalized to unit Euclid norm.

4.1.2 Limitations of the DFT

A spectral estimate by the DFT needs a rather stationary signal with periodic elements to yield as distinct features (peaks) as in Figure 9. Transient processes like accelerations, gear shifts and fast Doppler shifts will *smear out* the peaks considerably and make it more difficult to distinguish the periodogram of one vehicle from that of another. In that respect, the window length n determines a mandatory trade off between resolution in time and frequency. Since the spectrum cannot be expected to be constant over time, the periodograms from one signal *cannot* be naïvely aggregated over time by–say–the mean value.

4.2 Classification Method

A vehicle is classified by comparing the periodogram to a set of periodograms in a class library. This library is based on (training) recordings where the vehicle class is known beforehand, so if the new periodogram matches a periodogram in the library, it can directly be associated with a vehicle class. In this paper we exclusively study the *support vector machine* (SVM) classifier which is one among many methods to represent and use the class library. Below we will briefly describe the SVM, which is well-known in the literature, see [24, 26], and also how to fuse data from multiple microphones.

4.2.1 Support Vector Machine (SVM)

The SVM classifier is fundamentally a binary classifier $f : \mathbb{R}^n \to \mathbb{R}$, predicting the class label ω_i that takes on values in the set $\{-1, 1\}$. A new observation is classified using the signum function, $\hat{\omega}_i = \operatorname{sgn} f(X_i)$. For the library (training) periodograms $\{X_i, \omega_i\}_1^N$ with known ω_is, f is defined as the solution to

$$\min_c \sum_{i=1}^{N} V(f(X_i), \omega_i) + \xi c^T K c \tag{4}$$

with

$$f(X_i) = \sum_{j=1}^{N} c_j K(X_i, X_j) \tag{5}$$

and the loss $V(f(X_i), \omega_i) = \max[1 - \omega_i f(X_i), 0]$. Note that $c = \begin{bmatrix} c_1 & c_2 \cdots c_N \end{bmatrix}^T$. $K(X_i, X_j)$ is a *reproducing kernel*, and particularly in this paper, we use the polynomial kernel

$$K(X_i, X_j) = (\gamma X_i^T X_j + 1)^d \tag{6}$$

and the radial basis kernel

$$K(X_i, X_j) = e^{-\gamma \|X_i - X_j\|^2}. \tag{7}$$

γ and d are tuning parameters. The kernel matrix K has $K(X_i, X_j)$ on the ith row and jth column. The term $\xi c^T K c$ in (4) serves to smooth f (Tikhonov regularization). We use $\xi = 1$.

The minimization problem (4) is a *quadratic programming problem* with a problem structure that allows very large instances to be solved efficiently . Far from all periodograms in the class library need to be stored, which is an important feature when implemented in a network node with constrained resources. Compared to a simple *Gaussian classifier*, the SVM is, however, a computationally rather heavy program. The libraries used in this study varies between 12 and 20 MBytes.

In the literature there are many schemes to combine a set of binary SVM classifiers to handle multiclass problems. In the multiclass case, the class label takes on values $\omega_i = 1, 2, \ldots, q$, where q is the number of classes. It still is an open question how to do this the best way, see [26]. We use an implementation that defines an SVM classifier for every class pair. The results of the binary classifications are then combined in a non-trivial way. We refer to [27] for details.

Reliability Estimation When periodogram X_i is classified, also the classification reliability λ_i needed for efficient data fusion is estimated. λ_i is a number between 0 and 1, where values close to 1 indicate very confident classification. λ_i is induced by the class *posterior probabilities* $P_\omega(X_i)$ assumed available as an output of the classification program (SVM), one per class, $\omega = 1, 2, \ldots, q$. It is (disputably) assumed the probabilities sum up to unity,

$$\sum_{\omega=1}^{q} P_\omega(X_i) = 1, \quad i = 1, 2, \ldots \tag{8}$$

that is, the vehicle is in the library with probability 1. A confident classification is one with one class posterior probability significantly larger than all others. We define λ_i as

$$\lambda_i = \frac{\sup_\omega P_\omega(X_i) - q^{-1}}{1 - q^{-1}}, \quad \omega = 1, 2, \ldots, q. \tag{9}$$

4.2.2 VIP Voting Distributed Data Fusion

By *data fusion*, two or more observations of the same target made by two or more sensors are combined to a unified result or classification. Data fusion also combines observations made at different time instances, so we may distinguish between *spatial fusion* where observations are collected from different sensors, and *temporal fusion*, where observations are collected over time. *Decision fusion* is a special fusion concept, where each observation is classified individually, whereafter these separate classifications are combined to a unified classification (in contrast to combining raw data), see [6, pp. 205-238]. Decision fusion is closely related to distributed classification, and particularly appropriate to use with wireless sensor networks since raw data is newer transmitted. In this paper it is assumed that a recorded signal is due to one vehicle only. We use a decision fusion flavor called *VIP Voting*.

Ordinary voting is a simple fusion scheme where each classifier votes for a class, and the class in majority establishes the final fused result, see [28]. With the *VIP voting* scheme, this election is not entirely democratic since only the *most reliable* classifications are allowed to vote (the *very important persons*). This method requires that classifications are assessed with the measure of reliability, λ_i, described above, which in practice indicates how well the classified feature vector X_i matches the library. A value close to one means there is a very good match, while a value close to zero means X_i does not match any of the feature vectors in the library. The latter situation is typical for heavily disturbed or distorted signals, when it is motivated to rate the classification at hand as unreliable and disregard it.

4.3 Numerical Experiment

The data set used here contains recordings at 12 different test sites and at different seasons. The data sets are named A-L and are enumerated in table Table 1. The November measurements where conducted winter time with snow on the ground and partly icy roads. 14 different military vehicle models passing the microphone network at different speeds where recorded, see Table 2. The acoustic sensor setup consisted of 4 microphone nodes (u_1, u_2, u_3, and u_4) with one microphone in each node and with sampling rate well over the Nyquist frequency.

We are focused on classifying a vehicle passage using both spatial and temporal fusion on periodograms from all 4 microphone nodes and all time instances. The analysis presented here is completely conducted off-line in MATLAB with CPU intensive parts in C++, althogh the same algorithms recently have been successfully implemented in the wireless sensor network described in [2]. We use the SVM implementation *LIBSVM*, see [27]. The LIBSVM classifier outputs a score vector with estimated class probabilities $P_\omega(X_i)$. As desired, the score vector sums up to unity and the actual classification is simply the class corresponding to maximum $P_\omega(X_i)$.

Table 1: Sensor settings. Note the different ground surfaces, and the different seasons.

Setting	Season	Ground
A	Winter	Dirt road or grass
B	Winter	Grass
C	Winter	Dirt road through the woods
D	Winter	Dirt road
E	Summer	Grass
F	Summer	Asphalt

Setting	Season	Ground
G	Summer	Grass
H	Summer	Dirt road
I	Summer	Grass
J	Summer	Grass
K	Summer	Dirt road
L	Summer	Dirt road

Table 2: Type code, vehicle type, and number of recorded passages at the different sensor settings A-L.

Type Code	Weight	Propulsion	Tot	A	B	C	D	E	F	G	H	I	J	K	L
LW1	Light	Wheels	14							2	12				
LW2	Light	Wheels	67	14	8	13	2	5	11	2	12				
HW1	Heavy	Wheels	58	14	8	12	10	4	10						
LT	Light	Tracks	44	14	8	12			10						
HT3	Heavy	Tracks	48	14	11	12			11						
HT8	Heavy	Tracks	44	14	8	12			10						
HW3	Heavy	Wheels	82	14	8	12			10			8	18	12	
HT9	Heavy	Tracks	102	14	14	12			10	18		4	18	12	
HW2	Heavy	Wheels	38									8	18	12	
HT4	Heavy	Tracks	30									4	14	12	
HT2	Heavy	Tracks	34									4	18	12	
HT1	Heavy	Tracks	30									4	14	12	
HT10	Heavy	Tracks	65							34					31

Preprocessing The preprocessing serves to form a normalized periodogram with appropriate resolution:

1. Resample to 25 kHz
2. Extract a continuous signal part where the signal power exceeds 0.2 times the calibrated maximum
3. Calculate periodograms based on DFTs of non-overlapping 8192 point rectangular windows
4. Extract periodogram components indexed 5 to about 100 (15-300 Hz)
5. Normalize to unit Euclid norm

The data reduction followed by extracting certain low frequencies of the close passage reduce the computational load considerably and also, to some extent, avoid numerical issues. The SVM classifier is applied directly on the preprocessed data, that is, using no PCA compression or the like.

Validation Approach We assess the classifier performance as the percentage correctly classified test passages. None of the test data are included in the library. A set of alternative partitionings of the available data are tested using the different sensor settings in Table 1.

To be certain not to obtain results that are too optimistic, all single-vehicle passages in the available data base are used – even awkward sounding passages, starting, stopping, accelerating, and so on, are classified with no exception. This means that the classifier performance is evaluated on data that include test cases where there is really no hope to make a correct classification, not even for a trained human ear.

The number of periodogram components and kernel function to be used, have been selected to yield a classification accuracy of around 99% when testing the classifier on the library data (training data). With higher hit rates, the performance degrade due to over fitting (fitting to noise).

4.4 Classification Results

The overall results are given in Table 3. Not surprisingly, different choices of library and test data give different number of correctly classified passages, ranging from 81% to 99%. Libraries including many vehicle models usually give lower accuracy.

Table 4 gives an example of a confusion matrix, where the rows are the actual vehicle model, and the columns are the estimated. The test data set is from setting I and J and the library from setting K.

Table 3: Sum up of all passage classification tests. The Library and Test Data columns give the sensor settings from which data is collected, see Table 1. The rightmost column gives the average share of correct classifications.

Library	Test Data	No. of Vehicles	Correct
A	B, C, D	7	81%
K	I, J	7	91%
A, K	B, C, F, G, I, J	12	81%
I, J, K	A, B, C, F	2	99%

Table 4: Confusion matrix for classification at setting I and J. True class on the rows, estimated on the columns. The vehicle class library is from setting K.

	HW2	HT2	HW3	HT4	HT1	HT9	HT6
HW2	24		2				
HT2		19			1	1	1
HW3			26				
HT4	3			15			
HT1			1		17		
HT9			2		3	17	
HT6							18

5.0 Conclusions

A sensor network with microphone and geophone arrays has been deployed in an attempt to track passing vehicles. A lightweight distributed-type linear Kalman tracking filter fused bearings from 10 nodes placed besides a 500 m long dirt road in the back country. Both the microphone and geophone arrays contributed to the track estimate, although 2 geophone arrays apparently were subjected to ground conditions that rendered the bearing estimates more or less useless. The estimated tracks showed an, in our opinion, acceptable resemblance with reality. 10 nodes/500 m road in rough terrain appears to be an appropriate node density for this task.

We have also studied realistic audio recordings for model classification of passing military vehicles. It has been investigated how well this task can be accomplished using normalized spectral features of the audio recordings together with support vector classifiers. By using both temporal and spatial decision fusion (VIP voting), the over-all success rate in general exceeds 80% among 12 vehicle models. It appears that classification in the summer time is easier (90% successful classification among 7 vehicle models).

References

[1] M. Brannstrom, R. Lennartson, A. Lauberts, H. Habberstad, E. Jungert, and M. Holmberg. Distributed data fusion in a ground sensor network. In *Proc. of the 7th International Conference on Information Fusion*, pages 1096–1103, Stockholm, Sweden, July 2004.

[2] J. Stoltz, K. Davstad, M. Bjorkman, and D. Lindgren. An unattended and distributed ad-hoc sensor network for classification and tracking. In *Proc. of the 2nd International Conference on Military Technology*, pages 349–354, Stockholm, Sweden, October 2005.

[3] A. Lauberts and D. Lindgren. Generalization ability of a support vector classifier applied to vehicle data in a microphone network. In *Proc. of the 9th International Conference on Information Fusion*, Florence, IT, 2006.

[4] D. Lindgren, E. Dalberg, R.K. Lennartson, M.J. Levonen, and L. Persson. Surface ship classification in a littoral environment using fusion of hydroacoustic and electromagnetic data. In *Proc. of the Oceans'06 MTS/IEEE*, Boston, MA, 2006.

[5] F. Zhao and L. Guibas. *Wireless Sensor Networks*. Elsevier, San Francisco, CA, 2004.

[6] David L. Hall and Sonya A. H. McMullen. *Mathematical Techniques in Multisensor Data Fusion*. Artech House, Norwood, MA, 2 edition, 2004.

[7] G. Simon et al. Sensor network-based countersniper system. In *Proc. of the 2nd ACM Conf. Embedded Networked Sensor Systems*, Baltimore, MA, 2004.

[8] J.C. Chen, R.E. Hudson, and K. Yao. Maximum-likelihood source localization and unknown sensor location estimation for wideband signals in the near-field. *IEEE Trans. Signal Processing*, 50(1843-1854), Aug. 2002.

[9] J.C. Chen, K. Yao, and R.E. Hudson. Source localization and beamforming. *IEEE Signal Processing Magazine*, 19(2nd):30–39, March 2002.

[10] J. Chen, L. Yip, J. Elson, H Wang, D. Maniezzo R. Hudson, K Yao, and D. Estrin. Coherent acoustic array processing and localization on wireless sensor networks. In *Proc. of IEEE*, volume 91, pages 1154–1162, Aug. 2003.

[11] Duckworth et al. Acoustic counter-sniper system. In *Proc. of the SPIE International Symposium of Enabling Technologies for Law Enforcement and Technology*, 1996.

[12] Y. Huang, J. Benesty, and G.W. Elko. Passive acoustic source localization for video camera stearing. In *Proc. IEEE ICASSP*, volume 2, pages 909–912, June 2000.

[13] T.S. Brandes and R.H. Benson. Sound source imaging of low-flying airborne targets with an acoustic camera array. *Applied Acoustics, To Appear*, 2006.

[14] S. Blackman and R. Popoli. *Modern tracking Systems*. Artech House, Norwood, MA, 1999.

[15] R. Karlsson. *Particle Filtering for Positioning and Tracking Applications*. Dissertation 924, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 1998.

[16] S. Arulamaplam and B. Ristic. Comparison of the particle filter with range-parameterized and modified polar EKFs for angle-only tracking. In *Proc. SPIE, Signal and Data Processing of Small Targets*, pages 288–299, Orlando, FL, 2000.

[17] Gustaf Hendeby, Rickard Karlsson, Fredrik Gustafsson, and N. Gordon. Recursive triangulation using bearings-only sensors. In *IEE Seminar on Target Tracking: Algorithms and Applications*, Birmingham, March 2006.

[18] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. on Automatic Control*, 49(9):1520–1533, 2004.

[19] R. Olfati-Saber. Distributed Kalman filter with embedded consensus filters. In *Proc. of the 44th IEEE Conference on Decision and Control*, pages 8179–8184, 2005.

[20] R. H. Shumway. Discriminant analysis for time series. In P. R. Krishnaiah and L. N. Kanal, editors, *Handbook of Statistics*, pages 1–46. North-Holland Publishig Company, 1982.

[21] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed. Detection, classification and tracking of targets in distributed sensor networks. *Signal Processing Magazine*, pages 17–29, March 2002.

[22] L. Gu et al. Lightweight detection and classification for wireless sensor networks in realistic environments. In *Proc. of the 3rd ACM Conf. Embedded Networked Sensor Systems*, San Diego, CA, Nov. 2005.

[23] T. Kailath, A.H. Sayed, and B. Hassibi. *Linear Estimation*. Prentice Hall, 2000.

[24] T. Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[25] Sanjit K. Mitra. *Digital Signal Processing. A Computer Based Approach*. McGraw-Hill, 1998.

[26] R. Rifkin and A. Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101–141, 2004.

[27] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[28] J. Llinas. Fusion based methods for target identification in the absence of quantitative classifier confidence. In *Proc. of the SPIE Signal Processing, Sensor Fusion, and Target Recognition VI Conference*, Orlando, FL, April 1997.