Robust Distributed Detection Using Low Power Acoustic Sensors

Brian P. Flanagan and Kenneth W. Parker

The MITRE Corporation, 7515 Colshire Dr., McLean, VA 22102, USA;
bflan@mitre.org, kparker@mitre.org

ABSTRACT

This paper describes a robust detection algorithm implemented on a network of acoustic sensors. The sensors are severely constrained in both power and computational performance. A variety of techniques are employed to extract maximum detection range while minimizing false alarm rates under these constraints. These include automatic gain control, background estimation and adaptive thresholding, and collaboration among distributed sensors for false alarm mitigation. The resulting algorithm is both robust and sufficiently general to be applied in a variety of sensor domains. The algorithm was implemented and deployed on prototype hardware and operated in real time under realistic operational conditions.

Keywords: Detection, Acoustics, Netted Sensing

1. INTRODUCTION

Modern battles are increasingly being fought in complex urban environments. Surveillance in these areas is very difficult to perform using traditional stand off sensors. An approach with growing popularity is to use large numbers of very inexpensive sensors distributed throughout the area of interest. These sensors are typically very resource limited both in terms of processing capability and battery life. Consequently many traditional approaches to detection, classification, and tracking need to be rethought in this environment.

Previous work has focused on acoustic sensors due to their relatively low cost. Much of this work has been performed in relatively open areas with specific military targets that have strong harmonic signatures. In urban areas, however, the acoustic noise environment can be quite dynamic, requiring a highly adaptive approach. In addition, typical targets may not be military vehicles, but civilians passenger cars and trucks. Civilian vehicles tend to have far less distinctive acoustic signatures.

In this paper we will describe a detection algorithm which provides robust performance under these conditions which is also general enough to be applicable to a wide variety of sensor domains and environmental conditions. The remainder of this paper is organized as follows: Section 2 describes the background noise environment and the target signatures in detail. Section 3 describes the detection algorithm for a single acoustic sensor. Section 4 discusses the collaboration between sensors to perform a distributed detection algorithm. Section 5 describes the experiment used to evaluate the detector and shows its performance in an operation scenario. Section 6 concludes the paper and discusses future development plans.

2. ANALYSIS OF BACKGROUND NOISE AND TARGET SIGNATURES

2.1. Background Noise

Previous research has demonstrated that acoustic background noise can be highly non-stationary and non-Gaussian.[1] In the urban environment, we also have the additional difficulty of a large number of targets not of interest.

An example of the variability in background conditions is shown in Figure 1. This data was collected using the standard microphone on board a Crossbow Mica2 mote with a sample rate of 4000 samples/sec. Under calm conditions the noise background is quite stable, but wind can produce dramatic and rapidly varying noise levels that are very difficult to compensate for. Other common noise sources include animal sounds, construction equipment, thunder, rain, aircraft, etc.

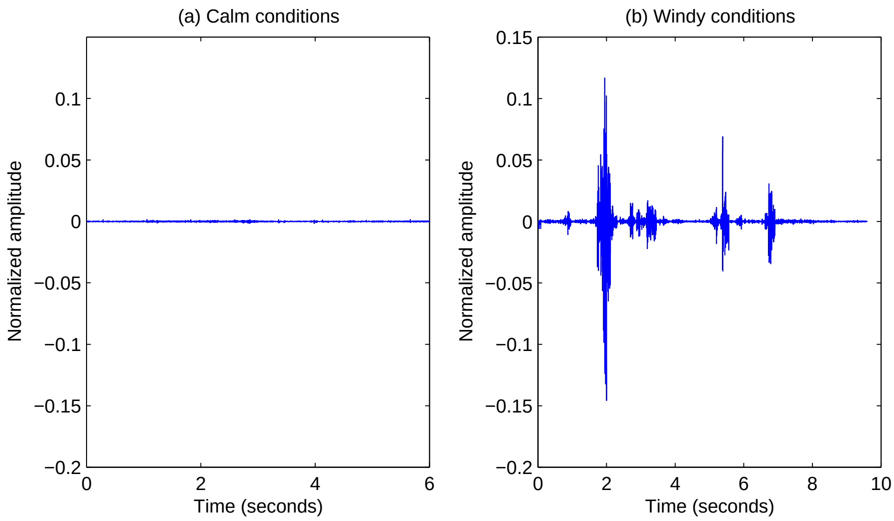

Figure 1. Acoustic time domain signatures for background energy in quiescent (a) and windy (b) conditions

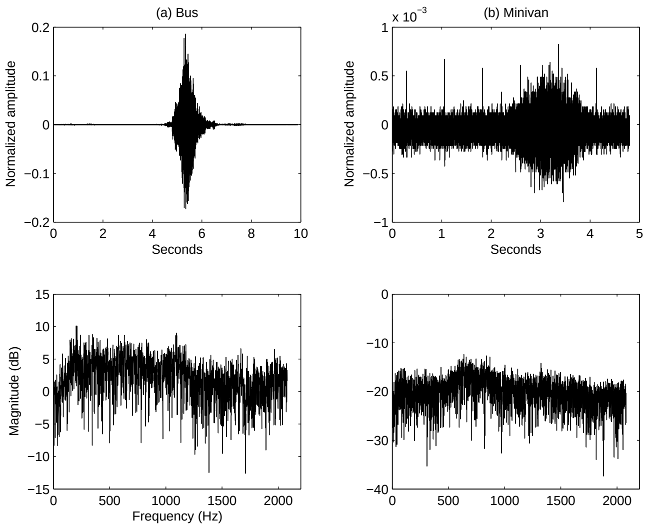

Figure 2. Acoustic time domain and frequency domain signatures for passenger bus (a) and minivan (b).

2.2. Target Signatures

Previous research on acoustic detection of vehicles has focused on military targets which have strong harmonic signatures.[1,2] Unfortunately, civilian vehicles often do not. The acoustic energy is typically spread over a fairly broad band with little evidence of harmonics.

Typical time domain and frequency domain signatures for civilian targets are show in Figure 2. As with the background data shown in the previous section, this data was collected using the standard microphone on board a Crossbow Mica2 mote with a sample rate of 4000 samples/sec. The targets were approximately 5 feet from the targets at closest point of approach during the data collection.

Harmonic features are not evident for either the bus or minvan. A slight elevation is evident in the bus spectrum from 100 to 1200 Hz. For the minivan there is a slight increase in energy from 500 to 1000 Hz. There is also almost a 30 dB difference in signal level between the bus and minivan so any detection algorithm must be capable of dealing with a wide dynamic range in target signal strength.

3. SINGLE SENSOR DETECTION

Given the wide variation in noise background and target signatures, it's clear that we need a very robust detection algorithm that can quickly adapt to changing conditions. A common approach when working with such a nonhomogeneous environment is to use an order statistic constant false alarm rate (OS-CFAR) detection algorithm.[3] In an OS-CFAR algorithm, the input samples over a given window size are rank ordered according to increasing magnitude. The k^{th} ordered value is then used as a test statistic. One advantage of this approach is that it can be very efficiently implemented using a median filter.

In the final detection algorithm we made multiple uses of order statistics to estimate both the noise power and the variation in the noise power over time. This resulted in a very robust and flexible algorithm that could be easily adapted to other domains. The complete detection algorithm is described in detail below.

3.1. Background estimation

We collect data in non-continuous blocks (referred to as snippets) to lower the duty cycle of the processor. We arbitrarily chose a rate of 4 snippets per second (each snippet being approximately 50 ms in duration at a 4 KHz sample rate). This corresponds to a duty cycle of 20%. We then calculate the median of the absolute value of each snippet. The median filter provides additional robustness against non-gaussian data as well as being computationally efficient.

The median of each snippet is then used in a secondary running median filter. This gives us an estimate of the noise background. Since each snippet is reduced to only a single value, we can make the running median filter relatively long (corresponding to a significant duration in time) without using an excessive amount of memory. This allows us to adjust the algorithm to be more or less responsive to changes in background noise level depending on operational needs. An additional running median filter is also performed on the deviation between the current snippet median level and the long term noise estimate. This gives us an approximate measure of the mean and standard deviation of background distribution (although we are using order statistics rather than sample averages). We can then use these estimates as a basis for a CFAR detector.

3.2. Automatic gain control

Our chosen platform, the Crossbow Mica2, only has 4 kilobytes of onboard memory so memory resources are highly constrained. This required us to only use 8 of the available 10 bits provided by the A/D converter so that each sample could fit into a single byte. This severely impacts performance, however, due to the wide dynamic range of the observed targets. We needed to judiciously select which 8 bits to keep from every 10 to preserve as much of the dynamic range as possible. The sensor also provided an additional 8 bits of adjustable gain which could be applied as well.

Using bit selection and the 8 bits of sensor gain, we developed an automatic gain control algorithm which successfully preserved the features of interest for all targets. The steps in the algorithm are shown below:

1. Begin collecting snippet, preserving only least significant 8 bits of each sample.
2. If a new sample requires more than 8 bits, preserve middle or upper 8 bits as necessary to prevent clipping. Store position where bit shift occurred and value of new shift (either 1 or 2 bits).
3. Repeat until entire snippet is collected.
4. After snippet is complete, adjust shift of all samples collected prior to any bit shift that occurred during the snippet so that all samples are consistent.
5. Calculate range of snippet (maximum value - minimum value)
6. Calculate gain for next snippet $g_{n+1} = \frac{g_n * r}{\hat{r}}$, where g_n is the gain for snippet n, r is the desired range, and $\hat{r}$ is the current measured range.

3.2.1. Target detection

Estimation of the mean and standard deviation allows the use of a CFAR detector. We choose between the two hypothesis H_0 = no target present and H_1 = target present using

$$\hat{m_n} - b \underset{H_0}{\overset{H_1}{\gtrless}} T\sigma, \tag{1}$$

where $\hat{m_n}$ = the median of the absolute value of the n^{th} snippet, b = the current background estimate, T is a user defined threshold (typically 3.0), and σ = the median deviation of the background.

To reduce false alarms we further require 3 out of 5 snippets to exceed the detection threshold before declaring a detection event. We then record the time that a detection event started, the time of maximum signal strength (corresponding to the closest point of approach or CPA), and the time the detection event ended.

The overall algorithm is summarized below:

1. Collect snippet of data
2. Calculate median of absolute value of snippet
3. Insert median value into running median filter. Output of median filter is an estimate of the noise background.
4. Subtract noise background from current median value. Insert running deviation into a second running median filter. Output of this median filter is an estimate of the standard deviation of the background.
5. If current deviation exceeds user defined threshold times the estimate of the background deviation for 3 out of 5 snippets, then declare detection.

A block diagram of the complete detection algorithm including automatic gain control is show in Figure 3

An example of the detection process is show in Figure 4. The absolute value of the raw collected data is shown first with the results of the median filter, and estimated background noise level overlaid on top. Subplot (b) shows the deviation from the background level and the noise threshold for detection. The filtered detection result is shown in (c).

4. DISTRIBUTED DETECTION

When we deployed this algorithm, we discovered that false alarms were still unacceptably high in noisy environments. To further reduce the false alarm rate we added a distributed detection component. There are many methods for performing distributed detection, however, recent work by Chamberland and Veeravalli[5] suggests that fusion of binary detection decisions is optimal in many cases for constrained networks.

For binary decisions and identical sensors the optimum decision rule is[3, 4]

$$u_0 = \begin{cases} 0 & if \quad \sum_{i=1}^{n} u_i < k \\ 1 & if \quad \sum_{i=1}^{n} u_i \geq k \end{cases}, \tag{2}$$

where $u_i = 0, 1$ is a binary detection decision from sensor i with $u_i = 1$ indicating a detection and $u_i = 0$ indicating no detection. Each decision is independent with identical false alarm, p_F, and detection, p_D, probabilities. The overall probabilities of detection and false alarm are then

$$P_D = \sum_{i=k}^{n} \binom{n}{i} (p_D)^i (1 - p_D)^{n-i}, \tag{3}$$

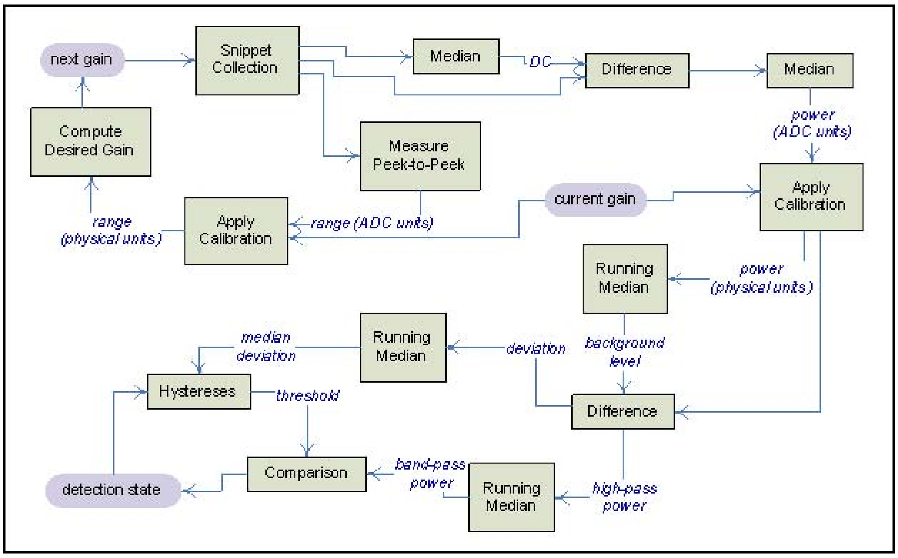

Figure 3. Block diagram of overall detection algorithm

$$P_F = \sum_{i=k}^{n} \binom{n}{i} (p_F)^i (1-p_F)^{n-i}, \tag{4}$$

where

$$\binom{n}{i} = \frac{n!}{i!(n-i)!},$$

and p_D and p_F are the probabilities of detection and false alarm, respectively, for an individual sensor and k is a positive integer. We then chose k to achieve a desired probability of false alarm.

To test this algorithm we deployed four nodes over a 10 x 10 meter area. When any one node detected a target, it would broadcast the event statistics (including start time, the CPA time, and the stop time of the event) to all the other nodes. Detection events from different nodes were associated if the CPA times differed by less than a user defined maximum value. Once two of the nodes had detected the target (corresponding to a 2 out of 4 detection rule), then a confirmed detection result was transmitted to an external monitor. We also attempted to eliminate loud non-moving events by requiring the CPA times between nodes to differ by a minimum value. This value was set consistent with the maximum expected velocity of a target.

The distributed algorithm also allowed us to estimate the speed and direction of the target based on CPA times and the known node positions. This could eventually be used as a further method for reducing the false alarm rate by eliminating targets with speeds or directions that are inconsistent with the known behavior of targets of interest. As noted above, eliminating targets with excessive speeds (which typically correlate with loud events originating far outside the sensor network) has already proved to be effective.

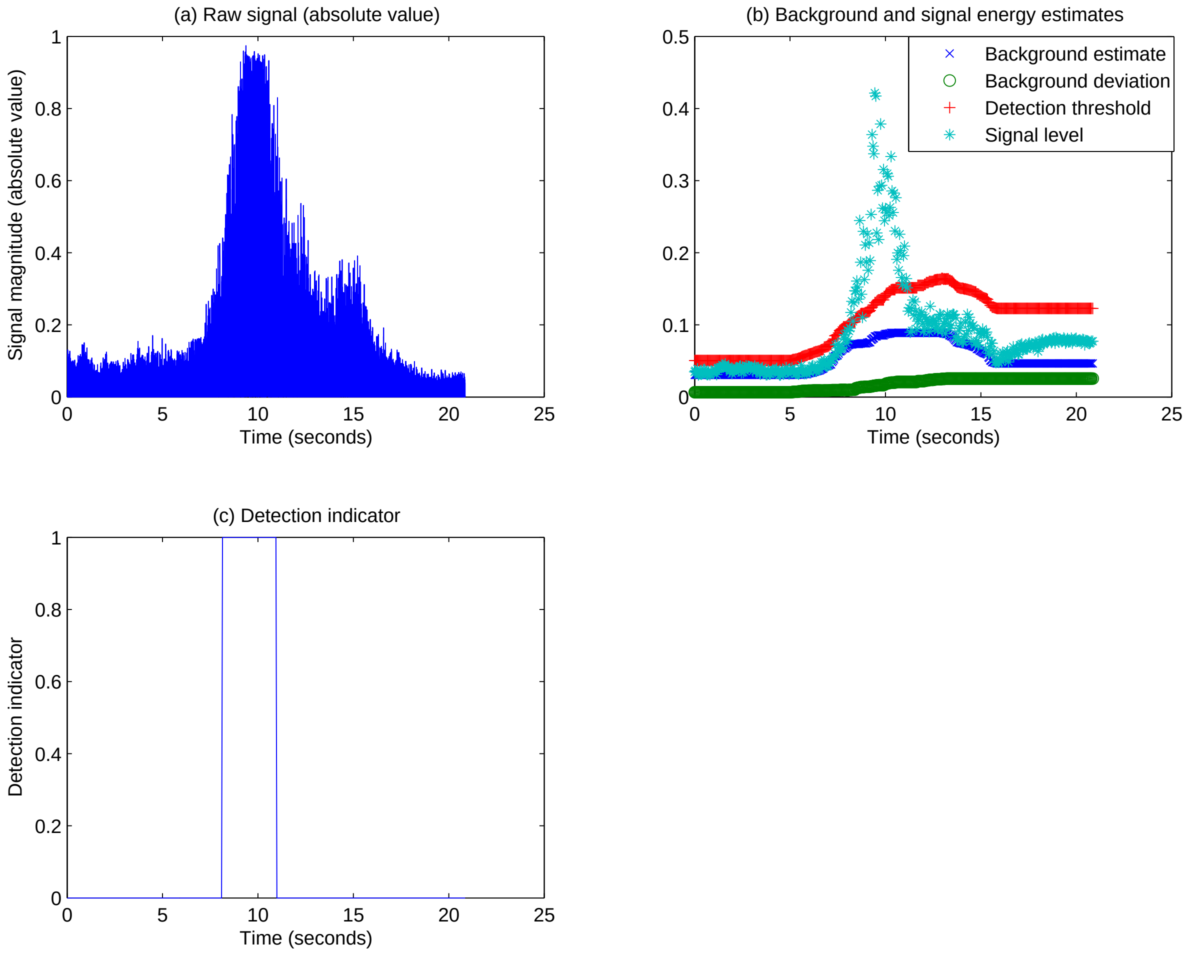

Figure 4. Stages of detection algorithm - (a) Raw signal, (b) Background estimate, noise deviation, detection threshold, and target signal deviation, (c) Detection result

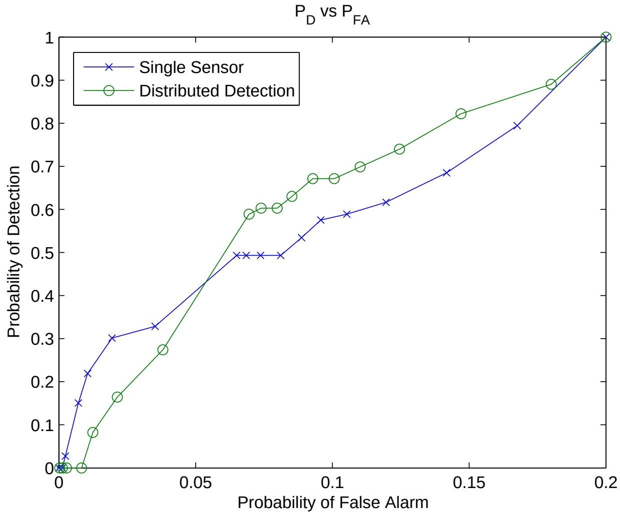

Figure 5. Detection results for individual and fused sensors. Targets included typical civilian vehicles (minivans, trucks, sedans, etc) and commercial delivery trucks.

5. EXPERIMENTAL RESULTS

To test the algorithm we deployed four Crossbow Mica2 motes with acoustic sensors over a 10 x 10 meter area in a parking lot and recorded results over a two hour time period. A simultaneous video collection was performed and analyzed for ground truth purposes. This was used to identify when targets of interest drove by the sensor network. 84 targets were observed during the collection interval. Data was collected in 50 millisecond blocks at a sample rate of 4 KHz. Blocks of data were collected four times a second. Each node used the detection algorithm described in section 3 to perform independent detection of targets. The results were also fused at a central node using the distributed detection algorithm described in section 4. Two out of the four nodes were required to detect a given target before a detection event was declared. A plot of detections versus false alarms is shown in Figure 5 for the individual and fused results.

The results show good detection performance for the individual sensors, but very high false alarm rates. The fused results reduce the false alarm rate somewhat, but not nearly as much as we had hoped. This suggests that most false alarms are acoustic events not of interest which are audible to all of the sensors.

6. CONCLUSIONS

We have demonstrated a robust, low power distributed detection algorithm operating under realistic conditions. The resulting algorithm currently operates with a 20% duty cycle. This could easily be reduced at the cost of a slight drop in probability of detection. An increase in the number of sensor nodes could be used to offset this, however. Automatic gain control and median filters were employed to extend the dynamic range of the sensors and increase the robustness of the detectors under highly nonhomogeneous conditions. A low overhead M out of N distributed detection algorithm was also implemented to reduce the false alarm rate without adversely effecting the detection performance. Future research will include possible use of spectral analysis to distinguish targets of interest versus targets not of interest as well as greater use of kinematics estimates to eliminate targets with unreasonable motion characteristics.

REFERENCES

1. A. Quach and K. Lo, “Automatic target detection using a ground-based passive acoustic sensor,” *Proceedings of the 1999 Conference on Information, Decision and Control* , pp. 187–192, 1999.

2. G. Goodman, "Detection and classification for unattended ground sensors," *Proceedings of the 1999 Conference on Information, Decision and Control* , pp. 419–424, 1999.
3. P. K. Varshney, *Distributed Detection and Data Fusion*, Springer-Verlag, New York, 1997.
4. A. Mamdouh, "Multiple-sensor distributed detection systems with data fusion," *Proceedings of the 13th International Conference on Digital Signal Processing* .
5. V. V. J. Chamberland, "Decentralized detection in sensor networks," *IEEE Transactions on Signal Processing* **51**, pp. 407–416, 2003.