Title
Target Classification and Localization in Habitat Monitoring

Authors
H. Wang
J. Elson
L. Girod
et al.

Publication Date
2003

Peer reviewed

TARGET CLASSIFICATION AND LOCALIZATION IN HABITAT MONITORING

*Hanbiao Wang**, *Jeremy Elson**, *Lewis Girod**, *Deborah Estrin**, *Kung Yao*†

Computer Science Department*, Electrical Engineering Department†, UCLA
Los Angeles, CA 90095, USA
email: {*hbwang, jelson, girod, destrin*} *@cs.ucla.edu**, *yao@ee.ucla.edu*†

ABSTRACT

We are developing an acoustic habitat-monitoring sensor network that recognizes and locates specific animal calls in real time. In this paper, we investigate the system requirements of such a real-time acoustic monitoring network. We propose a system architecture and a set of lightweight collaborative signal processing algorithms that achieve real-time behavior while minimizing inter-node communication to extend the system lifetime. In particular, the target classification is based on spectrogram pattern matching while the target localization is based on beamforming using Time Difference Of Arrival (TDOA). We describe our preliminary implementation on a Commercial Off The Shelf (COTS) testbed and present its performance based on testbed measurements.

1. INTRODUCTION

Recent developments in integrated circuit, wireless communication and Micro Electro-Mechanical System (MEMS) technology have allowed the construction of low-cost low-power small sensor nodes with signal processing and wireless communication capabilities [1]. These nodes can form distributed wireless sensor network systems that could revolutionize diverse sensing applications. The untethered densely distributed sensor network systems could enable non-intrusive micro-scale habitat monitoring that is hard to realize through traditional instrumentation [2]. In this paper, we describe the design and implementation of a wireless sensor network that recognizes a specified type of animal calls and then locates the calling animals.

The fundamental task of our habitat monitoring system has two parts. The first part is to determine whether observed animal calls are of the specified type using their spectrograms. Each type of animal call has its own characteristic spectrogram which is input to the system. The classification of an observed acoustic signal is determined by the maximum cross-correlation coefficient between its spectrogram and the specified characteristic spectrogram [3]. The second part is to locate the calling animal when its call is recognized. The system determines the target location by TDOA-based beamforming [4]. Cross-correlation between waveforms of the same signal recorded by two different sensors indicates TDOA between those sensors. Given locations of multiple sensors and TDOAs among them, the target location can be estimated using least square method.

Besides the fundamental task of classification and localization, the system has a goal of real-time in-network signal processing. Observed acoustic signals are processed inside the network. High-level information about target type and location becomes available for queries in a short time. Because communication is the primary energy consumer in wireless sensor networks, in-network processing is much more energy-efficient than transmitting all raw data to a central node for off-line processing. In addition, we also try to minimize the inter-node data transmission by data reduction and compression.

The rest of the paper is organized as follows. Section 2 identifies challenges of such a system and introduces tentative solutions. Section 3 presents detailed system design. Section 4 describes initial implementation on a COTS testbed and its performance. Section 5 discusses related work. Section 6 describes future work and concludes this paper.

Thanks to the NSF CENS and Intel Corporation for funding.

2. CHALLENGES

We face several challenges in constructing such a system.

The first challenge is the fine-grained time synchronization across audio codecs of sensor nodes. In order to locate a target with an error less than 3 cm, synchronization must be at least within 100 μs because the speed of sound is about 345 m/s. It is hard for a traditional Internet time synchronization protocol such as NTP to achieve such an accuracy in the wireless sensor network context. We choose Reference Broadcast Synchronization (RBS) described in [5]. Briefly, RBS synchronizes a set of receivers of a reference broadcast, in contrast to traditional protocols in which a receiver synchronizes with a sender. RBS achieves significantly better precision than traditional protocols. In our testbed, neighboring nodes can be synchronized within 1.5 μs.

The second challenge is the real-time processing. Acoustic signals are sampled at a rate of several KHz. Therefore it is too time-consuming to conduct target classification and localization whenever a new sample is obtained. We introduce staged event-driven processing in order to achieve real-time behavior. Acoustic data are processed in stages, starting with the most lightweight stage. Data are moved to the next stage for more heavyweight processing only when they pass the current stage of processing.

The third challenge is extending the lifetime of the battery-powered system. Although in-network processing has avoided transmitting raw data to a central database by processing data locally, the beamforming node still needs waveform data transmitted from multiple sensor nodes. We apply data reduction and compression before waveform data are transmitted to a beamforming node. Data reduction lowers data volume by discarding irrelevant information in the waveform data. Data compression encodes the reduced waveform data in a more compact format.

3. SYSTEM DESIGN

This section describes the design details about the system architecture, staged event-driven processing, and data reduction and compression.

3.1. System architecture

All nodes have integrated capabilities of sensing, processing, and communication. During the system initialization, nodes are organized into clusters. Clustering can be achieved automatically by self-assembly [6]. The cluster head is for collaboration and central data processing. All other nodes are for distributed sensing and data preprocessing. Because the cluster head is much more heavily loaded with data processing tasks than ordinary sensor nodes, it makes sense for the cluster head to have more computational resource than ordinary sensor nodes. In addition, GPS is also useful for the cluster head to provide time and location reference to the rest of the network. Location of other nodes can be determined iteratively given a group of reference nodes' locations [7, 8]. The automatic clustering and localization of nodes are currently not implemented in our testbed.

3.2. Staged event-driven processing

All nodes continuously sample acoustic signals and buffer the last several seconds of data. The whole data processing task is divided into three stages. From the fastest to the slowest, they are signal intensity monitoring, target classification, and target localization. Signal intensity monitoring is fast and runs all the time on the cluster head. Target classification using spectrograms is triggered only when the observed signal intensity exceeds the user-specified threshold. Only when the animal call is classified as the specified type, the cluster head estimates the target location using TDOA-based beamforming. Staged event-driven processing saves time and energy because unnecessary processing of irrelevant acoustic events are avoided.

3.3. Data reduction

The cluster head uses data from sensor nodes only to compute TDOAs for beamforming. Thus only time information in the data is needed. We propose a data reduction scheme that reduces data volume by discarding irrelevant information while retaining time information in the data. It is based on the following observation. Sample signs of a signal identify when the signal crosses zero point, and thus contain most time information of the data. [1] This simple scheme can be quite effective for a sufficiently long signal sequence. Give a waveform $\mathbf{a}$, we reduce it to $\mathbf{b}$ as follows:

$$b_i = \begin{cases} +1 & \text{if } a_i > 0, \\ -1 & \text{otherwise.} \end{cases} \tag{1}$$

for $i = 1, 2, ..., N$. Because each sample can be encoded into 1 bit, a raw waveform with a sample size of n bits is reduced by a factor of n. In addition, the reduced data can be further compressed using traditional techniques. We do not discuss data compression in detail in this paper.

The above data reduction scheme catches zero-crossing information of the raw waveform. Thus, the reduced waveform catches the most significant frequency component in the raw waveform. As long as the most significant frequency is from the target signal instead of the noise, the reduced waveform catches the arrival time information of the target signal instead of the noise. Therefore, strong noise must be filtered before data reduction is applied. In our experiments, the TDOA computed using reduced filtered waveforms and that using filtered waveforms are almost identical, differing in only 1 sampling interval.

[1] This was originally pointed out by Dr. R.E. Hudson in our discussion

4. IMPLEMENTATION

In order to evaluate the system design, we implemented a prototype system on a COTS platform and measured its performance in an out-door environment. Preliminary results show that staged event-driven processing and data reduction are effective in realizing real-time target classification and localization.

4.1. Testbed

We selected COMPAQ iPAQ H3760 Pocket PC as the testbed node. It has a built-in microphone. Its audio codec supports 8 KHz - 48 KHz sampling in signed 16-bit integer. Its 206 MHz StrongARM-1110 CPU, 32 MB ROM and 64 MB RAM provide reasonable resource for signal processing. In addition, we add an 11 Mpbs orinoco PC card to each iPAQ. Thus each node has integrated sensing, processing and communication capabilities. We chose the FAMILIAR distribution of Linux operating system [9] for the testbed. The combination of COTS hardware and open-source operating system makes a powerful and convenient development platform. Unfortunately, iPAQ H3760 has no Floating Point Unit (FPU). All floating-point computing is performed in software, which is about 10 times slower than in FPU. To overcome this difficulty, we implement the Fast Fourier Transform (FFT) and cross-correlation calculation completely in fixed-point arithmetic since they are intensively used in classification and TDOA-based beamforming.

4.2. Client server model

A server runs on each sensor node. It continuously samples the acoustic signal, time-stamps each Direct Memory Access (DMA) data transfer from the audio codec, and buffers the last few seconds of data using the "audio server" technique described in [8]. When the server receives a data request with the specified starting time and duration from the cluster head, it fetches the data from its buffer according to the specification, applies filtering and reduction to the data, and then sends the data back to the cluster head.

A daemon runs on the cluster head. It continuously monitors the average intensity for each short segment of signal. When signal intensity exceeds the specified threshold, the daemon temporarily stops signal intensity monitoring. The daemon then collects data locally of the duration of the specified type of animal calls for classification. The classification first analyzes the spectrogram of the observed waveform and then cross-correlates it with the specified reference spectrogram. If the maximum cross-correlation coefficient exceeds a predefined threshold, the observed signal will be classified as of the specified type, and the daemon will create one client thread of data request for each sensor node. These threads run concurrently. When all requested data are available, the daemon locates the target using beamforming and then re-starts the signal intensity monitoring.

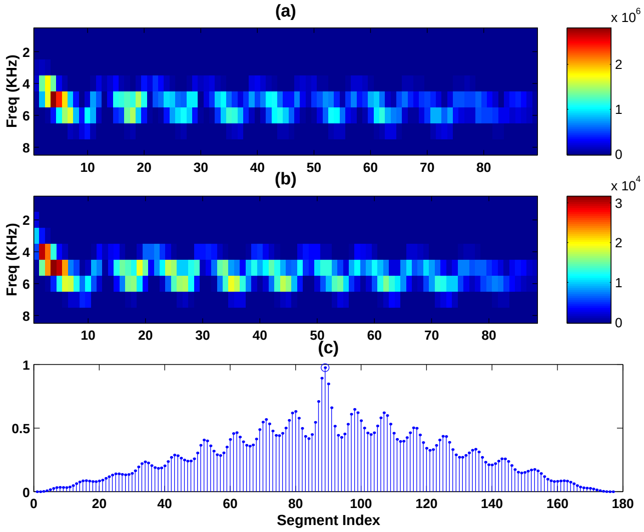

Fig. 1. (a) Spectrogram of observed frog call. (b) Reference spectrogram. (c) Their cross-correlation coefficients

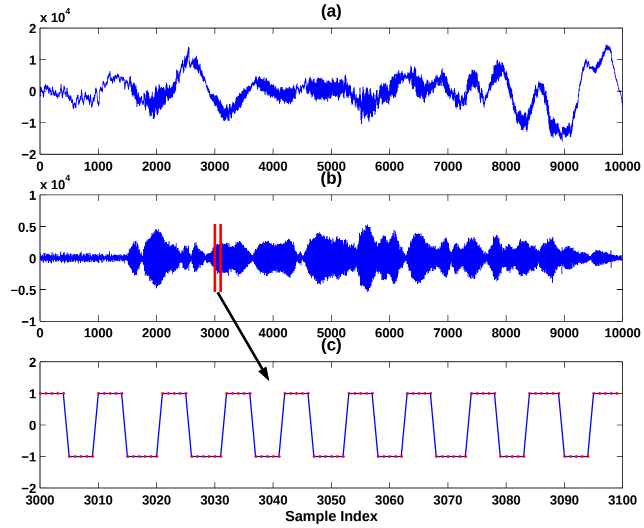

Fig. 2. (a) Raw waveform with wind noise. (b) Filtered waveform. (c) Reduced waveform segment

4.3. System performance

We tested these mechanisms in an outdoor environment. To simulate the target, We played back several different types of frog calls from a Harman/Kardon PC speaker. The performance metrics include the accuracy of classification and localization, the system response time, and the reduction of data transmission.

Cross-correlation coefficients between spectrograms are computed only with shifting along the time axis and only within the bandwidth of the specified type of frog calls. We chose 0.5 as the threshold for classification. The maximum cross-correlation coefficient among different types of frog calls is always less than 0.35 in the experiment. With a signal-to-noise ratio (SNR) of 6 dB within the specified bandwidth, the maximum cross-correlation coefficient between the observed frog calls of the specified type and the reference spectrogram is at least 0.65. When SNR within the specified bandwidth is higher than 6 dB, the system recognized 100% of the specified type of frog call and rejected 100% of other types. Fig. 1 shows an observed spectrogram, the reference spectrogram and their cross-correlation coefficients.

When a sensor node receives a data request from the cluster head, it fetches the requested data from its buffer, filters and reduces the data, and then sends them back to the cluster head. Fig. 2 shows filtering and data reduction applied to a raw waveform. There is a very strong low-frequency noise caused by the wind in the raw waveform. The noise of the wind is effectively filtered without changing the waveform's phase. The filtered data are further reduced by a factor of 16 in the experiment because the raw waveform samples are signed 16-bit integers.

TDOA is indicated by the lag of the maximum of the cross-correlation between two waveforms. Fig. 3 shows the consistency between TDOA by filtered waveforms and TDOA by reduced waveforms. In the experiment, these two TDOAs have a difference less than 100 us, which corresponds to an error of distance difference of 3 cm. TDOA of a frog call between two sensor nodes could be computed using segments of frog call waveforms instead of using whole-length frog call waveforms. The segments must be longer than their TDOA because the TDOA is supposed to be the lag of the maximal cross-correlation of those segments. We chose the segment length to be two time the maximal possible TDOA among any sensor nodes in the cluster. The shorter the segment , the less data sensor nodes need to transmit. In addition, the cross-correlation computation is faster for shorter segments.

For a system of N sensor nodes, there are $N-1$ non-linear equations which relate the target location, the sensor locations, the sound speed, and TDOAs. These equations are linearized using the method described in [4, 10]. Given the sensor locations, the sound speed, and TDOAs, the target location can be estimated using the least square method. We did not use the constraint equation described in [10] because we have not found an effective method to automatically choose a meaningful solution from four roots of the constraint equation. Instead, we deploy more sensor nodes to make the least square estimation over-determined. Results of the experiment show the effectiveness of the above beamforming method. Fig. 4 shows the sensor geometry, three real target locations, and their estimated locations. The estimation is more accurate when the target is inside the convex hull of sensor nodes as compared to other geometric relationships among the target and sensor nodes.

We measured the response time of several operations in the system. Signal intensity within the bandwidth of the specified type of frog calls is estimated for each block of 2 ms acoustic data. Signal intensity monitoring includes data fetching, data filtering and intensity estimation. One iteration of signal intensity monitoring takes only 785$\pm$5 μs, much less than 2 ms. Classification includes spectrogram analysis and cross-correlation. The cross-correlation is very time consuming. One classification operation takes 361$\pm$1 ms. Localization using beamforming includes data transfer from sensor nodes, computation of TDOAs, and the least square estimation of the target location. Computation of TDOAs is very time consuming because it uses cross-correlation. One beamforming operation for a cluster of 5 nodes takes about 500 ms. Therefore, the type and location information can be available about 1 s after a frog call occurs.

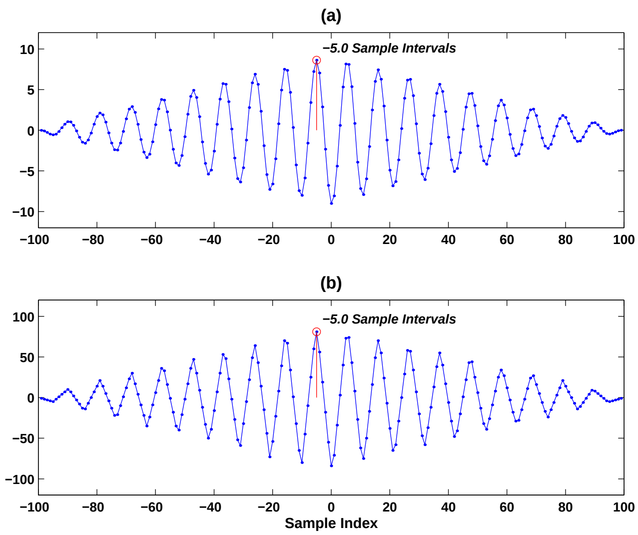

Fig. 3. (a) TDOA using filtered waveforms. (b) TDOA using reduced waveforms

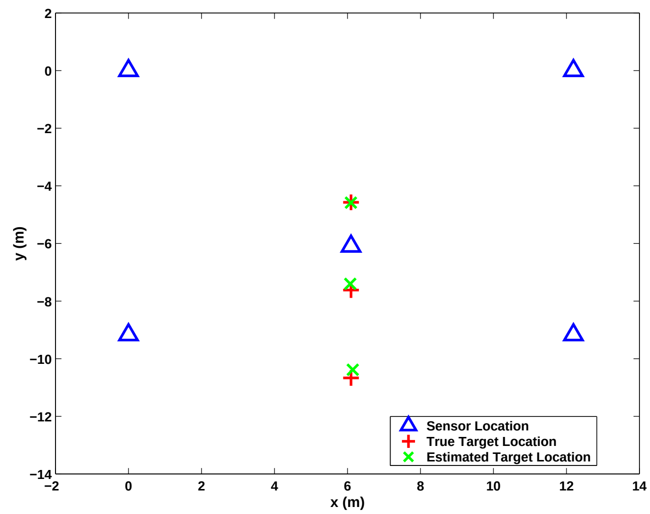

Fig. 4. Sensor geometry, real target locations, and estimated target locations

5. RELATED WORK

A sensor network architecture was proposed in [2] to address requirements of monitoring the micro climate in a habitat. Our system emphasizes monitoring animal calls. A method was described in [3] for animal sound recognition by spectrogram correlation. We use the same method to classify animal calls. In addition, we use classification results to trigger localization and filter irrelevant events. In [10], TDOA based beamforming algorithm was tested with data collected by a synchronized wireless testbed of iPAQs. This paper focuses on system design that enables low-power, real-time realization of such algorithms.

6. CONCLUSION AND FUTURE WORK

This paper describes the design and implementation of a habitat monitoring sensor network that classifies and locates targets in real time. Experiments with the testbed in an outdoor environment validate the system design. Staged event-driven processing effectively enables the system to operate in real time. Data reduction efficiently reduces the data volume by a factor of n, where n is the sample size of raw waveforms.

Our next step is to test these mechanisms on a tiered platform and directly measure energy saving of data reduction. We also plan to investigate multi-target classification and localization in the future.

7. REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–442, March 2002.

[2] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "wireless sensor networks for habitat monitoring," in *Proc. WSNA'02*, Atlanta, GA, USA, September 2002.

[3] D. K. Mellinger and C. W. Clark, "Recognizing transient low-frequency whale sounds by spectrogram correlation," *J. Acoust. Soc. Am.*, vol. 107, no. 6, pp. 3518–3529, June 2000.

[4] K. Yao, R.E. Hudson, C.W. Reed, D. Chen, and F. Lorenzelli, "Blind beamforming on a randomly distributed sensor array system," *IEEE J. on Select. Areas in Comm.*, vol. 16, no. 8, pp. 1556–1567, October 1998.

[5] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proc. OSDI'02*, Boston, MA, December 2002.

[6] K. Sohrabi, W. Merrill, J. Elson, L. Girod, F. Newberg, and W. Kaiser, "Scalable self-assembly for ad hoc wireless sensor networks," in *Proc. IEEE CAS Workshop on Wireless Communication and Networking*, Pasadena, CA, USA, September 2002.

[7] A. Savvides, C. C. Han, and M. B. Srivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in *Proc. MobiCom'02*, Rome, Italy, July 2001.

[8] L. Girod, V. Bychkovskiy, J. Elson, and D. Estrin, "Locating tiny sensors in time and space: A case study," in *Proc. ICCD'02*, Freiburg, Germany, September 2002.

[9] the Familiar Project, http://familiar.handhelds.org.

[10] H. Wang, L. Yip, D. Maniezzo, J. C. Chen, R. E. Hudson, J. Elson, and K. Yao, "A wireless time-synchronized cots sensor platform, part ii: applications to beamforming," in *Proc. IEEE CAS Workshop on Wireless Communication and Networking*, Pasadena, CA, USA, September 2002.