Sensor network based vehicle classification and license plate identification system

Jan Frigo*, Vinod Kulathumani†, Sean Brennan*, Ed Rosten‡, Eric Raby*
*Distributed Sensor Networks Group, Los Alamos National Labs
†Dept. of Computer Science and Electrical Engineering, West Virginia University
‡Dept. of Engineering, University of Cambridge

Abstract—**Typically, for energy efficiency and scalability purposes, sensor networks have been used in the context of environmental and traffic monitoring applications in which operations at the sensor level are not computationally intensive. But increasingly, sensor network applications require data and compute intensive sensors such video cameras and microphones. In this paper, we describe the design and implementation of two such systems: a vehicle classifier based on acoustic signals and a license plate identification system using a camera. The systems are implemented in an energy-efficient manner to the extent possible using commercially available hardware, the Mica motes and the Stargate platform. Our experience in designing these systems leads us to consider an alternate more flexible, modular, low-power mote architecture that uses a combination of FPGAs, specialized embedded processing units and sensor data acquisition systems.**

Keywords: wireless sensor networks, seismic, acoustic vehicle classification, license plate detection

I. Introduction

In this paper, we describe the design, development and implementation of two sensor network applications (1) Real time classification of vehicles using commercial off the shelf Mica motes and Stargates (2) Real time license plate classification using low power cameras and Stargates.

The vehicle classification system analyzes spectral features of real-time seismic and acoustic data as a vehicle approaches and passes the node. We use a low complexity wavelet algorithm for detection using a Mica mote which triggers a higher sampling rate and higher complexity classification algorithm on the Stargate. Real time classification uses the spectral features of the vehicle classes to compute ideal projection vectors for Fisher Linear Discriminant Vector analysis. When tested under conditions similar to the training environment, an accuracy of 99% is obtained almost immediately after the vehicle has passed the detection region of the sensors. We also ensure that vehicles enter the field one at a time. Training the classifier in a dynamic manner to different environments is a much more difficult problem and is a subject of future work.

The license plate detection system uses a camera to capture vehicle images and an efficient learning algorithm to reduce the original image to license plate pixels. This system has a latency of around 5 seconds for processing the image.

Both these applications have some common characteristics: events occur rarely, processing is computationally intensive, event processing needs to be fast and reliable and the system needs to be energy efficient. We note that the existing COTS hardware platforms, the motes and the slightly more resourceful Stargate class of devices allow easy programming and by using a combination of the two we are able to decrease the run-time for the Stargate and therefore be more energy efficient. That said, we also note that specialized embedded processors for heavy computations can make the system more energy efficient. Our experiences with the design of these two applications lead us to explore a more flexible, modular, low-power distributed sensor network system that separates the *real-time* sensor data acquisition, data processing and network communications processing. By implementing high-performance, energy-efficient *processing at-the-sensor*, power savings and improved network response-time can be realized. Implementing and testing the applications described herein on this modular node architecture is a subject of our on-going work.

II. Seismic-Acoustic Based Vehicle Classification System

In this section, we describe a real time vehicle classifier system for traffic monitoring, developed using seismic and acoustic sensors connected to a Mica2 sensor and a Stargate respectively. The goal of this system is to classify vehicles as they approach a specific region into 3 categories: (1) a light-weight vehicle such as a compact car, (2) a moderately heavy vehicle and (3) a very heavy vehicle. For our training and testing we chose a compact *car* [1], a *truck* [2], and a *HumV* [3] as representative vehicles of each category respectively. We assume that vehicles do not enter the monitoring area concurrently.

Challenges: The vehicles travel between 10 to 40 mph and stay within the influence region of the sensors for 8 to 10 seconds. However the spectral signature of a vehicle changes over time. It is only for a very short time of 1 to 2 seconds during which the vehicle is at a sufficient distance

[1] 1994 Honda Accord LX, manual drive
[2] 2006 Diesel Chevy C4500 4x4
[3] 1994 H-1 with a 6.5 L Detroit Diesel Engine

from the sensors such that spectral analysis yields accurate classification. But the classifier has no knowledge as to when the vehicle is closest to the sensor. Yet, the requirement for the classifier is an accuracy of over 99%.

Related work: The use of sensor networks in seismic and acoustic signal processing for vehicle and/or personnel detection and classification is a widely published area of research [15], [8], [11]. Some vehicle detection techniques use a very detailed frequency analysis to determine vehicle weight, number of cylinders and gears, and the type of fuel used by the engine [5]. Vehicle classification has been performed using sensor networks also using acoustic sensors on hardware such as Crossbow's or Moteiv's mote technology [6], [14] along with a custom interface. In [2], the authors describe a vehicle classification system using a network of acoustic sensors using feature vectors formed by spectral characteristics at a local node and then combining the hypothesis at a central base station.

Most of these existing sensor network solutions using spectral characteristics assume a continuous sampling of the order of 1-4 KHz on the acoustic channel. Note that for our application requirements, fast processing is required and therefore an extremely resource constrained device like a Mica mote is not suitable. At the same time, continuous acoustic sampling on a device like a Stargate is not energy efficient. So, here we use a low complexity wavelet algorithm for vehicle detection on the Mica mote and for the higher sampling rate and higher complexity classification algorithm we use the Stargate.

The existing solutions also assume that the closest point of approach is known for a vehicle, at which time the classification is performed. However, in the real deployment scenario, the spectral characteristics vary as the vehicle approaches and passes a sensor, thus, creating false classification results. In our algorithm, we handle this by choosing our feature vectors for different classes in such a way that an order is imposed. So if a vehicle is identified as a series of classes when it is within the region of detection, the class can be chosen as the *highest* in rank.

System Setup: The seismic-acoustic node has a seismic sensor, a 4.5 Hz geophone (GeoSpace Technology, GS-11), placed 50 feet from the roadway to eliminate acoustic feedback in the sensor. The geophone is connected to the Mica2 via a custom signal conditioning interface board and a 16-bit A/D board. The acoustic sensor is a C01U - USB Studio Condenser Microphone placed 10-12 feet from the road and mounted 1 foot off the ground. The microphone is connected directly via a USB port to the Stargate (400 MHz, Intel PXA255 Processor, Linux based). The microphone has directional response and is mounted facing toward the roadway. Samson windshields on the microphones help filter wind noise.

This implementation uses the Crossbow Mica2 mote to trigger an "event" based on seismic information. A Haar Wavelet algorithm is used to extract frequency information from the seismic data. This vehicle event is transmitted to the Stargate processor over the 900 MHz radio link on the Mica2 mote. The Stargate then samples the microphone and processes acoustic information and sends a classification to a base station visualizer (via an 802.11 network). Fig. 1 shows the spectral characteristics for the three vehicle classes. Real time classification is achieved using the spectral features of the vehicle classes to compute ideal projection vectors for Fisher Linear Discriminant Vector analysis [4].

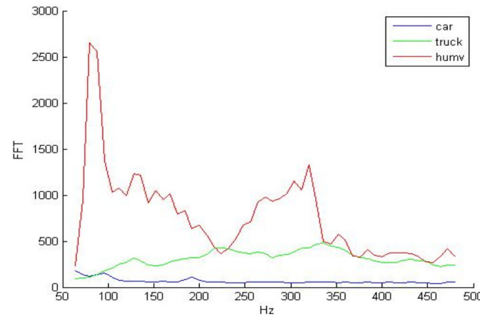

Fig. 1. Mean FFT coefficients for 3 vehicle classes

This method of seismic detection triggering acoustic sampling and processing turns out to be an energy-efficient way of fusing the multi-sensor information to yield a single classification. The seismic detection runs continuously using approximately 60 mWatts. But a seismic sensor alone is not enough. The frequency characteristics for seismic detection show very similar peak frequencies and therefore more frequency analysis would have to be done in order to develop an accurate classification. The Mica2 does not have sufficient computing capabilities to do this. For these reasons, we choose to combine both seismic and acoustic sensors to achieve a more reliable and energy-efficient classification.

Table I shows the energy consumed for this seismic-acoustic based vehicle classification implementation. The acoustic processing uses 4.8V, 4200 mAH capacity NiMh batteries. At this rate, the Stargate based classifier can last for about 3217-4500 vehicle detections before the battery runs out. The seismic detection runs continuously at approximately 60 mWatts, and approximately, 4028 trials (10 seconds each) or 11.1 hours of continuous operation can be achieved under ideal conditions.

	Time	Current	Pwr	Energy
	sec	mamp	Watt	Joule
Seismic Detection	10	20	0.06	0.6
Acoustic classification	420-470	8-10	2.25	18-22

TABLE I
ENERGY FOR SEISMIC-ACOUSTIC PROCESSING

III. VIDEO BASED LICENSE PLATE DETECTION NODE

The video sensor node performs license plate detection of a vehicle traveling on a roadway using a camera, a magnetometer and a Stargate. The system requirements are as follows: to capture a 640x480 pixel image of the aft end of a vehicle at anticipated vehicle speeds of 10 to 60 mph, extract license plate pixels from the original image, thereby reducing the original image by approximately 60-90%, and to convert the license plate image to text with 99% accuracy using optical character recognition (OCR).

Challenges: The vehicles travel between 10 to 60 mph and stay within the field of view of the camera for 2 to 12 seconds. Thus, the image capture routine must be able to store a number of images quickly and select the best image among these to be processed.

Related work: Video based sensor network processing has been implemented on specialized platforms [10] and on the Stargate for many applications such as low resolution image registration [1], fast image motion computation [7], and face detection [13]. Our license plate detection algorithm works by applying a classifier to every pixel in an image to create a rough segmentation of the license place, if it exists. From this, the bounding box of the license plate is found, and that section of the image is then resampled to a fixed size (see Fig. 2). The resampled image is converted to text with the OCR.

System Setup: Our license plate detection system [4] consists of a webcam (with a 12 mm telephoto lens) connected directly to the Stargate via the USB port. A magnetometer (Honeywell HMR2300-232) based detection algorithm is used as a trigger for image capture. A learning algorithm running on the Stargate converts the selected image to license plate pixels only. The sensors and Stargate are mounted approximately 10 feet away from the road and 3 to 4 feet off the ground. To eliminate glare, the assembly is slanted at about a 45 degree angle to the road.

Learning algorithm: The license plate detection software has to be able to process full video images on the Intel XScale Stargate processor. To achieve very high speed video processing we use a technique which takes elements from two algorithms known to produce very efficient classifiers, namely the Viola-Jones [12] object detection algorithm and the ID3 [9] decision tree classifier.

The license plate detection is performed using a machine learning algorithm, trained on labeled data. Briefly, the Viola-Jones detector works by learning a cascade classifier. Each stage of the cascade consists of a weighted sum of weak classifiers learned using the AdaBoost[3] algorithm. If the weighted sum of the classifier results exceed zero, the stage classifies as foreground, otherwise it classifies as background. The weak classifiers are convolution kernels combined with a threshold: if the image convolved with the kernel exceeds the threshold, the classifier value is 1, otherwise it is -1. The kernels consist of a number of rectangular regions of uniform value. The use of the integral image allows the sum of pixel values within a rectangle of arbitrary size to be computed in constant time.

The goal of our algorithm is to not just detect, but also segment license plates, and in some cases, it can consist of a significant fraction of the image. As a result, we have found that a decision tree produces a more efficient classifier. Since decision trees are a richer classifier than cascades (cascaded are degenerate trees), the classifiers at each node need not be as strong, and so they can contain fewer features. This allows us to compute whether the weighted sum exceeds zero using a lookup table. Secondly, because there can be a large number of foreground pixels, the average depth of the tree is lower than the average depth required of a cascade, so pixels are classified in fewer operations.

Bounding box and resampling: The classifier is applied to every pixel of the image, and the number of pixels classified as a license plate in each row and column are recorded. Each row/column has a position and count associated with it, so the median (or any rank) row/column can be efficiently found with a linear search. The license plate is taken to initially lie between the $25th$ and $70th$ percentile horizontally and vertically. The width and height are grown a pixel at a time until the change in rank becomes small.

This algorithm is very robust to misdetections, both in terms of missing regions on the license plate and false detections in the background. It is also very efficient, even when the license plate is a significant fraction of the image, since it avoids relatively expensive operations such as connected components analysis and hole filling. Finally, if the rectangle is unlikely to be a license plate, or contains a license plate that is too small it is rejected. The rectangle must reach a minimum size and pixel density to be considered.

Once the extent and therefore the bounding box of the license plate has been found, the rectangle is then resampled to a fixed size. Size reduction is performed by averaging rectangles of pixels to create a single pixel. This is performed efficiently by using the integral image already computed for the classifier, and the time is only dependent on the size that it is being resampled to. We do not upsample the image: if the bounding box is smaller than the desired size, we use the image within the bounding box without alteration. Once the license plate has been extracted, it is compressed using the standard *libpng* library. Conversion of the license plate image to text is a topic of ongoing research.

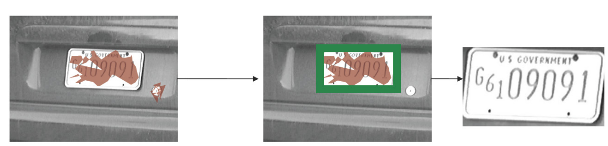

Fig. 2. License Plate Detection Algorithm

On the Stargate XScale license plate detection node, the average run-time performance and energy utilization is shown in Table II. The majority of the energy is consumed by the image capture routine. The idle power for the video node (with the webcam connected to the Stargate) is 1.8 Watts. Approximately, 5582 license plate detection trials can be made with the 4.8v, 4200 mAH NiMh batteries assuming ideal operating conditions.

	Time	Pwr	Energy
	s	W	J
Image capture	5	2.600	13
Detection	0.615	2.375	1.46
Resampling	0.0034	2.352	0.008
Compression	0.0173	2.369	0.041

TABLE II
ENERGY FOR LICENSE PLATE DETECTION PROCESSING

IV. MODULAR NODE ARCHITECTURE

In the previous sections, we described the design and implementation of two sensor network applications built using existing COTS hardware. Both these applications have some common characteristics: events occur rarely, event processing needs to be fast and reliable and requires significant computation and the system needs to be energy-efficient. We note that the existing COTS hardware platforms, the motes and the slightly more resourceful Stargate class of devices allow easy programming and by using a combination of the two we are able to decrease the *on* time for the Stargate and therefore conserve energy. That said, in terms of energy consumption, a more efficient implementation is required. Using custom reconfigurable, embedded processors can provide far more energy-efficiency for such typical sensor network based monitoring applications.

Our proposed node architecture that is suitable for sensor network systems where events are infrequent, but significant computational complexity is required. Moreover, sensor types and processing are usually multi-modal with each type requiring different system resources. Our proposed architecture is designed to be ruggedized for a deployed natural environment, low power, modular, and an experimental platform capable of interfacing to a wide variety of processing and sensor hardware. It is designed for ultra-low power data acquisition as well as *in-network* processing.

The architecture as shown in Fig. 3 combines a low power high performance ARM microcontroller mezzanine board (Phytec phyCOREARM9/LPC3180), an embedded GPS module, and a Texas Instruments CC2431 wireless chip, with four sensor interface options. The mezzanine carrier board's wireless subsystem consists of a single self-contained COTS wireless system on chip (SoC), a CC2431 containing an embedded 2.4GHz 802.15.4 compliant radio, a 32 MHz 8051 microcontroller, 8 KBytes RAM, and 128 KBytes flash storage, as well as hardware accelerated encryption, location computation and MAC layer functionality. Additionally, a GPS module is mounted to the carrier board whose power and operation are controlled by the CC2431's embedded 8051 microcontroller. The CC2431 development tools consist of a C compiler and assembler for straight forward algorithm development.

The carrier board can fully operate without the ARM mezzanine board if a high performance co-processor is not needed. This architecture will allow us to replace the ARM microcontroller, with a DSP, FPGA, or any other low power co-processor as the application dictates.

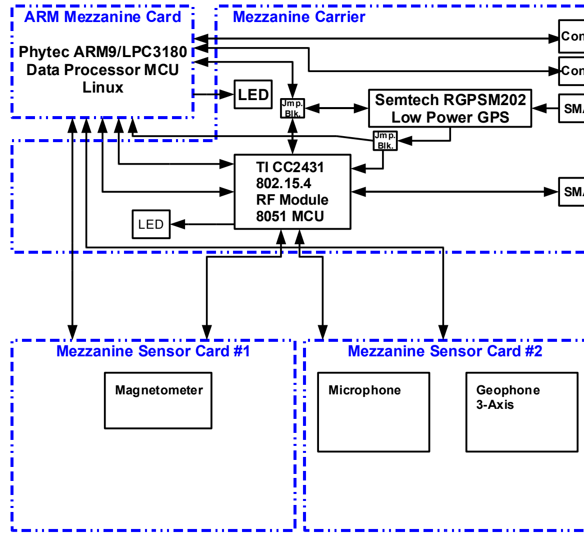

Fig. 3. Proposed modular node architecture

REFERENCES

[1] e. a. Balasubramanian, Harini. Image registration in low resolution visual sensor netowrks. In *International Conference on Information Processing in Sensor Networks*, 2008.

[2] M. Duarte and W. Hu. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64:826–838, 2004.

[3] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and anapplication to boosting. In *Computational Learning Theory*, pages 27–37, 1995.

[4] J. e. a. Frigo. Radiation detection and situation management by distributed sensor networks. In *SPIE Proceedings on Defense, Security and Sensing*, 2009.

[5] R. A. Gramann, M. B. Bennett, and T. D. OBrien. Vehicle and personnel detection using seismic sensors. *SPIE Conference on Sensors, C31, Information and Training Technologies for Law Enforcement, vol. 3577*, Nov. 1998.

[6] e. a. Gu, Lin. Lightweight detection and classification for wireless sensor networks in realistic environments. *SenSys 2005*, Nov. 2005.

[7] X. Lu and R. Manduchi. Fast image motion computation on an embedded computer. In *Conference on Computer Vision and Pattern Recognition*, 2006.

[8] G. Prado. Acoustic-seismic sensors: Past experiences and future prospects. *Processings of SPIE, vol. 5611, p. 117-124*, 2004.

[9] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[10] M. Rahimi and R. Baer. Cyclops: Image sensing and interpretation in wireless sensor networks. *Reference Manual*, Feb. 2005.

[11] e. a. Succi, George P. Problems in seismic detection and tracking. *Processings of SPIE, vol. 4040, p. 165-173*, 2000.

[12] P. A. Viola and M. J. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 511–518, Lihue Kauai, Hawaii, USA, June 2001. Springer.

[13] Y. Weng and A. Doboli. Smart sensor architecture customized for image processing applications. In *IEEE Real-Time Embedded Technology and Applications Symposium*, Aug. 2004.

[14] G. Werner-Allen, O. Marcillo, J. Johnson, M. Ruiz, and J. Less. Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, Mar. 2006.

[15] H. Xiao and et. al. Military vehicle classification via acoustic and seismic signals using statistical learning methods. *International Journal of Modern Physics C, vol. 17, no.2, pp. 197-212*, 2006.