An Efficient Distributed Event Detection System in Wireless Sensor Networks to Detect Real Time Events

Durgesh Shastri[1] Yogeshwari Honna[2]
[1]Assistant Professor [2]M.Tech Student
[1,2]Department of Computer Science & Engineering
[1,2]GNDEC, Bidar, Karnataka

Abstract— Recently, wireless sensor networks (WSNs) [1] have become much more than being simple fine-grained continuous monitoring platforms and become one of the important technologies for event-detecting systems. Event detection functionality of WSNs can be of great help and importance for (near) real-time detection of various events like earthquakes, wild and residential fires etc. In the previous proposed schemes, the raw or the aggregated sensed data is periodically sent to a data consuming center. However, the main drawback of this scheme is that it fails to report the occurrence of an emergency event in a timely manner, as the raw data is gathered by the sensor nodes and later evaluated at the base station to detect an event. Also, in sensor networks it is highly desirable to conserve energy so that the lifetime of the network can be maximized. In this proposed scheme, several sensor nodes collaborate to identify specific events. Since the sensor nodes evaluate raw data directly, communication cost is reduced and events can be detected in real time.

Key words: Wireless Sensor Networks, Real Time Event Detection

I. INTRODUCTION

Wireless Sensor Networks (WSNs) [1] are used in increasingly complex application scenarios such as vehicle tracking [3], undersea monitoring [4], or classification of human motion sequences [6]. In this paper, a system for distributed event detection in WSNs is discussed that allows several sensor nodes to collaborate in order to identify application-specific events. This system is capable of correctly identifying a configurable number of different classes of events, which can be freely trained on the deployed system. Communication cost is reduced to a minimum as raw data is evaluated directly on the sensor nodes. Only small feature vectors of the sampled data, are locally exchanged between nodes; only the information about which event was detected is reported back to the base station. The general approach is to adapt algorithms from the field of pattern recognition to WSNs and train the deployed sensor network to recognize new events. The extraction of features from the data samples is performed locally on each sensor. Hence, this approach combines the energy savings of local data processing with a distributed evaluation to yield high detection accuracy. Further, this system is not tied to any particular application scenario because the pattern recognition algorithms are not specific to the type of sensor used or the characteristics of the deployment area.

II. LITERATURE SURVEY

In [3], Gu et al. describe the VigilNet project which has the objective of tracking vehicles, people and people carrying metallic objects. The sensor nodes employ threshold values on the readings from a magnetometer, a motion sensor and an acoustic sensor in order to classify events. The results are sent to a dynamically assigned group leader for evaluation and tracking. Tracking information about identified events is reported to the base station.

Tavakoli and others [4] consider a scenario in which targets are tracked using an undersea acoustic sensor network. Similar to the previous approach, the sensor nodes report their local classification result to a cluster head which then in turn performs an evaluation of the data and may report the outcome to a base station. Additionally to the number of incoming reports, the cluster head also considers the accuracy of these reports in the past.

The system proposed by Yang et al. [6] is aimed at recognizing human motions. It is a Body Area Network (BAN) consisting of eight sensor nodes attached to the body of a person who may perform one out of twelve actions. Features are extracted from an accelerometer and gyroscope and and classified on each node. If a local classification is promising, the data of all nodes is transmitted to the base station and classified once again. The classification process identifies an action by matching the linear representation of the extracted feature vector to one of several subspaces, each of which corresponds to one type of action.

Wang and his partners [2] describe a habitat monitoring system that is capable of recognizing and localizing animals based on acoustics.

Li and his co-researchers [5] use the example of a coal mine surveillance system to evaluate a WSN that detects events in a 3D environment without relying on threshold values in the raw data.

The distributed event detection system proposed by Martincic and Schwiebert [8] groups the sensor nodes into cells based on their location. All nodes in a cell transmit their data samples to a cluster head which averages the results and retrieves the averages from adjacent cells. Event detection is performed on the cluster heads by arranging the collected averages in the form of a matrix and comparing it to a second pre-defined matrix that describes the event. An event is detected if the two matrices match.

As part of the SensIT project, Duarte and Hu [7] evaluate several classification algorithms in a vehicle tracking deployment. Each sensor node gathers acoustic and seismic data and classifies events using features extracted from the frequency spectrum after performing a Fast Fourier Transform (FFT).

III. SYSTEM ARCHITECTURE

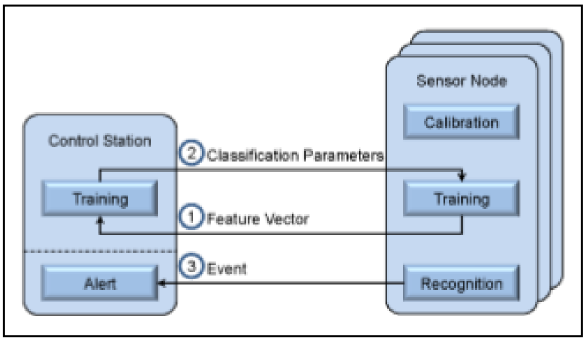

Fig. 1: System Architecture

The two principal components of this system are the sensor nodes and the control station as shown in Figure 1. The sensor nodes are deployed in a way that is suitable to detect the spatial extension of the physical effects of the events. The information about which nodes are adjacent to each other is relevant in our system, because the nodes combine feature vectors from their respective neighbors in the classification. Thus, every node must know its position relative to its neighbors. The control station is required to support and coordinate the training, but is not part of the event detection during the deployment. However, during the deployment, the control station can also be used as a base station for event reporting. The sensor nodes are responsible for carrying out three distinct processes, i.e. calibration, training, and recognition. The calibration phase begins when the user first turns on the deployed sensor nodes. Each sensor node adjusts the attached sensors to the environmental conditions at the deployment site by considering a fixed number of samples to measure the background noise of the observed physical quantity. The training phase is usually initiated by the user via the control station as soon as all sensor nodes have been deployed in the field. The main purpose of this phase is to train the WSN to recognize specific events by providing a set of training events for each event class. Once the training phase completes, the system enters into the detection phase where the sensor nodes share their locally extracted features with other nodes in the surroundings via broadcasts. Each sensor node combines the features from multiple sensor nodes into a feature vector which is then either classified or rejected. Detected events are reported to the base station of the WSN.

IV. METHODOLOGY

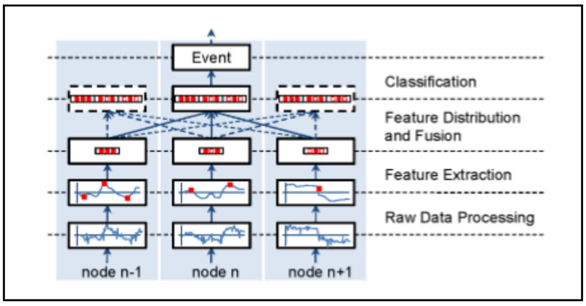

Fig. 2: Event Detection Process

The event detection process can be broken down into the following stages:

A. Pattern Recognition

It is a process of classifying the raw data i.e. assigning a set of data samples to one of the multiple classes. The process can further be subdivided into the following steps:

1) Sampling: The sensor gathers the object of interest that describes the data and pre-processes it to eliminate the background noise.
2) Feature Extraction: Features are extracted from the data to be combined into a feature vector. The main purpose here is to reduce the dimensionality of the data while preserving all characteristic information.
3) Classification: In this stage, feature vectors are classified, either using a priori knowledge from a previous training session or by statistical means.

B. Training

In order to remove the noise from the data stream, it is necessary to pre-process sampled data points so that desired features can be extracted from the raw data generated by a training event. Now using the threshold values established during the calibration phase, the data segment can be segmented into portions corresponding to event candidates. To achieve the highest possible accuracy, the segments are normalized to the WSN platform word size range so that features with known values can fit into a fixed range. Finally, features should be extracted from the normalized segments of raw data.

C. Feature Selection

The main goal here is to select a set of features that is appropriate to act as a prototype for each event class. When the training is complete, the control station calculates how many feature vectors were received from each node. If this value is below a configurable threshold for more than one of the events, the features are discarded by the algorithm because it finds that the physical effects of the events are not strong enough to trigger the segmentation. The other remaining features are evaluated based on their ability to differentiate between the trained classes of events.

D. Distributed Event Detection

The steps involved in Distributed Event Detection are somewhat similar to those used in the training phase. The samples are gathered by the sensors at a preset frequency and the data is pre processed, segmented and normalized. Only those features that are used in the prototype vectors are extracted. Once all the required feature vectors are received, each node performs a fusion of the features by combining the feature vectors based on the bitmask and the relative position of the sender. Only those nodes are able to detect an event whose view of the event matches the view trained at the fixed node.

E. Event Rejection

Since there are a number of events occurring in a WSN, the system needs a way to reject feature vectors which cannot be matched to a prototype vector with required level of possibility. The basic idea behind event rejection is to check whether a feature vector is present in a prototype region. Those feature vectors which are present in the prototype region are classified as matching the corresponding prototype vector and those feature vectors which are not present in the specified region are rejected.

F. Event Reporting

Once the task of successfully classifying the feature vector as belonging to an event class is done, the next step is to report it to the base station of the WSN. The energy of the nodes can be saved by configuring the WSN to only report high priority events, and ignore the remaining.

V. IMPLEMENTATION

In this project, Network Simulator 2 is used as a simulation tool. It contains an open source code that can be modified and extended. It is an object oriented, discrete event simulator for networking and provides best support for simulation. It is written in c++. For command and configuration interface, Otcl interpreter is used. When executed, Otcl instructions are translated to machine code by the interpreter.

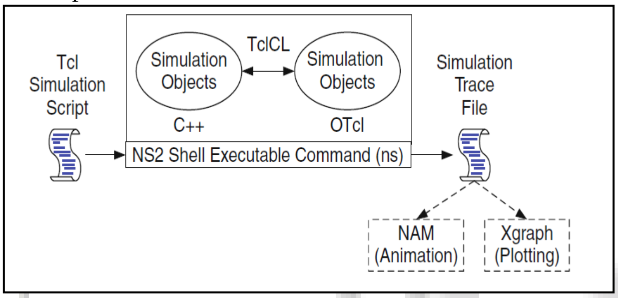

Fig. 3: Basic Architecture of NS2

Various parameters are used to evaluate the performance of this system.

Simulator	NS2
Front End	Otcl
No. of sensor nodes	43

Table 1: Parameters used

VI. RESULTS AND DISCUSSION

We use the X-Graph to evaluate the performance of the system. Following evaluation metrics are considered to do the same.

- Throughput
- Delay
- Packet Delivery Ratio
- Overhead
- Energy

The below table shows the parameters that are being considered. The resulting graphs are also subsequently plotted.

Parameters	Existing System	Proposed System
Throughput (kbps)	12.16	12.16
Delay(msec)	1118.32	376.493
PDR (%)	0.71	0.89
Overhead(load)	13.9	9.5
Energy	5.14	2.5

Table 2: Comparison of both approaches

X-Graph is a plotting utility that is provided by the ns. It allows us to create postscript, Tgif files and others. It can be invoiced within the tcl command which thus results in an immediate display after the end of the simulation.

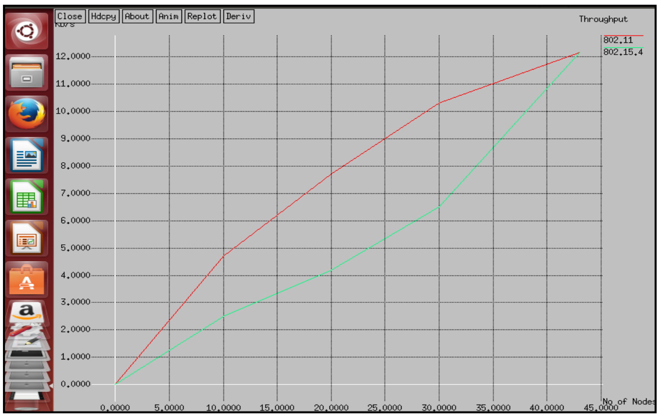

Fig. 4: Throughput comparison graph

Fig. 4 shows throughput for existing and proposed system. The goal is to have both systems reach same throughput. The red line represents the throughput for the existing system and the green line represents the throughput for the proposed system.

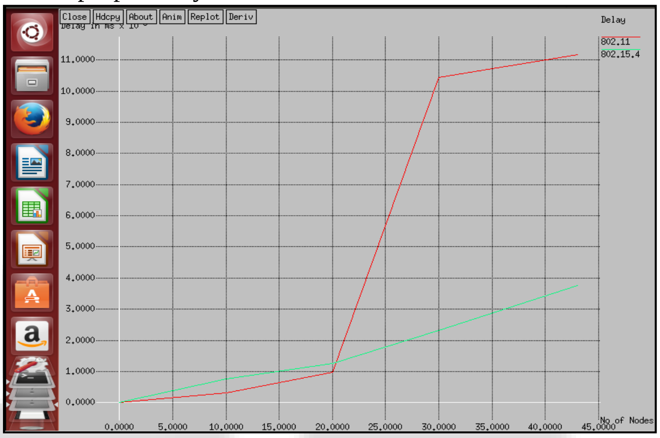

Fig. 5: End to End Delay Comparison Graph

Fig. 5 shows the delay present in both the systems. The end to end delay is calculated using difference in sent and received time, measured in mili seconds or micro seconds. The delay is highly reduced in the proposed system thus making the system more efficient and reliable.

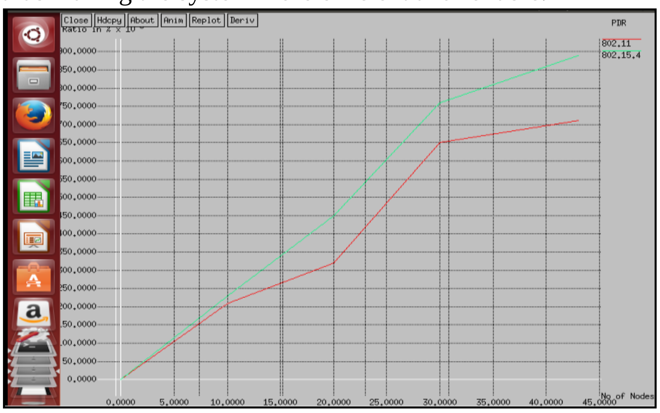

Fig. 6: Packet Delivery Ratio

Fig. 6 shows the packet delivery ratio of both the existing and the proposed system. The PDR is the ratio between the received packets by the destination and the generated packets by the source. The number of packets received by the base station in the proposed system is higher than the number of packets received by the base station in the existing system.

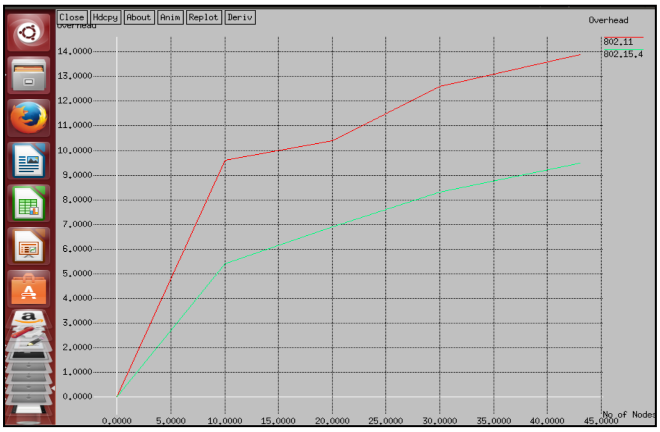

Fig. 7: Overhead Comparison Graph

Fig. 7 shows the overhead calculated for both the systems. The overhead has been reduced comparatively in the proposed system.

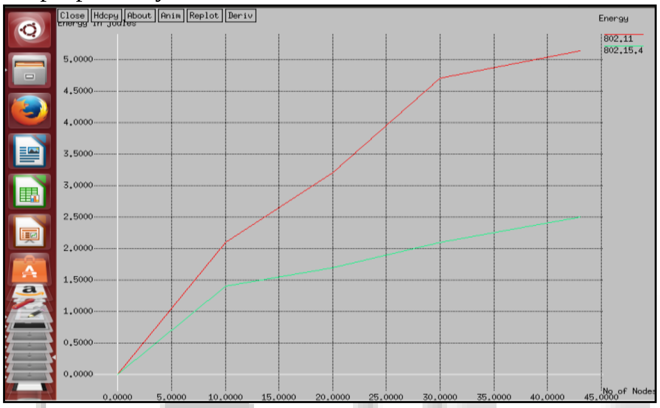

Fig. 8 Energy Comparison Graph

Fig. 8 shows the energy consumption of the two systems. It can be clearly seen that the energy consumed by the proposed system is far less than the energy consumed by the existing system. Energy consumption is one of the major factors responsible for efficient working of the system.

VII. CONCLUSIONS

In this paper, a system to detect events using WSN has been proposed that allows several nodes to work together to find out which specific event has occurred. This system allows us to detect an event accurately along with an efficient utilization of resources. Using a large scale deployment in a realistic setting, it can be shown that this system is a reliable option for correctly identifying different classes of events that have been previously trained and recorded. The overall goal of this system is to achieve high level of accuracy that is suitable for production–level deployments for construction site and freight terminal monitoring.

ACKNOWLEDGMENTS

We are indebted to the management of GNDEC, Bidar, for excellent support in completing this work at the right time. A special thanks to the authors mentioned in the references.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A Survey on Sensor Networks. IEEE Communications Magazine, 40(8), Aug. 2002.

[2] H. Wang, J. Elson, L. Girod, D. Estrin, and K. Yao. Target Classification and Localization in Habitat Monitoring. In Proc. of IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP '03), Hong Kong, China, Apr. 2003.

[3] L. Gu, D. Jia, P. Vicaire, T. Yan, L. Luo, A. Tirumala, Q. Cao, T. He, J. A. Stankovic, T. Abdelzaher, and B. H. Krogh. Lightweight Detection and Classification for Wireless Sensor Networks in Realistic Environments. In Proc. of 3rd Intl. Conf. on Embedded Networked Sensor Systems (SenSys '05), San Diego, CA, USA, Nov. 2005.

[4] A. Tavakoli, J. Zhang, and S. H. Son. Group-Based Event Detection in Undersea Sensor Networks. In Proc. of 2nd Intl. Workshop on Networked Sensing Systems (INSS '05), San Diego, CA, USA, June 2005.

[5] M. Li, Y. Liu, and L. Chen. Non-Threshold based Event Detection for 3D Environment Monitoring in Sensor Networks. In Proc. of 27th Intl. Conf. on Distributed Computing Systems (ICDCS '07), Toronto, Canada, June 2007.

[6] A. Yang, S. Iyengar, S. S. Sastry, R. Bajcsy, P. Kuryloski, and R. Jafari. Distributed Segmentation and Classification of Human Actions Using a Wearable Motion Sensor Network. In Proc. of IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW '08), Anchorage, AK, USA, June 2008.

[7] M. F. Duarte and Y. H. Hu. Vehicle Classification in Distributed Sensor Networks. Journal of Parallel and Distributed Computing, 64(7), July 2004.

[8] F. Martincic and L. Schwiebert. Distributed Event Detection in Sensor Networks. In Proc. of Intl. Conf. on Systems and Networks Communications (ICSNC '06), Tahiti, French Polynesia, Oct. 2006.