Towards Distributed Event Detection in Wireless Sensor Networks

Norman Dziengel, Georg Wittenburg, and Jochen Schiller

Abstract—**Distributed event detection in wireless sensor networks (WSNs) is the process of observing and evaluating an event using multiple sensor nodes without the help of a base station or other means of central coordination and processing. Current approaches to event detection in WSNs transmit raw data to an external entity for evaluation or rely on simplistic pattern recognition schemes. This implies either high communication overhead or low event detection accuracy, especially for complex events.**

In this paper, we present our currently on-going work on a system for distributed event detection that particularly suits the specific characteristics of WSNs. Adapting traditional pattern recognition algorithms to highly embedded devices, it uses the distributed sampling of sensor nodes to optimize the accuracy of the event detection process. Four different algorithms for distributing, classifying and fusing "fingerprints" of the raw data sampled on each sensor are proposed and quantitatively evaluated in a small-scale experiment.

Index Terms—**Wireless Sensor Networks, Distributed Event Detection, Pattern Recognition**

I. Introduction

WIRELESS sensor networks (WSNs) consist of battery-powered miniature computers, that sample physical properties of their environment with diverse sensors and use radio communication to exchange data among themselves or with a base station. Event detection in a WSN comprises the steps of gathering raw data from one or several sensors, recognizing a previously learned pattern in the raw data and mapping this pattern to an event that is semantically relevant to the application of the WSN. For instance, with the approach presented in this paper, a WSN employing nodes with acceleration sensors attached to a fence can recognize the acceleration patterns caused by a burglar climbing over the fence and alert the proprietor. The key idea in our approach is that the detection accuracy of a group of collaborating sensor nodes is superior to that achievable by individual nodes. Hence, the existing redundancy in a densely deployed WSN is leveraged to improve the accuracy of the overall event detection. Furthermore, data fusion techniques as used in our system are considered beneficial for reducing the energy consumption and thus extending the lifetime of WSNs [8].

Manuscript received April 21, 2008.

Norman Dziengel, Georg Wittenburg, and Jochen Schiller are with the Department of Mathematics and Computer Science, Freie Universität Berlin, Takustr. 9, 14195 Berlin, Germany (phone: +49-30-83875116; fax: +49-30-83875194; e-mail: {dziengel, wittenbu, schiller}@inf.fu-berlin.de).

In current research, very high-performance and thus large and energy-inefficient sensor nodes are employed for distributed event detection [2], while other approaches make use of a central basis station [12] or several additional micro-servers [1]. Furthermore, WSNs based on low-power sensor nodes currently either rely on basic threshold values in order to define simple events [13, 11] or focus on the discovery of event patterns [9] rather than the detection of specific events as we do. Hence, distributed event detection in WSNs currently either implies the use of complex hardware or of simplistic or special purpose pattern recognition algorithms.

Our system *Patrec* is capable of self-contained, distributed pattern recognition and event detection on low-power sensor nodes without requiring a base station or any other form of external processing. Sensor nodes locally recognize previously learned patterns based on features extracted from the collected raw data. A feature is a characteristic attribute which describes a pattern observed in the raw data. Simple features include attributes such as duration or average value. In our approach, we use features based on histograms of the raw data which are highly descriptive while keeping computational overhead low. After local data collection is complete, the nodes proceed to exchange their data with other nodes that were triggered by the same event. Data can be fused by exchanging different combinations of information on extracted features or feature classification results. In our implementation on the ScatterWeb MSB-430 sensor node [10], we use raw data sampled by the three-axis acceleration sensor.

In the following, we briefly introduce our event detection architecture and describe four different approaches to data fusion as part of the distributed pattern recognition process. We also report on initial results obtained from an experimental evaluation of our system.

II. Local and Distributed Event Detection

The distributed event detection algorithm is divided into four phases. In the first phase, the HELLO phase, all nodes locate neighboring nodes. In the subsequent CALIBRATION phase, the sensors (in our case the accelerometer) are calibrated by establishing the values of their neutral position and, if necessary, bounds to the background noise. In both the TRAINING and the RECOGNITION phases, the evaluation of the raw data follows the classical pattern recognition model according to [3, 7]. Key components to this process are preprocessing, segmenting and feature extraction as well as

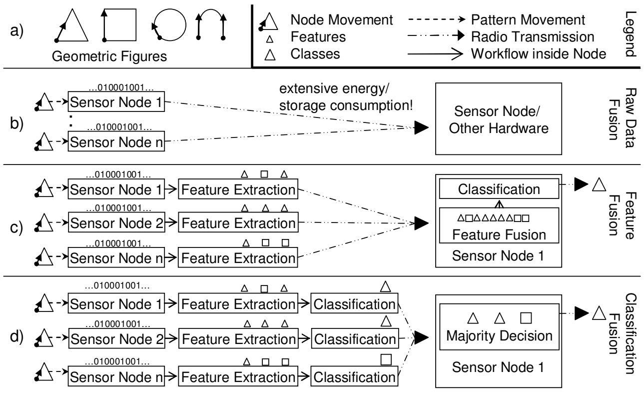

Fig. 1. *Feature* and *classification fusion* is applied for energy efficiency.

classification. In order to handle different amplitude ranges and durations the segmented acceleration data is normalized. Extracted features are normalized to make them comparable in the multi-dimensional feature space. The final classification maps the feature vector to one of the trained classes.

The TRAINING phase is used to learn new reference patterns for future runs of the RECOGNITION phase. During the supervised training, the sensors of the node are exposed to typical raw data values for each application-specific class of events. The features of each class as extracted from the raw data are then sent to the neighboring nodes. Each node stores the received features together with the features extracted from the local raw data. The combined data is used by each node to create a reference feature vector for each class from the concatenated feature vectors of all involved sensor nodes. During the RECOGNITION phase, the raw data is processed in a similar manner and classified using a Prototype Modeler [5] via comparison of its Euclidean distance in the feature space to all the previously stored reference feature vectors.

The Omnibus Model [8] defines several fusion techniques. In our context the *raw data fusion* and *soft-decision fusion* are considered. In comparison, raw data fusion as used in [12] has the disadvantage of large amounts of data that need to be transmitted thus leading to a heavily reduced lifetime of sensor nodes, see Fig 1. b). An additional problem of collecting raw data of all involved nodes is the limited per-node storage. We decided to use soft-decision fusion, which works on condensed raw data in the form of features or classification data. The soft-decision approach allows fusing sensed data within the WSN, thus leading to a self-contained event detection system with acceptable requirements on storage and power.

Both *feature fusion* and *classification fusion* are supported soft-decision techniques implemented in Patrec. In the classification fusion, the locally determined classes of the individual sensor nodes are used to make a majority decision, see Fig 1. d). The majority decision corresponds to a simplified weighted evaluation, similar to the approach presented in [6]. The feature fusion adds the feature vectors of neighboring nodes to a combined feature vector, see Fig 1. c). This vector is then classified by the same algorithm that is also

Fig. 2. Comparison between local and distributed event detection.

used for local classification but uses the concatenated and higher dimensional feature vector of the distributed training and distributed recognition.

Patrec also supports combinations of the fusion methods described above. The *cooperative fusion* is similar to the classification fusion, but falls back to the feature fusion, if no majority can be established based on the collected local classification results. This requires an additional radio request for the feature vectors of the other participating nodes. The *cooperative fusion with veto* actively requests additional feature vectors from other nodes, if a configurable fraction of nodes has reported a deviating local classification.

III. Initial Results

In order to compare the quality of both local and distributed event detection we used the metric of recognition *accuracy*. Accuracy is defined as the ratio of correctly classified events and all events.

In a supervised training, Patrec was taught four different two-dimensional geometrical shapes, see Fig. 1.a). Each pattern was trained ten times, which according to our experience is a suitable amount of training sets beyond which overall event detection accuracy of the system does not improve significantly. Once the training had been completed, we conducted our experiments with three sensor nodes, each handled by a different test person. The size of the group was kept small to ensure repeatability of the evaluation process, and the training was conducted without the participation of the test persons in order to avoid any possible influence.

The three test persons were told to simultaneously draw one of the previously trained geometrical shapes with the sensor nodes by moving them according to the requested shape. The group completed 160 experiments for distributed event detection, which corresponds to 480 runs for local event detection on the individual sensor nodes. Hence, in the context of distributed event detection, the distributed event consisted of three individual local events. We sampled all movement patterns at 50 Hz and extracted six histogram features out of the collected acceleration data. A movement pattern was generated by moving the node along a preprinted stencil. Interrupting the motion indicated the end of the pattern. Further details of the experiments are described in [4].

We use our previous work [13] as a reference for evaluation which, like our current work, also employs the ScatterWeb MSB-430 sensor node. It reports on data of 90 test runs in

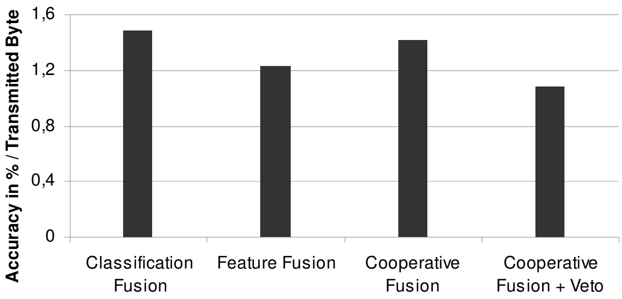

Fig. 3. Cost-benefit analysis of fusion methods in Patrec.

which a WSN is used to detect a person climbing over a fence. In relation to [13], we were able to increase the accuracy for both local and distributed recognition by 37%, see Fig. 2.

This increased accuracy is due to the Prototype Modeler in our pattern recognition system. The local classification scheme results in an accuracy of 89%, which is further increased by the four different methods of the distributed classification resulting in accuracy between 93.8% and 96.3%. Classification fusion performs worst, since this method is based on sensor data that is the most compressed. The feature fusion approach has a slightly higher accuracy, while the cooperative methods do not have a considerable impact.

In order to capture the tradeoff between accuracy and communication cost, we evaluated the system in a cost-benefit analysis. In our considerations, the costs are defined as transmitted bytes and the benefit is defined as the accuracy of the event detection. This gives us a good idea whether a higher accuracy can only be reached with a disproportional amount of transmitted data. The amount of data is calculated as specified in [4] and reflects required communication overhead for each of the four fusion methods. As illustrated in Fig. 3, the most efficient fusion is the classification fusion which is thus to be preferred if communication needs to be minimized and lower accuracy is acceptable. The cooperative fusion with veto is inefficient as it requested the feature vectors too frequently in our experiments. We attribute this behavior to the limited number of nodes in our experiments and plan to re-evaluate it during our next deployment. Finally, the cooperative fusion does not need to request the whole feature vector for each event in order to reach an accuracy that is comparable to the feature fusion. Hence, the cooperative fusion represents a good compromise between accuracy and communication costs.

IV. CURRENT WORK

We are currently working on implementing additional features required for an extended field test in the area of construction site surveillance that we intend to conduct in the near future. Missing features include the ability to differentiate between trained and non-trained patterns and collaboration of a dynamic number of nodes triggered by the same event. Both are equally relevant to realistic deployment scenarios because neither can all possible patterns be trained nor can we assume the WSN to be static over time in light of environmental factors.

Furthermore, we plan to evaluate whether the overall accuracy of the system can be further improved by additional feature extraction methods looking at properties of the raw data such as regression parameters or slope. We also need to add a dynamic leader election to the system, possibly similar to the one described in [11], in order to establish which node should be responsible for handling data fusion, event evaluation and reporting to the base station.

V. CONCLUSION

We have presented our ongoing work on our distributed event detection system for WSNs. In contrast to prior approaches, our system is self-contained, i.e. it operates without a central component for coordination or processing, and makes active use of the redundantly placed sensor nodes in the network to improve detection accuracy. Our experimental results show that distributed event detection yields higher accuracy than local detection on a single node. We have implemented and evaluated four different fusion techniques. With the high accuracy of the feature fusion, the high efficiency of the classification fusion and the cooperative fusion as a compromise between the two previous methods, our system presents interesting new alternatives to distributed event detection in WSNs.

REFERENCES

[1] Bokareva, T., Hu, W., Kanhere, S., Ristic, B., Gordon, N., Bessel, T., Rutten, M., Jha, S.: Wireless Sensor Networks for Battlefield Surveillance. In: Proceedings of the Land Warfare Conference (LWC '06), Brisbane, Australia, October 2006.

[2] Brooks, R. R., Ramanathan, P., Sayeed, A. M.: Distributed Target Classification and Tracking in Sensor Networks. In: Proceedings of the IEEE, 91(8):1163-1171, August 2003.

[3] Duda, R. O., Hart, P. E., Stork, D. G.: Pattern Classification. Wiley-Interscience, Chichester, England, October 2001.

[4] Dziengel, N.: Verteilte Ereigniserkennung in Sensornetzen, Master Thesis. Freie Universität Berlin, October 2007.

[5] Kalton, A., Langley, P., Wagstaff, K., Yoo, J.: Generalized Clustering, Supervised Learning, and Data Assignment. In: Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining, New York, USA, 2001.

[6] Chung-Ting, L., Hong, H., Tao, F., Xiao, S.: Classification Fusion in Wireless Sensor Networks. In: Acta Automatica Sinica, Shanghai, 2006.

[7] Niemann, H: Klassifikation von Mustern. Springer-Verlag, Berlin, 1983.

[8] Ruser, H., Puente León, F. (eds.): Informationsfusion - Eine Übersicht. Oldenburg Verlag, 2007.

[9] Römer, K.: Discovery of Frequent Distributed Event Patterns in Sensor Networks, EWSN 2008, Bologna, Italy, January 2008.

[10] Schiller, J., Liers, A., Ritter H.: ScatterWeb: A Wireless Sensornet Platform for Research and Teaching. In: Computer Communications, 28:1545-1551, April 2005.

[11] Wälchli, M., Skoczylas, P., Meer, M., Braun, T.: Distributed Event Localization and Tracking with Wireless Sensors. In: Fifth International Conference on Wired/Wireless Internet Communications (WWIC '07), Bern, Switzerland, 2007.

[12] Werner-Allen, G., Lorincz, K., Johnson, J., Lees, J., Welsh, M.: Fidelity and Yield in a Volcano Monitoring Sensor Network. In: Proceedings of the Seventh USENIX Sympo-sium on Operating System Design and Implementation, Seattle, USA, November 2006.

[13] Wittenburg, G., Terfloth, K., Villafuerte, F. L., Naumowicz, T., Ritter, H., Schiller, J.: Fence Monitoring - Experimental Evaluation of a Use Case for Wireless Sensor Net-works. In: Proceedings of the Fourth European Conference on Wireless Sensor Networks (EWSN '07), Delft, The Netherlands, 2007