Proceedings of the 2005 Winter Simulation Conference
M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.

LEVEL-OF-DETAIL BASED MODELING OF OBJECT MOVEMENT IN TRACKING AND SURVEILLANCE SYSTEM

Jianfeng Peng
Lizhi Yang
Faisal Momen
Jerzy W. Rozenblit

Department of Electrical and Computer Engineering
The University of Arizona
Tucson, AZ 85719, U.S.A.

ABSTRACT

This paper proposes a Level-of-Detail (LOD) based method for modeling object movement in wireless sensor network tracking and surveillance applications. This method addresses the dilemma of achieving maximum information with limited power and bandwidth in wireless sensor network. LOD provides the researchers with various fidelity levels of data to perform such tasks as regional traffic statistics analysis, object classification and clustering, and at the highest level, object behavior analysis. A simulation test bed extracted from the sensor network hardware platform is being built to allow users to easily generate objects with various behavioral models, design three-dimensional sensor deployment layouts and related surveillance environments, test the accuracy and efficiency of tracking, and calculate lifetime of system and power consumption. An illustrative experiment with preliminary results is provided at the end of this paper.

1 INTRODUCTION

Recent advances in micro-electromechanical systems, embedded computing, and lower power RF communication technology have sparked the advent of massively distributed wireless sensor networks. These sensor networks consist of a large number of small, low cost, low power sensor nodes, which collect and disseminate environmental observational data. Tracking and surveillance in wireless sensor networks has received great interest among researchers in the last few years with the emergence of these massively distributed wireless sensor networks.

Object tracking in a multi-modal wireless sensor network surveillance system, however, is a complicated problem due to the nature of these sensor devices. The limited computational power, battery life and network bandwidth, all impose constraints on the efficacy of the sensor network. Researchers thus have been trying to coordinate such large networks while dealing with these inherent limitations. While most work has been focused on collecting and transferring data across the network via the limited availability of network bandwidth, little has been done on how to best utilize the collected data and provide users with maximum information in tracking and surveillance applications.

We propose a LOD based method for modeling object movement in the confined space under tracking. The method models object movement at various levels as shown in Figure 1:

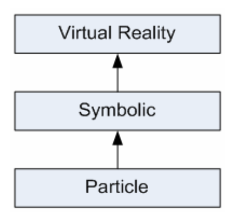

Figure 1: Level of Details

At the lowest level, the target being tracked is represented as a mass point or particle that provides such information as position, movement speed and direction. In the next level, the target is represented by abstract symbology which carries additional information such as sex, approximate height and age provided by the data fusion engine. Symbolic visualization is ideal for users who wish to grasp a high-level understanding of the situation with low computation and communication overhead (Suantak et al, 2001). At the highest level, the symbolic objects are replaced by realistic 3D human models with skeletal animation that not only give users an intuitive impression of the targets' characteristics, but more importantly, the capability to animate the 3D models using skeletal animation techniques to create the level of virtual reality needed for special tracking applications. The sensor network management component works in the backend, dynamically coordinating the sensor network to provide the LOD data while minimizing power consumption and bandwidth usage across the whole network.

The main contribution of the LOD method is providing an efficient way to use relatively short term sensing data from sensor networks to model object activities in virtual environment. By analyzing these data, it is possible to predicate future activities within the confined space using a reasoning engine. This is a new area in wireless sensor network that has not been addressed earlier, but with equivalent significance low power devices or network routing protocols. Applications of LOD based tracking methods range from studies of wild animal behavior, regional traffic statistics analysis, object classification and clustering, to object behavior analysis.

The organization of the rest of the paper is as follows. In section 2, we briefly introduce the unified sensor network architecture and the components. Section 3 describes the Level of Details (LOD) modeling method. Section 4 presents our sensor network hardware platform. Then in Section 5, we introduce our design concept of the software simulation test bed. Section 6 concludes the paper and outlines directions for future work.

2 UNIFIED SENSOR NETWORK ARCHITECTURE

2.1 Architectural Overview

The unified sensor network architecture, shown in figure 2, provides a system framework for the decomposition of the confined space tracking problem and the allocation of resources to research on various components of the system.

At the bottom of the architecture is the physical layer of sensors. On top of the physical layer is the network communication layer that transports data and commands using a standard protocol (Perrig et al. 2001). From the LOD modeling layer's point of view, the physical sensor network components (inside the grey rectangular box) is seen as a black box hiding behind the network layer. The sensor network sends sensory data to the LOD layer, and receives user commands from the network. Tracking is performed by our single/multiple target tracking algorithm with movement prediction. And at the highest level is the visualization layer that provides an interface for users to interact with the whole system.

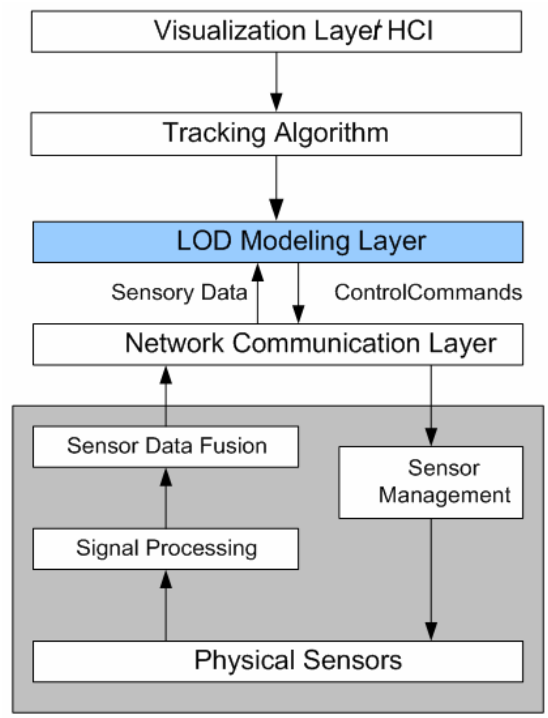

Figure 2: Unified Sensor Network Architecture

2.2 Physical Sensors

The proposed unified sensor network model shown in figure 2 classifies sensor nodes into three different categories based on the functional requirements of LOD: the event-detecting sensors, position tracking sensors and super monitoring sensors.

- The event-detecting sensors consume very little battery power and have a much longer lifetime. They are switched on all the time. Once a sensor detects an event such as temperature or light changes etc., it will activate its neighboring sensor nodes to further track the target. Typical event-detecting sensors are line-breaking sensors or infrared sensors etc.
- The second type of sensors is position detection sensors, such as sonar sensors. Due to the higher power consumption, these sensors will sleep most of the time to minimize power consumption. Once it receives an activation command from neighboring sensors, it will wake from sleep mode and start to track positions of targets and refine the observation.
- The super monitoring sensors are the most powerful ones in the sensor network such as acoustic sensors, high speed video sensors, etc. In order to reduce power consumption, the super monitoring sensors are normally in sleep mode unless they receive an activation command messages from other nodes. The super monitoring sensors collect the largest quantity of

data such as sound, image, and even real-time video. With these power saving considerations, different types of sensors can work together efficiently to extend the lifetime of the whole system.

Using these three types of sensor nodes, users can not only build various heterogeneous sensor networks for different tracking and surveillance applications, but more importantly the can get the data needed for modeling moving objects based on the three levels of modeling requirements of LOD.

At the system boot up stage, the wireless sensor nodes will be automatically registered to its cluster head and the sensor gateway. The event-detecting sensors will be responsible for one particular monitored zone and they will be associated with several position tracking and super monitoring sensors to refine the observation. At the beginning, only the event-detecting sensors are switched on to minimize the system power consumption (He et al. 2004). Once any of the event-detecting sensors detects an event, it will activate the higher level sensors to refine the observation. When the interested object leaves the focus zone, all the monitoring sensors except for event-detecting sensors in that zone will be set to sleep mode to reduce power usage.

2.3 Sensor Data Fusion

The sensor data fusion module in our heterogeneous sensor networks plays an important role. It is responsible for hierarchically transforming between observed parameters and decision regarding the location, characteristics and identities of entities, interpretation of the observed entity, and their relations. Based on the fused data, we can calculate movement trajectory, including position and speed information, for level 1 modeling of LOD. More detailed information such as height and weight can be extracted for LOD level 2 modeling. Finally, by processing audio, image, video and other necessary data, modeling virtual reality in LOD can be achieved.

3 LOD MODELING METHOD

All objects in the LOD modeling method stem from the common base class *CObject* that defines basic attributes and operations. Sub classes derived from *CObject*, such as the *CHuman* shown in the following figure, carry their own specific attributes and operations.

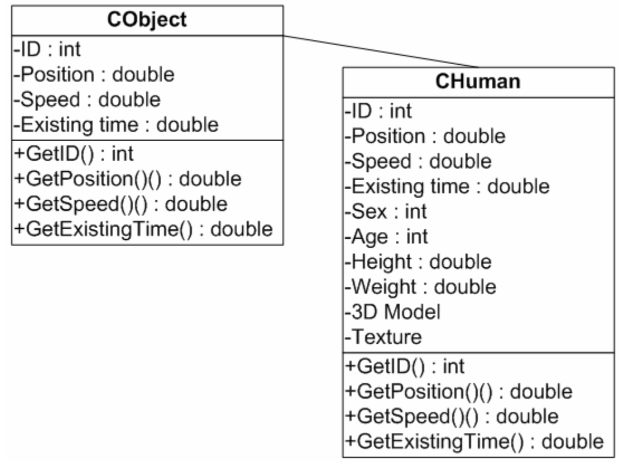

Figure 3: Classes Carrying Different Level of Information

We envision this object-oriented approach as an ideal representation of levels of details visible to users. For instance, at a certain moment, users may become interested in not only the target's position and movement, but would also like additional information about the target being tracked, such as height, weight, and age. Users then initiate a request in the GUI for the desired information, which passes the command to the sensor management layer. The sensor management reschedules the sensors in the field to get the additional information as needed. The underlying LOD layer then loads the corresponding class, *CHuman*, and tracking is performed on this new entity with the added information. The remainder of this section further describes the working mechanism of the LOD layer.

3.1 Data Packet

There are two types of packets being transmitted on the network: the sensor data packets and user command packets. Sensor data packets are variable length structures depending on the content inside the packet. A sensor data packet always contains a header, which defines what kind of data is contained in the packet, and the actual chunk of data. Command packets are simpler, fixed length structures. A command packet contains predefined commands that users will use to perform a certain operation, such as retrieving height information of a target. Along with the command, there are some auxiliary variables in the packet, such as target ID, which identifies the target users desire to operate upon.

3.2 Communication

Communication between the sensor network and the LOD layer is done by sending/receiving the above mentioned data packets between the modeling/visualization layer and the sensor management layer. In our experimental setup, we have created a UDP socket on the server that runs the LOD modeling layer and Visualization layer. The socket then listens to a specific port. As soon as a new data packet comes in to the port, the LOD modeling layer captures it, processes it, and sends it to the tracking layer. Communication from users to the sensor management is performed in a similar way by writing command packets to a specific port. On the other side of the network, the listening socket of the sensor management component receives the command and executes it.

3.3 Advanced Data Manipulation

The LOD also has a special feature that allows users to record the current tracking activity into a scene file on the computer. At a later time users can open that file and review the tracking history. What distinguishes this from regular video stream recording is that users can interact with the entities on the scene file. For instance, users can click on an object and retrieve its ID, average speed, and physical characteristics. They can also replay the entity's movement track path or perform behavior analysis such as studying the most frequently visited spot by that object.

4 WIRELESS SENSOR NETWORK HARDWARE PLATFORM

In this section, we first briefly introduce our sensor network hardware platform that is used in confined-space tracking research. In section 5, we discuss the design of a software simulation test bed based on this platform. The purpose of building a software abstraction of the physical platform is to provide the researchers with a virtual environment mimicking the real-world setup to carry out experiments with minimum implementation time and cost. Furthermore, experimental results from the simulations can be used to calibrate the hardware and enhance tracking efficiency by optimizing sensor deployment, object movement prediction, as well as other factors.

4.1 Heterogeneous Wireless Sensor Network

The confined space used for our test bed is a 60m by 60m square area. Four line-breaking sensor nodes, mounted on the periphery of the area, serve as event-detection sensors. Inside the confined space there are a total of *181* sonar sensors that will be used as position tracking sensors for tracking the movement of the target. At a given time instance *t*, there are a set of *k* sensors which are activated or ON and are used in detecting the target. This set of *k* sensors is denoted by $\{S_{ON}(t)\}$. The physical positions of all sensors are known (Clouqueur et al. 2002). One video camera is deployed over this space as a super node. During normal operation, the camera is in sleep mode to save power consumption. When the sensor management modules need imagery data, it turns on the camera. The imaging processing component is responsible for processing video images and extracting pictures of objects. Finally, all the three categories of sensors work together as a heterogeneous sensor network. The sensor models can found in Figure 4:

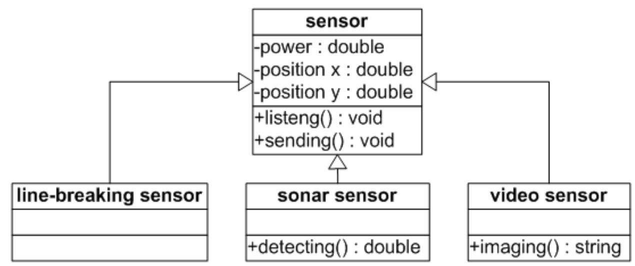

Figure 4: Model of Sensors

4.2 Object Tracking and Movement Prediction

One important task of the data fusion module is to calculate position information of targets and send it to the LOD layer. Thus we designed a prediction-based tracking algorithm that not only guarantees the quality of tracking, but also helps minimize power usage of sensors. By quality of tracking we mean that the system should be able to provide necessary data for modeling object movement. Essentially, based on data from the event-detection and position tracking sensors, the system can project the next position of the object. Based on that prediction, the sensor management module sends commands to activate the position tracking sensors in that region. The detailed algorithm can be found in (Vaidya et al. 2005). This algorithm also works for tracking multiple targets.

4.3 Sensor Management Module

Based on the tracking algorithms described above, the sensor management module in our simulation test bed takes sensory data from sensors and schedules the activation of the sensors. If there are multiple objects present within the sensing range of the sensors, then the sensor managing system keeps track of each object. In addition to this self-organizing mechanism, the sensor management module also takes user commands that tell the management module to retrieve more detailed data or reduce the amount of data. Upon receiving these commands, the management module either turns on additional sensors or turns off some of the active sensors to provide the level of data details needed by users.

4.4 Power Consumption Analysis

A sensor has three basic operation modes: sensing/receiving, transmitting, and sleep mode. Power consumption for these modes are denoted by λ* E_r, λ* E_t and λ* E_s respectively, where λ is the sensor's characteristic parameter. For visual sensors, λ would be higher than for sonar sensors because visual sensors consume more battery power than sonar sensors.

The power consumed by the sensors in the sleep mode or when they are off is assumed to be negligible. Hence, if at any given instance t there are k sensors being used in detecting N objects, then the power consumed for the sensing activities in the surveillance zone is given by:

$$P(t)=\sum_{j=1}^{k}\lambda[E_{r_j}(t)+E_{t_j}(t)].$$

Hence, tracking in the confined space is a collective effort of different modules that work together to track the objects, predict their movement, transfer the data over the network, receive user commands, and reschedule the sensors to provide the level of detail requested by users.

5 SIMULATION TEST BED

5.1 Creating a Visual Experiment

The goal of the simulation test bed is to provide a set of tools for the researchers to create and execute experiments seamlessly in an integrated software environment. Figure 5 shows a snapshot of the prototype under development.
The visual environment allows users to create a simulation by following the steps below:

1. Create a virtual tracking environment with the 3D layout tool;
2. Deploy sensors in the tracking environment by specifying their location, sensor type, initial state, and life span;
3. Load objects with various behavioral models from a database. Movement of objects is controlled by the object data generator.

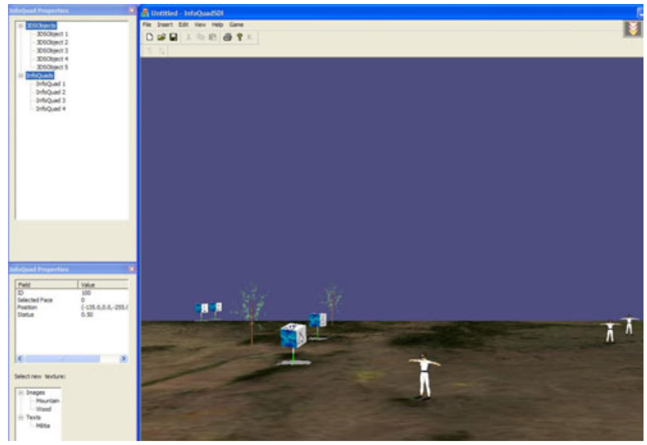

Figure 5: 3D Simulation Tool

5.2 Object Data Generator and Acceptor

In our simulation test bed, we are interest in tracking moving people in a confined space. To simulate moving targets, we build a generator module which is responsible for moving objects around in the confined space based on randomly generated position coordinates. The generator runs on a different machine on the network, and sends the data to the host that runs the simulation. If users want better control of the movement, they can create paths in the 3D environment and assign the paths to the objects. This often happens when users desire to calibrate the sensors or change prediction parameters.

The machine hosting the simulation software is constantly listening to a port. When new data comes in, it is sent to the LOD modeling layer.

5.3 Human-in-the-loop Interface

Human involvement and interactivity in the loop is another signification aspect of the proposed LOD method. The simulation software is created in such a way that users can click and select any object in the scene. When an object is selected, its properties are displayed in a table as shown in figure 5. Some of the properties might not be available depending on the current or selected level of detail. If users want to know more about the object, they can increase the level of details by selecting a specific property (such as *Height*) and changing the state from *Disable* to *Enable*. The simulation software will automatically send a predefined command *GetHeight* together with the selected object's ID to the sensor management layer. When the latter receives the command, it turns on the video camera to take pictures of the object. The image processing system returns the result back to the simulation software, which automatically updates the 3D symbology with the latest height information.

6 CONCLUSIONS

A new Level-of-Detail (LOD) based method is proposed for modeling object movement in wireless sensor network tracking and surveillance applications. This method allows users to customize the sensor network on the fly to get only the data they are interested in. LOD allows users to perform such tasks as regional traffic statistics analysis, object classification and clustering, and at the highest level, object behavior analysis based on the data received. By using the LOD based method, lifetime of a wireless sensor network can be greatly prolonged because redundant sensor nodes are deactivated. A simulation test bed is currently being developed to allow users to easily generate experiments using the integrated software environment. Future work includes completion of the simulation test bed, and further tracking experiments based on the proposed LOD method.

REFERENCES

Clouqueur, T., Phipatanasuphorn, V., Ramanathan, and P., Saluja, K. K. 2002. Sensor Deployment Strategy of Target Detection. *ACM WSNA'02,* 42-48.

He, T, et al. 2004. Energy-Efficient Surveillance System using Wireless Sensor Networks, *The Second International Conference on Mobile Systems,Applications, and Services (MobiSys).*

Perrig, A., Szewczyk, R., Wen V.m Culler, D., Tygar, J. D. 2001. SPINS: Security Protocols for Sensor Networks. In *proceedings of the 7th annual international conference on Mobile computing and networking,* 189-199.

Vaidya, D, Jianfeng, P, Yang, L, Rozenblit, J. W. 2005. A Framework for Sensor Management in Wireless and Heterogeneous Sensor Network. 1*2th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems*, 155-162.

Suantak, L., Momen, F., Rozenblit, J.W., Barnes, M., Fichtl, T., 2001. Intelligent decision support of support and stability operations (SASO) through symbolic visualization. *Proceedings of the 2001 IEEE International Conference on Systems, Man, and Cybernetics*, 2927-2931

Delp, S.L., Loan, J.P., 2000. A computational framework for simulating and analyzing human and animal movement, *Computing in Science & Engineering*, Volume 2, Issue 5, Page(s):46 – 55

Zhang, Y., Ackerson, L., Duff, D. G., Eldershaw, C., Yim, M. H., 2004. STAM: a system of tracking and mapping in real environments. *IEEE Wireless Communications*. 11 (6): 87-96.