Visualization of Environmental Data Generated by Wireless Sensor Networks

Valerie Szudziejka* Oliver Kreylos* Bernd Hamann*

1 Introduction

An emerging area of research in environmental monitoring involves the use of wireless networks of *motes*, autonomous sensor nodes. This "smart dust" is deployable in nearly any environment. Areas that could greatly benefit from the use of motes include energy conservation, preservation of avian species, and determination of structural integrity of buildings. The Smart Dust project at UC Berkeley aims to fit a mote within a cubic millimeter and to make the manufacturing cost negligible – under a dollar per mote [1]. Motes currently in use range in size from a matchbox to a large coin and may cost up to \$100 each; however, a prototype the size of an aspirin and one-tenth the cost may be in production within a year. These motes can be configured to collect and transmit a multitude of data such as temperature and light. Sensor networks can contain a few or a few thousand motes, and when deployed, motes are generally not positioned in a regular fashion. Motes can transmit data synchronously or asynchronously; in order to preserve battery life it is preferable to transmit data asynchronously. We describe reconstruction and visualization methods for two specific mote applications.

2 Scattered Data Methods

Scattered data methods seek to reconstruct a function from samples taken at irregular positions. We have employed two common scattered data methods, Shepard's method and Hardy's multiquadric method, to create reconstructions. Both methods work well but are slow, and hence not appropriate for real-time visualization. Shepard's method uses inverse distance weighting of nodes, such that nodes closer to some arbitrary point have more influence on the reconstructed value. Shepard's method is defined by the formula

$$F(x,y) = \frac{\sum_i \frac{f_i}{d_i^n}}{\sum_i \frac{1}{d_i^n}} \ ,$$

where d_i is the distance, in a Euclidean or non-Euclidean metric, from a point (x, y) to node i, and n is commonly 2. Hardy's multiquadric method uses radial basis functions defining a reconstruction as

$$F(x,y) = \textstyle\sum_i c_i \sqrt{R^2 + d_i^2} \ ,$$

where R is an ad hoc positive real-valued constant and the coefficients c_i are the solution to the interpolatory constraints

$$\begin{bmatrix} \sqrt{R^2 + d_{1,1}^2} & \sqrt{R^2 + d_{1,2}^2} & \cdots \\ \sqrt{R^2 + d_{2,1}^2} & \sqrt{R^2 + d_{2,2}^2} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix} \ ,$$

where $d_{i,j}$ is the distance between nodes i and j [4].

*Center for Image Processing and Integrated Computing (CIPIC), Department of Computer Science, University of California, Davis; {szudziej,kreylos,hamann}@cs.ucdavis.edu

3 Case Study: Great Duck Island

Great Duck Island, Maine, may be one of the largest breeding colonies of Leach's Storm Petrel, a seabird common to the eastern United States [2]. The island is relatively untouched, so researchers wish to investigate the habitat of the petrel without disturbing the birds. Motes provide an ideal, non-intrusive solution. In the summer of 2002, researchers from UC Berkeley placed 32 motes in an area slightly smaller than a football field, recording temperature, humidity, and barometric pressure values. Motes were placed above ground and inside burrows. Although there were few motes, the number of sensor readings between June and October 2002 totaled over one million, making it difficult to analyze the data.

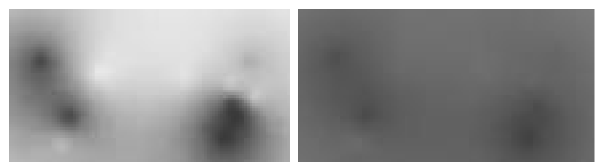

Figure 1: Two visualizations of temperature on Great Duck Island; left: day conditions, right: night conditions.

3.1 Reconstruction and Visualization

The images in Figure 1 were generated using Hardy's local multiquadric method with twenty neighbors for function reconstruction. Dark areas are cooler and light areas are warmer. The measured temperature ranges from about 50° F to 100° F. Note that the temperature recorded is the temperature inside a mote enclosure rather than the temperature in the environment. On a typical day, the temperature in the environment does not reach 70° F. The buried motes recorded nearly consistent temperatures throughout the day, whereas the motes above ground recorded much warmer temperatures during the day than at night.

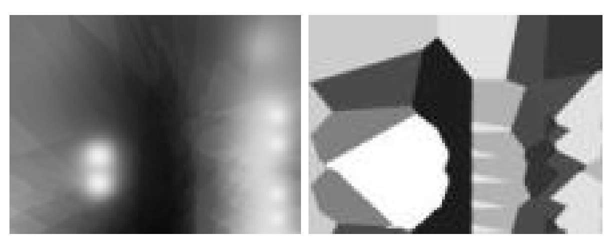

Figure 2: Two interpolations of the data set, with one spatial dimension on the horizontal axis and time on the vertical axis. Left: actual reconstruction, right: Voronoi diagram.

Since all motes were placed close to ground level, it is possible to treat the Great Duck Island data set as a two-dimensional time-varying data set. For visualization purposes, we can alternatively interpret the data set as a three-dimensional data set as sensor readings are weighted based on their distance in space and time. The data set is extremely dense in the temporal dimension but sparse in spatial dimensions. Since the data set is not a true three-dimensional data set but rather a $(2 + 1)$-dimensional data set, an animation is the most appropriate visualization tool. Some example movies are located on the website `http://graphics.cs.ucdavis.edu/~valie/sensor.html`. Two frames from two different movies are shown in Figure 2. Both images are taken at the same point in time and space. The image on the left uses Shepard's local method with twenty neighbors for reconstruction, and color corresponds to temperature, light and dark representing warmer and cooler respectively. The image on the right is a Voronoi diagram [5] such that each colored tile corresponds to a particular sensor. The dark area in the center of the image on the left correspond to the darkest sensor in the Voronoi diagram. Notice how the area of influence from the darkest sensor expands near the top as the lightest sensor stops reporting sensor readings and fades out of relevance.

4 Case Study: Cory Hall

Cory Hall, a building on UC Berkeley's campus, is the subject of research involving "smart energy," a potential breakthrough concept in energy conservation. Sensors can be placed in the environment, monitoring lighting and temperature, or can be attached to a particular appliance, analyzing its usage. Sensors theoretically can passively monitor or actively control energy consumption. If sensors are used to actively control energy consumption, energy efficiency can be dramatically increased by reducing usage during peak times; hence, the cost of energy will drop significantly [3]. In 2001, fifty sensors were installed on a floor in Cory Hall to monitor temperature and light.

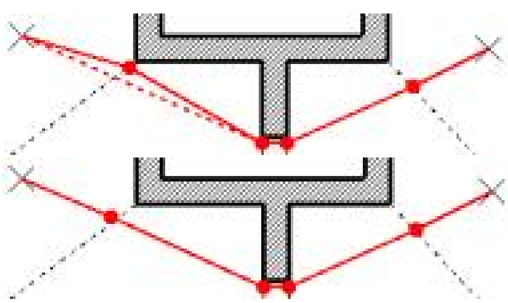

Figure 3: Two stages of the shortest-path algorithm to calculate the distance needed for reconstruction purposes.

4.1 Distance Metric

When reconstructing functions from sensor data inside a building, one has to consider its floor plan. Nodes might be close in Euclidean space, but may be separated by walls or other obstacles. Hence, we need a non-Euclidean metric that takes the building's floor plan into account by weaving through halls and around corners. The complex domain of a building requires us to use a specific data structure and a visual aid to model the floor plan. Each room, hall, or doorway is split into convex polygons, or sectors [6]. In order to visualize functions obtained by reconstruction using standard scattered data methods, it is necessary to know the distances between a location where we wish to approximate the unknown function and certain sensor locations. Figure 3 is a two-dimensional illustration of an approach that finds the shortest path through a complex domain. The floor plan is structured as a graph, with each sector linked to any other sector it is connected to. Thus, it is quite easy to traverse a path from one point to another within the floor plan. We first find a path from the origin to the destination by traversing the graph, and determine which sectors the path passes through. We also calculate a point on each line connecting two sectors, which the path travels through. These points may be manipulated to minimize the distance between the origin and destination, as shown in the lower image.

As we are still putting the framework in place, no visualizations of Cory Hall data are available yet. However, once the framework is in place, little additional work will have to be done, as we can use the same methods applied to the Great Duck Island data set.

5 Future Work

Sensors have already been placed on Great Duck Island for the summer of 2003. More motes are in place, the sensors are more reliable, and mote positions are more accurate. Height above sea level is now available.

The motes used on Great Duck Island are placed statically. The next step in wireless sensor network technology would involve motes that fly or crawl. Existing reconstruction and visualization techniques support moving sensors; however, these techniques may need to be adapted to be used by the motes themselves, so they may self-organize, moving to optimal locations for data reception. Current scattered data methods are limited since they are too slow for real-time data processing. New techniques must carefully sort through the large amount of data received and decide on the fly which data is important and which can be discarded.

References

[1] Wasowicz, L., *Feature: In Sensors Smaller May be Smarter*, `http://www.upi.com/view.cfm?StoryID=20030714-112803-3242r`

[2] *Habitat Monitoring on Great Duck Island*, `http://www.greatduckisland.net/biology.php`

[3] Rabaey, J., Arens, E., et al., *Smart Energy Distribution and Consumption: Information Technology as an Enabling Force*, `http://citris.berkeley.edu/SmartEnergy/SmartEnergy.pdf`

[4] Franke, R., Nielson, G.M., *Scattered Data Interpolation and Applications: A Tutorial and Survey*, in: Geometric Modeling, Methods and Applications, (1991), pp. 131–160

[5] de Berg, M., van Kreveld, M., et al., *Computational Geometry: Algorithms and Applications*, 2nd ed., (2000), New York, New York

[6] Luebke, D.P. and Georges, C., *Portals and Mirrors: Simple, Fast Evaluation of Potentially Visible Sets*, in: Proceedings SIGGRAPH Symposium on Interactive 3D Graphics (1995), pp. 105–106