Contour Map for Event Detection
In Sensor Networks

Jun Yang
CPS 296.1, Spring 2007
Sensor Data Processing
*With contents from W. Xue*

Announcements (Feb. 1)

- The first round of presentations will be finalized by the end of this week
- Course project milestone 1: March 1
- Reading and review for next Tuesday: Model-Driven Processing by Deshpande et al.

Contour map for event detection

Motivation

- Events in real world (e.g., gas leaks in coal mines) are complex spatiotemporal patterns
- Simple selections not sufficient
  - E.g., SELECT ... WHERE gas_density > 0.2
- Simple aggregates not sufficient
  - E.g., SELECT MAX(gas_density) FROM ...

Approach

- Base station periodically computes a contour map for each value being measured over the entire space
  - Hopefully more efficiently than collecting all readings
- Base station supports complex event detection queries over the history of contour maps
  - E.g., SELECT alarm()
    FROM contour_map(temp, 0.1, 0.3) c
    WHERE pyramid(snapshot, "gas_leakage.xml")
    SAMPLE PERIOD 2 min;

What exactly is a contour map?

- Partition space into regions
- Each region spans an area with "similar" readings
- Typically, we partition the domain of all possible values into a number of non-overlapping ranges
- Each region corresponds to one of these ranges, and covers all readings that fall within the range
A slightly more liberal definition

- Not your typical contours!
  - Used by the paper in computing contour maps
  - Post-processing required to convert them back to standard ones
- Within each region, all readings can be approximated by a single function (of location)
  - Example: a linear regression function \( f(x, y) = c_0 + c_1 x + c_2 y \) predicts the reading at \((x, y)\)
- Different regions are approximated by different functions
- What's the point of this alternative definition?

Why a contour map?

- Contour maps can be represented more concisely
  - For each region we only need
    - A function (3 parameters for linear regression)
    - A bound on the average error
    - Boundary
- Contours are amenable to compression as needed
  - Merge regions
  - Compress boundary descriptions
- Contour maps can be computed in-network just like an aggregate function
  - Easily ODI

In-network computation
Merging regions

- Benefit: result region requires fewer bits to represent than original regions combined
- Controlled by two user-specified parameters
  - Max average error (within any region): $\epsilon$
  - Max area of any region (as a fraction of the total area): $p$
- Merge only if resulting region has
  - Area bounded by $p$
  - Average error bounded by $\epsilon$
  - Size (of representation) smaller than the original regions combined
- Greedily merge pairs with highest benefit/cost (size reduction/error bound) ratio

Other tricks

- Compressing boundary descriptions
  - Shared boundaries don’t need to be stored twice
  - With orthogonal polygonal regions, only store every other boundary point
- Snooping
  - Don’t transmit a region if it is subsumed by another one transmitted by a neighbor

Incremental computation

- Each node compares its new partial map with one from the last epoch
- If a region’s boundary stays the same, do not retransmit boundary
  - Same applies to function parameters
  - For error bound, do not retransmit if it decreases
- What if the boundary changes just a little bit?
Experiments

- Synthetic dataset extrapolated from real data

![Graph showing network traffic with labels: QG1, QG2, QG3, QG4, Merge, Merge+compression, Merge+snooping, Merge+incremental, Merge+all.]

Discussion

Course roadmap revisited

- Introduction
- Applications
- Sensor network as a database
- Next theme: more principled use of statistical models in sensor data processing