Management of Large-scale GPS Trace Data: Compression and Query Techniques

Jonathan Muckell
Department of Informatics

Jeong-Hyon Hwang
Department of Computer Science

Catherine Lawson
Department of Geography and Planning

S. S. Ravi
Department of Computer Science
GPS Data

- GPS Trajectory (right):
  - Time-stamped locations

- GPS Devices are Widespread
  - Cell Phones
  - Vehicle Tracking
  - Personnel Tracking

- Large Volume of GPS Data is being Generated
Research Motivation

- Large increase in the amount of GPS data generated
  - Over 68 million GPS devices on the market
  - 116% increase in units sold from 2006 to 2007

Reasons for Compression

- **Reduce Data Size**: Tracking 4,000 objects for one day at 10 second intervals requires 1 Gigabyte of storage
- **Reduce Transmission Cost**: Annual cost of 4,000 vehicles over the course of a year is $1,800,000 to $2,500,000
- **Improve Application Performance**: Time consuming to detect patterns in large datasets
Compressed Approximation

GPS Trajectory Compression approximates the original trajectory using a subset of the original points.

BLUE = the original trajectory
RED = the compressed version

Compression Algorithm Evaluation

We compare algorithms across a consistent set of metrics and datasets to determine algorithm performance.

- Error Metrics
  - Synchronized Euclidean Distance
  - Speed and Heading Error

\[ sed = \sum_{i=1}^{n} \sqrt{(x_{ti} - x'_{ti})^2 + (y_{ti} - y'_{ti})^2} \]
Compression Algorithm Evaluation

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Run Time</th>
<th>Error Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Sampling</td>
<td>$O(n)$</td>
<td>N/A</td>
</tr>
<tr>
<td>Douglas-Peucker</td>
<td>$O(n \log n)$</td>
<td>Spatial Distance</td>
</tr>
<tr>
<td>TD-TR</td>
<td>$O(n^2)$</td>
<td>Time Distance Ratio</td>
</tr>
<tr>
<td>OPW-TR</td>
<td>$O(n^2)$</td>
<td>Time Distance Ratio</td>
</tr>
<tr>
<td>OPW-SP</td>
<td>$O(n^2)$</td>
<td>Time Distance Ratio, Max Speed</td>
</tr>
<tr>
<td>Bellman’s</td>
<td>$O(n^2)$</td>
<td>Spatial Distance</td>
</tr>
<tr>
<td>STTrace</td>
<td>$O(n^2)$</td>
<td>Synchronized Euclidean Distance, Heading, Speed</td>
</tr>
<tr>
<td>SQUISH</td>
<td>$O(n \log \beta)$</td>
<td>Synchronized Euclidean Distance</td>
</tr>
</tbody>
</table>

Selected specific algorithms for comparison based on previous work:

2011 ITS-NY EIGHTEENTH ANNUAL MEETING & TECHNOLOGY EXHIBITION.
Uniform Sampling simply involves taking every $i$th point. In the figure above, every 3$^{rd}$ point is selected.

Advantage: Online, simple & fast

Disadvantage: Loss of sharp changes in speed and direction
Dead Reckoning

- **Basic Idea**: Use previous trajectory information to make prediction. If deviation from prediction is greater than the epsilon, include previous point.

- **Advantages**:
  - Fast, online method

- **Disadvantages**:
  - Error rates are higher under non-aggressive compression ratios & speed metrics
  - Unable to specify a specific compression ratio
Douglas-Peucker Algorithm

- Batch Algorithm: $O(n \log n)$
- Greedy Algorithm – minimize maximum error
- Recursive Algorithm
  - Connect first and last point with a line segment
  - Find point with maximum error (split line into two segments)
  - Recursive call for each line segment
SQUISH Algorithm

- SQUISH = Spatial QUality Simplification Heuristic

- Basic Idea:
  - Buffer of size $\beta$ contains the “best” points
  - Each point contains an estimation of the error that it would introduce if removed from the buffer
  - Point with lowest estimated sed error is removed when new point is inserted
  - Buffer can be implemented with a variety of data structures, however, a heap is the most efficient allowing insertions & deletions in $O(\log(\beta))$ time.
Local Estimation of SED Error

Estimated the SED Error:
1) Connect line segment to adjacent neighbors
2) Calculate SED error if point is removed
SQUISH Algorithm

- **Time 0:**
  - Points are inserted until buffer is full
- **Time 1:**
  - Point with lowest estimated SED error is removed (B)
  - Priority of neighbors updated
  - New point is inserted into buffer (E)
- **Time 2:**
  - Point with lowest estimated SED error is removed (C)
  - Priority of neighbors updated
  - New point is inserted into buffer (F)
GPS Trace Data set

- Microsoft GeoLife Data
  - 65,000 traces
  - Multi-modal transportation data
  - Activities: shopping, going to work, camping, biking, etc..
  - 165 users in Beijing, mostly near Beijing, China over a period of two years
SQUISH outperforms other compression algorithms when at least 10% of the original points remain.
SQUISH Speed Error

- SQUISH outperforms other compression algorithms with respect to evaluating speed error.
Future Work

- Improve heuristic for estimating sed error. Mitigate problems with error rate at high compression ratios

- Effectiveness of SQUISH as a preprocessing algorithm

- Conduct experiments to determine effectiveness of compression on common spatial applications such as:
  - Traffic flow modeling
  - Identification of congestion bottlenecks
  - Identification of speeding violation hot-spots
Mining Frequent Trips

1. Detecting Trips

Frequent Trips Applications

- Backhaul Detection
- Trip surveys
- Personal routine learning
- Route prediction
- Trip arrival time prediction
- Transportation mode learning

2. Cluster Frequent Trips

Method

1. Isolate individual trips and eliminate noisy data
2. Cluster trips according to geographically similar starting and ending locations using a hierarchical clustering method
Historical Freight Movement

Representation of high frequency freight movement, essential in detecting collaborative opportunities

Detect and Rank Backhaul

Detect Backhaul based on intersection of the empty and historical network

Rank based on the frequency and cargo status

Algorithm: Detect Backhaul

If $\text{Historical\_Network} \cap \text{Empty\_Trip} = \emptyset$ then
    print "no viable backhaul opportunity"
Else
    $\text{RANK} = \text{frequency} \times \text{cargo\_status}$
End

2011 ITS-NY EIGHTEENTH ANNUAL MEETING & TECHNOLOGY EXHIBITION.
Backhaul DEMO

2011 ITS-NY EIGHTEENTH ANNUAL MEETING & TECHNOLOGY EXHIBITION.
Backhaul DEMO
Backhaul DEMO

End Time: Wed Aug 20 10:27:37 2008 (Elapsed Time: 0.00 seconds)
Executing (Summary Statistics): Statistics "C:\ArcGIS\Jon Backup\NY_ARCGIS_WORK\August3_9\Thursday\Reliability.gdb \Backhaul_Location" C:\ArcGIS\ArcTutor\Representations \Exercise_5\Representations_5.gdb\TopographicMap \Backhaul_DEMO "Walmart_Freq_Trips_Texas_Frequency MEAN: Walmart_Freq_Trips_Texas_Frequency RANGE" #
Executed (Summary Statistics) successfully.
End Time: Wed Aug 20 10:27:36 2008 (Elapsed Time: 1.00 seconds)
End Time: Wed Aug 20 10:27:36 2008 (Elapsed Time: 2.00 seconds)
Backhaul DEMO
Acknowledgements

This work is supported in part by:

- The Research and Innovative Technology Administration (RITA) of the U.S. Department of Transportation through the Region 2 - University Transportation Research Centers Program

- University at Albany through the Faculty Research Awards Program (FRAP- A).
Thank You - Questions