Automated Road Safety Analysis Using Video Data

Nicolas Saunier

Department of Civil Engineering, University of British Columbia

saunier@civil.ubc.ca
http://www.confins.net/saunier/
Outline

1. Introduction
2. Vehicle Detection and Tracking
3. Traffic Conflict Detection
4. Future Work
1. Motivation

- Traditional road safety is a reactive approach, based on historical collision data.
- Pro-active approach: "Don't wait for accidents to happen".
- Need for surrogate safety measures that
  - bring complementary information,
  - that can be easily collected,
  - are based on more frequent events,
  - are still related to safety (accidents).
- Traffic conflicts (near-misses).
1. Video Sensors

- Main bottlenecks of traffic conflict techniques
  - collection cost,
  - reliability and subjectivity of human observers.
- Advantages of video sensors
  - they are easy to install,
  - they can provide rich traffic description (e.g. vehicle tracking),
  - they can cover large areas,
  - they are cheap sensors.
- Computer vision is required to interpret video data.
1. Computer Vision

- Subfield of Artificial Intelligence.
- The purpose is to program a computer to "understand" a scene or features in an image or a sequence of images.
- Some computer vision systems:
  - optical character recognition, event detection (video surveillance), image database indexing and querying, tracking, object modeling (medical image analysis), human-computer interfaces...
1. Pattern Recognition and Machine Learning

- Required for higher level interpretation.
- Supervised learning: the algorithm learns a function (classifier) that maps input to outputs, given labeled training examples.
- Unsupervised learning: the algorithm models a set of inputs.
1. Modular System

Implement a complete system

Vehicle Detection and Tracking

Traffic Conflict Detection
2. Vehicle Detection and Tracking

- 4 categories of methods:
  - Model-based tracking (often using 3D models),
  - Blob-based tracking (often using background/foreground segmentation),
  - Contour-based tracking,
  - Feature-based tracking.

- Feature-based tracking was chosen since
  - it is the most readily available method (Kanade Lucas Tomasi implementation in Stan Birchfield's or Intel OpenCV Library),
  - it is robust to partial occlusion, variable lighting conditions, and requires no special initialization.
2. **Feature-based Tracking**

- Extension of the feature-based tracking algorithm by Beymer et al. (1997) to intersections (CRV 06):
  - Accuracy between 84.7 % and 94.4 % on 3 sets of sequences.
3. Detecting Traffic Conflicts

- **Input**
  - vehicle trajectories \((x_1, y_1, \ldots, x_n, y_n)\), and velocities \((v_{x_1}, v_{y_1}, \ldots, v_{x_n}, v_{y_n})\).

- **Output**
  - actual traffic conflicts,
  - selected short sequences containing the traffic conflicts for further human review.
3. Traffic Conflicts Description

- Traffic conflicts are rare events. Data is limited for training and test.
- Characteristics:
  - collision course (extrapolation hypotheses),
  - evasive action,
  - actual spatiotemporal proximity.
- Detection should be based on these characteristics:
  - simple manual rules,
  - classifiers learnt on traffic conflict examples.
3. Collision Course Estimation

- Extrapolation hypotheses for motion prediction:
  - default: constant speed and direction,
  - learn (automatically) additional knowledge from the observation of traffic data: typical motion patterns.
- Motion patterns can be stored as trajectories or more complicated models, such as probabilistic models for sequential data (DBNs, HMMs).

Defined by
- prior probabilities,
- transition probabilities,
- output distributions.
3. Learning Motion Patterns and Sequential Data Clustering

- Sequence similarity / distance.
  - ex: Euclidean distance, edit distance, DTW, LCSS.
- Extract a set of features for each sequences, for use with traditional fixed length vector-based clustering methods.
  - ex: leading Fourier coefficients.
- Statistical sequence clustering: sequences are similar if they have a common similarity to a model, computed by the likelihood $P(Observation|Model)$.
3.1. First Approach

HMM-based clustering of trajectories

- iterative K-means approach,
- discard small clusters.

train each HMM on its assigned trajectories

assign trajectories to HMMs
3.1. Semi-Supervised Learning

- An extra training step uses some available traffic conflict instances
  - to adapt HMMs (means and covariances of the Gaussian output distributions),
  - to memorize "conflicting" models.
- Detection process
  - interacting vehicles (close and nearing each other) are detected,
  - the 2 trajectories are assigned to models,
  - if the models were memorized as conflicting, a traffic conflict is detected.
3.1. Experimental Results

- 10 video sequences used for the training of traffic conflict observers (1980s),
- 560 trajectories in 8 sequences used for learning,
- only 5 traffic conflicts.
3.1. Example of Trajectories Clustering
3.1. Detection Results

- HMM-based clustering is very sensitive to initialization.
- Continuum of traffic events.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>CD</th>
<th>Uncertain TC</th>
<th>FA</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;0&quot;</td>
<td>10</td>
<td>17</td>
<td>38</td>
</tr>
<tr>
<td>0.05</td>
<td>10</td>
<td>13</td>
<td>6</td>
</tr>
<tr>
<td>0.10</td>
<td>10</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>0.15</td>
<td>10</td>
<td>12</td>
<td>6</td>
</tr>
<tr>
<td>0.20</td>
<td>10</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>0.25</td>
<td>10</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>0.30</td>
<td>10</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>0.35</td>
<td>10</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>0.40</td>
<td>10</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>0.45</td>
<td>10</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>0.50</td>
<td>10</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>
3.2. Learning Prototype Trajectories

- Keep actual trajectories as prototypes for motion patterns.
- Use Longest Common Subsequence Similarity (LCSS):

  \[
  LCSS_\epsilon(T_i, T_j) = \begin{cases} 
  0 & \text{if } m = 0 \\
  0 & \text{if } n = 0 \\
  1 + LCSS_\epsilon(Head(T_i), Head(T_j)) & \text{if the points match} \\
  \max(LCSS_\epsilon(Head(T_i), T_j), LCSS_\epsilon(T_i, Head(T_j))) & \text{otherwise}
  \end{cases}
  \]

  Two points \( t_{i,k_1} \) and \( t_{j,k_2} \) match if \( |x_{i,k_1} - x_{j,k_2}| < \epsilon \) and \( |y_{i,k_1} - y_{j,k_2}| < \epsilon \).

- Online learning of prototypes.
3.2. Collision Probability

- Prototype trajectories are used for collision probability estimation:

\[ \sum_{i,j} P(T_i|A)P(T_j|B) e^{-\frac{t^2}{2\sigma^2}} \]

where \( t \) is the predicted time of the potential collision if A follows trajectory \( T_i \) and B follows trajectory \( T_j \).

- This is an additional indicator for traffic conflict detection.
- Other uses include detailed exposure measurements unavailable to this day.
3.2. Experimental Data
4. Conclusion and Future Work

- Traffic conflict detection is feasible.
- Collecting more data:
  - other sources,
  - artificial data,
  - interactive labeling, active learning.
- Traffic Intelligence: automatically collect traffic data for traffic diagnosis and management.
Thank you!
**Sequence of Indicators**

**Vehicle Trajectories**

Interaction characteristics: distance, relative velocity, Time To Collision, collision probability...

Detection Method: manually-tuned rules, classifiers learnt on examples...

\[ a_1(t), a_2(t), a_3(t), \ldots \]
Annex. Algorithm Figure

Assignment of trajectories to clusters
Discard small clusters and relearn the others

Assignment of conflicting trajectories to clusters and adaptation of these clusters

Model of usual events

Model of traffic conflicts
3. Influence of the Number of Models

The graph illustrates the average $K_i$ values for different numbers of models ($M = 2, M = 3, M = 4, M = 5$) as a function of $K_i$. As $K_i$ increases, the average $K_i$ generally increases for all model counts, with $M = 5$ showing the highest values overall.