Automated Road Safety Analysis Based on Video Sensors

Nicolas Saunier, Tarek Sayed and Clark Lim

UBC Transportation Engineering Group
Outline

1. Introduction
2. Feature-based vehicle detection and tracking
3. Traffic Conflicts and Collision Probability
4. Experimental Results
1. The Need for 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. A Modular System

- Motion Patterns
- Volume, Origin-Destination Counts
- Driver Behavior...

---

- Image Sequence
- Trajectory Database
- Interaction Database

---

- Traffic Conflict Detection
- Exposure Measures
- Interacting Behavior...

---

- Interpretation Modules
2. Feature-based Vehicle Detection and 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.

Demo
3. Road Safety Analysis
3.1 Traffic Conflicts

- Traffic conflicts are characterized by
  - road users on a collision course,
  - and at least one emergency evasive action.
- Focus on the collision course: "unless the speed and/or the direction of the road users changes, they will collide".
  - movement extrapolation hypotheses are required.
3.1 The Possibility and Probability of Collision

- Given 2 interacting road users, various chain of events can lead them to collide.
- If a collision is possible, the collision probability can be computed, as the sum of the probability of all chain of events that can lead to a collision.
- The collision probability is a (the ?) severity indicator.
- "Better" definition of a collision course.
3.1 Computation of the Collision Probability

\[ P(Collision|A_{1,t \leq t_0}, A_{2,t \leq t_0}) = \sum_{i,j} P(H_i|A_{1,t \leq t_0})P(H_j|A_{2,t \leq t_0})e^{-\frac{(t_{i,j}-t_0)^2}{2\sigma^2}} \]

where \( P(H_i|A_{1,t \leq t_0}) \) is the probability of road user \( A_1 \) to move according to extrapolation hypothesis \( H_i \) (same for \( A_2 \) and \( H_j \)).
3.1 Simple Example

\[ P(Collision) = 0.4 \times 0.7 \times e^{-\frac{(t_1-t_0)^2}{2\sigma^2}} + 0.4 \times 0.3 \times e^{-\frac{(t_2-t_0)^2}{2\sigma^2}} \]
3.2 Motion Patterns

- How to predict road users' movements to compute the collision probability?
- Road users do not move randomly. Typical road users movements, traffic motion patterns, can be learnt from the observation of traffic data.
- Incremental learning of trajectory prototypes.
4. Experimental Results
4. Motion Patterns

128 prototype trajectories (47084 trajectories)

58 prototype trajectories (2941 trajectories)

Video
4. Traffic Conflicts

Demo
4. Collision Probability
Conclusion

- Framework for automated traffic data collection, and specifically traffic safety data.
- Work in progress:
  - improve vehicle detection and tracking: detect shadows, estimate vehicle size.
- Need for more data:
  - other sources,
  - artificial data,
  - interactive labeling, active learning.
Annex
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.
3.2 Motion Pattern Learning

- Similarly to trajectory clustering algorithms, a method to learn motion patterns must address three problems:
  - choose a suitable data representation of motion patterns,
  - define a distance or similarity measure between trajectories or between trajectories and motion patterns,
  - define a method to update the motion patterns.
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 sequence, 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 $P(\text{Observation}|\text{Model})$. 
3.1 HMM-based Motion Pattern Learning

- HMM-based clustering of trajectories
  - iterative K-means approach,
  - discard small clusters.

![Diagram showing HMMs and trajectories]

- 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.

Demo
3.1 Limits

- HMM-based clustering is very sensitive to initialization.
- In reality, there is a 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>”0”</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 Longest Common Subsequence Similarity

Let $\text{Head}(T_i)$ be the sequence $\{t_{i,1}, \ldots, t_{i,n-1}\}$. Given a real number $0 < \epsilon < 1$, the LCSS similarity of two trajectories $T_i$ and $T_j$ of respective lengths $m$ and $n$ is defined as

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

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