Clustering Vehicle Trajectories with Hidden Markov Models Application to Automated Traffic Safety Analysis

Nicolas Saunier and Tarek Sayed

Abstract— **The importance of reducing the social and economic costs associated with traffic collisions can not be overstated. The first goal of this research is to develop a method for automated road safety analysis using video sensors in order to address the problem of a dependency on the deteriorating collision data. The method will automate the extraction of traffic conflicts (near misses) from video sensor data. To our knowledge, there is limited research primarily applied to traffic conflicts. In this paper a method based on the clustering of vehicle trajectories is presented. The clustering uses a k-means approach with hidden Markov models and a simple heuristic to find the number of clusters automatically. Traffic conflicts can then be detected by identifying and adapting pairs of models of conflicting trajectories. The technique is demonstrated on real world video sequences of traffic conflicts.**

I. Introduction

Traffic safety is one of the major world health problems. According to the World Health Organization, 1.2 million people are killed in road traffic crash in 2002, and between 20 millions and 50 millions are injured[1]. Traffic safety diagnosis has been traditionally undertaken using historical collision data. However, there are well-recognized availability and quality problems associated with collision data. In many jurisdictions, the quantity and quality of collision data has been degrading for several years. In addition, the use of collision records for safety analysis is a reactive approach: a significant number of collisions has to be recorded before action is taken. Because of these problems, the observation of traffic conflicts has been advocated as an alternative or complementary approach to analyze traffic safety from a broader perspective than collision statistics alone [1], [2], [3]. The Traffic Conflict Technique (TCT) involves observing and evaluating the frequency and severity of traffic conflicts at an intersection by a team of trained observers. Traffic conflicts are observational situation in which two or more road users approach each other in space and time to such an extent that a collision is imminent if their movements remain unchanged.

Traffic conflicts are more frequent than collisions, and their study can give detailed information about safety. The technique therefore provides a means for the analysts to immediately observe and evaluate unsafe driving maneuvers at an intersection. Whereas monitoring the traffic conflicts occurring in a given location for a few hours is sufficient to assess its safety, the main drawbacks of TCTs are the collection cost, the subjectivity and reliability of observers. Traffic conflicts are manually registered by human observers in the field or by watching video recordings. This can be solved by using video sensors and computer vision techniques to analyze automatically the video data flow.

The goal of this research is to implement a complete system to interpret vehicle interactions and detect traffic conflicts in real world video data. The emphasis is on building a generic, robust and low-cost solution for regular use by traffic safety practitionners. Intersections are crucial parts of the road networks, especially for traffic safety monitoring. Monitoring intersections faces more problems than highways. These problems are related to the highly variable structure of the intersections, the presence of multiple flows of the vehicles with turning movements, the mixed traffic that ranges from pedestrians to lorries and vehicles that stop at traffic lights. Specific classification and occlusion management techniques are required. Common problems to highways and intersections include global illumination variations, multiple object tracking and shadow handling.

Despite the safety implications of such systems, limited computer vision research is directly applied to road safety, and especially the detection of traffic conflicts [4], [5], [6], [7]. This requires a high level understanding of the scene. Such a system is traditionally composed of two modules. Vehicles are detected and tracked by a first module. The reconstituted vehicle trajectories are analyzed by a second module to detect traffic conflicts. The system presented in this paper relies on a feature-based tracking method [8], [9].

Traffic conflict detection is achieved with generic machine learning methods. Vehicle trajectories are sequential data that are naturally handled by statistical models such as hidden Markov models (HMMs), and the more general dynamic Bayesian networks (DBNs) [10], [11]. Traffic conflicts, although more frequent than collisions, are rare events. Therefore, limited training data are available, and unsupervised methods to explore the characteristics of real world dynamic events are needed. However, the goal of this research implies a precise discrimination of the observations, i.e. the detection of traffic conflicts among all traffic events. To achieve this, the supervising of the learning can not be avoided. For that purpose, methods for semi-supervised learning [12] are needed. The approach implemented in this research relies on

- the clustering of vehicle trajectories to build good general models,
- the use of a few traffic conflict instances to identify and adapt the clusters of the conflicting trajectories, called

Nicolas Saunier and Tarek Sayed are with the Departement of Civil Engineering, University of British Columbia 6250 Applied Science Lane, Vancouver BC V6T1Z4, Canada (phone: 1-604-221-4787; fax: 1-604-822-6901; email: {saunier,tsayed}@civil.ubc.ca)

[1] `http://www.who.int/violence_injury_prevention/road_traffic/en/`

conflicting clusters,
- the use of the adapted conflicting clusters to process new traffic data.

Traffic conflicts are therefore described by more general models of the conflicting trajectories than what could be learnt with the available instances. This approach can be seen as the identification of conflicting vehicle streams.

The contribution of this paper includes a k-means approach to HMM-based clustering to deal with the variable length of trajectories, a simple heuristic to find the number of clusters automatically, and a new method to detect traffic conflicts. The following section presents some related work. Section III describes how to cluster trajectories. Section IV presents the method for traffic conflict detection, followed by a demonstration of the approach on real world data in section V.

II. Related Work

Traffic conflict detection based on video sensors can be achieved with various methods. One would first try the direct approach extrapolating vehicle trajectories. The approach described in [7], [13] detects pairs of vehicles that would collide if they maintain their current velocities. It employs a region-based tracking method with background subtraction, and a novel method for three dimensional vehicle size estimation. Tracking and size estimation appear to be accurate, but there is no validation of the collision prediction results. [4] describes an alternative approach based on occupation rates to evaluate the average probability of accident for a passing vehicle, but without direct detection of actual events.

The direct approach is likely to have unreliable performance because the tracking data are imperfect and noisy. Such explicit systems, using rules adjusted by trial and error, rarely provide generic and robust solutions, especially when the environment and the event patterns change. This would for example mean to redevelop the system for every intersection. A better model may be learnt from observation.

A first approach casts the task as a classification problem for which a classifier can be learnt on instances of each class. Traffic conflict detection is seen as a binary classification problem between traffic conflicts and non-conflict interactions. In [6], ”ordinary” traffic vehicle interactions are learnt with HMMs. An HMM is learnt for each type of activity to recognize. The results are very limited because there are very few training instances and the test set is not separate from the training set.

Traffic conflicts, although more frequent than collisions, are unusual events, defined by Zhang *et al.* in [14] as rare and unexpected. Therefore Zhang *et al.* conclude that collecting sufficient training data for supervised learning will often be infeasible. Furthermore, more than one type of unusual event may occur in a given data sequence, which implies that training a single model to capture all unusual events will generally be a difficult task. In the context of the increasing availability of raw video data, it is impossible to rely on human experts to manually process hours of video recordings. Supervised learning techniques for sequential data, like HMMs, require considerable data for the reliable estimation of their parameters.

That is why unsupervised methods are required to learn the characteristics of traffic events. Self-Organizing Maps (SOM) are used in [5] to quantize the vehicle observations and predict the vehicle movements, which is then applied to traffic accident prediction. Traffic conflicts can be detected based on the trajectories of the involved vehicles, i.e. the sequence of the vehicle positions. There are three categories of strategy for the clustering of sequential data [15]:

- In sequence similarity, the comparison between two sequences is viewed as a process to transform a given sequence into another. An edit distance can be based on that principle. This is a typical approach in handwritten recognition, which has also been used for trajectories clustering in [16].
- In indirect sequence clustering, a set of features is extracted from the sequences. In this space, classical vector space-based clustering algorithms can be used. However, the process causes loss of information and needs additional knowledge. For example, the leading Fourier coefficients are used in [17], and are said to be inherently unsuited for the representation of highly complex trajectories.
- In statistical sequence clustering, statistical models like HMMs and more general Dynamic Bayesian Networks (DBNs) are built to describe the dynamics of each group of sequences (with nominal or numerical values).

The arbitrary casting of the problem to the traditional fixed length vector clustering framework is avoided. Sequential data are treated in a more natural way with stochastic models. It is stated in [18] that they can well describe traffic events consisting of a large number of random processes. A DBN was successfully developed to model moving vehicles in a complex freeway traffic scene [19].

Some works have already explored statistical sequence clustering, mainly using HMMs [20], [21], [22], [23]. Most adopt a k-means formulation, which is extended with soft memberships and applied to video data in [20]. The key idea of HMM-based clustering is that observations are defined to be similar in terms of common similarity to a model, expressed through the likelihood function $P(Observation|Model)$.

Model selection in machine learning, i.e. determining the model parameters, especially the number of model components in HMM-based clustering, remains a complicated and uncertain process. Sufficient complexity to interpret the characteristics of the data is needed, but not too much to avoid the overfitting to the training data, which leads to less generalization ability to unkown data. A large number of attempts have been made to estimate the appropriate number of clusters, using heuristics and stopping rules, like the number of observations presented later. Other methods try to optimize some criterion functions under a probabilistic mixture-model framework, such as the Bayesian Inference Criterion [23] and the Minimum Description Length [20]. However, the results are not guaranteed, e.g. in some experiments of [20].

A closely related subject is the unsupervised learning of unusual events [24], [14]. Another related problem is the unsupervised learning of spatio-temporal activity patterns, with methods that have been already mentionned, SOM in [25], HMMs in [26], or clustering of sequences of quantified features [27].

The supervising of the learning can not be avoided to detect traffic conflicts among all traffic events. Vehicle trajectories are clustered and HMMs are learnt on each cluster. These general models are learnt without supervision, and there is therefore no reason for them to be good models of conflicting trajectories. That is why a few training traffic conflict instances are useful to adapt the general models. The resulting ”conflicting clusters” are more general models of conflicting trajectories than what could be learnt with the available training instances. In speech recognition, it is a traditional method to adapt general HMM to each speaker. This was used recently for unusual event detection in [14].

III. HMM-based clustering

The input data to the system are vehicle trajectories. Vehicles involved in traffic conflicts are on a collision course, i.e. vehicles will collide if their movements continue unchanged. To decide if two vehicles are on a collision course, their positions and velocities are required. That is why velocity estimations are included in the trajectories descriptions, whereas most other works use the actual vehicle displacements. A vehicle trajectory is therefore a sequence of four-dimensions vectors, composed of its coordinate (x, y) and the velocity (s_x, s_y) at each time step. An estimation of the size of the vehicles should be also useful, but it didn’t improve the results in the experiments. Since the trajectories obtained through the vehicle tracking algorithm are noisy, they are smoothed using a moving average filter.

The HMM notation used in the rest of the paper is first defined. For a detailed overview of HMMs, readers are directed to [28]. A complete specification of a first-order HMM with M states $\{S_1, S_2, ..., S_M\}$ and a simple Gaussian observation density is formally given by:

- A set of prior probabilities $\Pi = \{\pi_i\}$ where $\pi_i = P(q_1 = S_i), 1 \leq i \leq M$.
- A set of state transition probabilities $H = \{h_{i,j}\}$ where $h_{i,j} = P(q_{t+1} = S_j|q_t = S_i), 1 \leq i, j \leq M$.
- A set of output distributions $B = \{b_j\}$ where $b_j(o_t) = P(O_t = o_t|q_t = S_j) = \mathcal{N}(O_t, \mu_j, \Sigma_j), 1 \leq j \leq M$, where μ_j and Σ_j are the mean and covariance of the Gaussian of state S_j.

where q_t and O_t are respectively the state and observation at time t. It is common to denote a mixture of K HMMs by $\lambda_m = (H_m, B_m, \Pi_m), 1 \leq m \leq K$. Algorithms exist to:

- compute the probability of observing a sequence, given a model.
- find the state sequence that maximizes the probability of the given sequence, when the model is known (the Viterbi algorithm).
- induce the HMM that maximizes (locally) the probability of the given sequence (the Baum-Welch algorithm, an expectation-maximization (EM) algorithm).

The clustering algorithm has an overall classical k-means formulation (see Algorithm 1). The HMMs are initialized in a standard way: uniform priors and transition matrices are used, the means and variances of the Gaussian observation density are estimated using k-means on the observation data. At each iteration, every trajectory is assigned to the HMM that maximizes the probability of the trajectory, and the HMMs are then retrained with their assigned observations.

The number of clusters K is automatically adjusted through the use of a minimum number of assigned observations per cluster, in order to ensure a minimum reliability of the parameters of the HMM. The HMMs that do not satisfy this condition are discarded. The initial number K_i of HMMs should be larger than the ”natural” number of clusters of the data. The learning process stops when the number of clusters doesn’t evolve anymore for a given number of iterations.

Algorithm 1: Algorithm for the clustering of variable length sequences with HMMs.

```
Input: Let $\lambda_m = (H_m, B_m, \Pi_m), 1 \leq m \leq K$ be a
       mixture of $K$ HMMs
       Let $K_i$ be the initial number of clusters
       Let $O = \{O^i\}, 1 \leq i \leq N$ be a set of
       $N$ observations, where $O^i$ is a sequence of $T_i$
       vector observations
       let $N_{min}$ be the minimum number of observations assigned to a cluster to keep and retrain
       it
       Let $n$ be the maximum number of iterations
       without a change in the number of clusters
Output: A mixture of $K_f$ HMMs.
begin
    Initialize randomly $K = K_i$ HMMs $\lambda_m$
    $i = 1$
    while $i \leq n$ do
        Assign each observation $O^i$ to the HMM $\lambda_{m_i}$
        that maximizes $P(O^i|\lambda_m)$
        Let $O_{\lambda_m}$ be the set of $N_m$ observations
        assigned to $\lambda_m$
        Keep only the HMMs $\lambda_m$ such that
        $N_m \geq N_{min}$, reset and train $\lambda_m$ with $O_{\lambda_m}$
        if No HMM was discarded then
            $i = i + 1$
end
```

IV. Detecting Traffic Conflicts

The HMM-based clustering algorithm provides a mixture of K HMMs that model the vehicle trajectories. A few training traffic conflict instances are used to adapt the general models [14]. The parameters of the simple Gaussian observation density, i.e. the means and covariance matrices, are adapted. They are estimated according to the maximum likelihood criterion on the trajectories involved in the training traffic conflict instances, assigned to each cluster. The well-known formulas [29] are used to compute the new means μ_j^c and covariance matrices Σ_j^c of the Gaussian of state S_j, for a given HMM λ_m and a set of N trajectories O^c:

$$\mu_j^c = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T_i} P(q_t = j|O^i, \lambda_m) O_t^i}{\sum_{i=1}^{N}\sum_{t=1}^{T_i} P(q_t = j|O^i, \lambda_m)} \tag{1}$$

$$\Sigma_j^c = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T_i} P(q_t = j|O^i, \lambda_m)(O_t^i - \mu_j^c)(O_t^i - \mu_j^c)^T}{\sum_{i=1}^{N}\sum_{t=1}^{T_i} P(q_t = j|O^i, \lambda_m)} \tag{2}$$

The parameters of the HMM λ_m can then be adapted using the following equations:

$$\mu_j^{'} = (1 - \alpha)\mu_j + \alpha\mu_j^c \tag{3}$$

$$\begin{aligned} \Sigma_j^{'} = \quad & (1 - \alpha)(\Sigma_j + (\mu_j^{'} - \mu_j)(\mu_j^{'} - \mu_j)^T) \\ & + \alpha(\Sigma_j^c + (\mu_j^{'} - \mu_j^c)(\mu_j^{'} - \mu_j^c)^T) \end{aligned} \tag{4}$$

where $\mu_j^{'}$ and $\Sigma_j^{'}$ are the mean and covariance of the Gaussian of the state S_j of the adapted HMM. α is a weighting factor that controls the balance between the original model and the new estimates on the trajectories involved in the training traffic conflicts. Both the original model and the adapted model are kept in the set of HMMs used for detection.

A traffic conflict is an interaction, defined as an observational situation in which two or more vehicles are close enough in space and time, and are nearing each other. So far, the method has built a set of HMMs, among which some are models of conflicting trajectories. But a trajectory is conflicting only with respect to another one. It is their conjunction in time and space that creates a danger of collision. Therefore, the models of the conflicting trajectories (to which the trajectories involved in the training traffic conflict instances are assigned) are memorized by pairs (e.g. pairs of models 4 and 7, 11 and 2, 3 and 7). The traffic conflict detection proceeds as follows:

1) Vehicles are tracked.
2) If two vehicles are close enough (threshold on their distance) and nearing each other (their distance decreases), an interaction is detected.
3) Each interacting vehicle trajectory is assigned to a HMM.
4) If the HMMs of both interacting trajectories were both memorized as conflicting (e.g. any of the pairs of models 4 and 7, 11 and 2, 3 and 7), a traffic conflict between these two vehicles is detected.

V. Experimental Evaluation

A simple vehicle detection and tracking algorithm is used, based on the implementation of the KLT feature tracking algorithm of [8], used in [9]. The advantages of feature-based algorithms include the abilities to work well under different lighting and weather conditions, and to handle partial occlusions, with better results than methods based on background differencing and blob tracking. Kevin Murphy's toolbox is used for HMMs[2].

The evaluation is based on a set of ten traffic sequences on the same location (see Figure 1), initially used for the training of traffic conflict observers in the 1980s. Their length ranges from 10 seconds to 60 seconds. Despite the videotape aging, the approximate alignment of the field of view between sequences and occasional camera jitter, it could be digitized and used to test our method. The training program highlights nine traffic conflicts in nine sequences (see Figure 2). However, only five of them are used as training instances. Another one (fourth sequence) is not used because the vehicles are not detected anymore at the time they are really in conflict, and the rest of the traffic conflicts involve pedestrians and cyclists that are very difficult to distinguish and track on account of the videotape quality. There is no traffic conflict highlighted in the last sequence. It is sometimes difficult to judge the performance of the method since there is no ground truth, apart from the traffic conflicts used in the program.

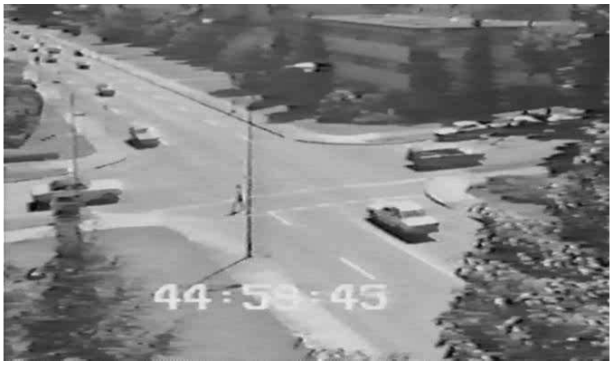

Fig. 1. An image of the traffic sequences.

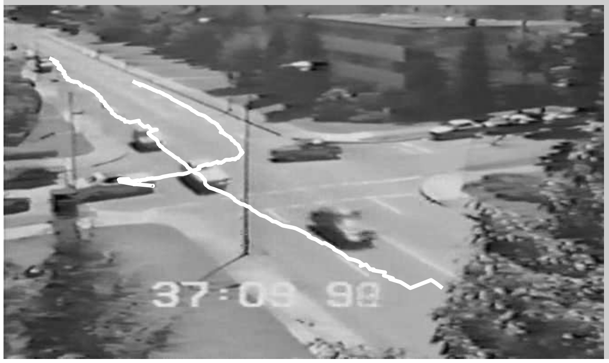

Fig. 2. An example of trajectories involved in a traffic conflict.

All K HMMs of a mixture have the same structure parameter values (number of states, simple Gaussian observation

[2] `http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html`

density), which are tuned for each cluster of observations with the algorithm. The trajectories are measured in the image space and are smoothed with an average moving filter. HMMs are clustered with a set of 560 trajectories from eight traffic sequences. The trajectories of two sequences are not used for learning since there are many detection and tracking errors, caused by camera jitter. They are however used to test the generalization ability of the method. The set is large and representative of all trajectories, but small enough to maintain reasonable computation times. The parameters n (number of iterations once the number of HMMs doesn't change anymore) and N_{min} (minimum size of the clusters) are tuned by trial and error respectively to 3 and 5.

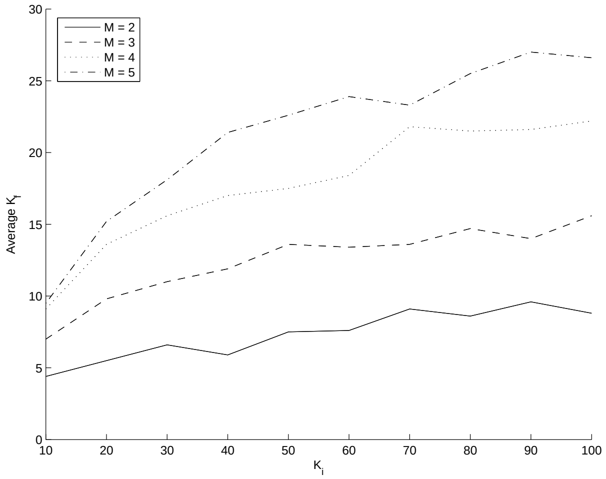

Fig. 3. The average final number K_f of HMMs, over 10 runs of the clustering algorithm 1, as a function of the initial number K_i, for different numbers of states M.

The Figure 3 shows that for different settings of the number of states M for all HMMs, the final number K_f of HMMs increases as a function of the initial number K_i. When K_i is superior to a certain value, K_f increases very slowly, and this value can be seen as the "natural" number of clusters to describe the set of trajectories. A value of $M = 3$ states seems a good tradeoff for accurate and fast learning. An observation of the learnt HMMs indicates that the 3 states (the means of the Gaussian observation density) often correspond to the approach of the intersection, the conflict zone and the leaving of the intersection. Besides, the stopping criterion of the algorithm is quickly reached, most discarding of HMMs occurring at the first iteration of the algorithm, followed by n more iterations.

An example of the result of the clustering algorithm is presented in the Figure 4. The figures display the trajectories assigned to each cluster. The main vehicle streams are represented, sometimes mixed in the same cluster. Most adapted HMMs have fewer assigned trajectories than non-adapted models. However, 66 trajectories are assigned to the largest cluster. The examination of these clusters suggests a plausible division of the trajectories space, covering different paths with differing velocities.

The detection results are presented in terms of interactions. They are computed on all the available data, the trajectories detected in the ten sequences. This is a two-class classification problem, traffic conflicts against non-conflict interactions, with two types of errors: false alarms (FA), when the system accepts a non-conflict interaction, and false rejections (FR), when the system rejects a traffic conflict. Designed as the system is, the traffic conflicts used to identify and adapt the HMMs will always be detected. It is sometimes difficult to decide upon the other detected traffic conflicts: some are clearly FA, but some could be real traffic conflicts, and are counted in their own category. When these uncertain traffic conflicts are not detected, no FR is counted.

TABLE I

DETECTION RESULTS AS A FUNCTION OF THE WEIGHTING FACTOR α, WHICH CONTROLS THE ADAPTATION OF THE HMMS TO THE TRAFFIC CONFLICT INSTANCES. "CD" INDICATES THE NUMBER OF CORRECTLY DETECTED TRAFFIC CONFLICTS, "UNCERTAIN TC" THE NUMBER OF UNCERTAIN TRAFFIC CONFLICTS, AND "FA" THE FALSE ALARMS. THE ROW "$\alpha = 0$" INDICATES THE RESULTS WITHOUT ANY ADAPTATION.

α	CD	Uncertain TC	FA
"0"	10	17	38
0.05	10	13	6
0.10	10	13	10
0.15	10	12	6
0.20	10	3	3
0.25	10	5	2
0.30	10	5	2
0.35	10	4	1
0.40	10	4	0
0.45	10	4	0
0.50	10	3	0

The tradeoff between adaptation and generalization is illustrated by studying the performance of a mixture of HMMs, with different levels of adaptation, controlled by the weighting factor α. In the Table I, the detection results are presented for the mixture of HMMs for which the clustered trajectories are displayed in Figure 4. 1501 interactions are detected. The numbers are quite high because the tracking is sometimes lost, which increases the number of trajectories measured, and therefore the number of interactions. This is also why the five training traffic conflicts instances are detected as ten interactions. The most striking is the improvement of results between the non-adapted HMMs ("$\alpha = 0$") and the smallest level of adaptation ($\alpha = 0.05$). It must be noted that most FA are caused by detection and tracking errors in one of the test traffic sequences, and not by the traffic conflict detection method. The system can generalize as there are no FA for all levels of adaption on the other test sequence without traffic conflicts.

There seems to be a good tradeoff between adaptation and generalization for $\alpha = 0.15$ or 0.2. As the adaptation is more important, the system becomes very specific to the training traffic conflicts instances, and is not able anymore to generalize to unknown data, as it is showed by the overall decrease of the number of uncertain traffic conflicts. For

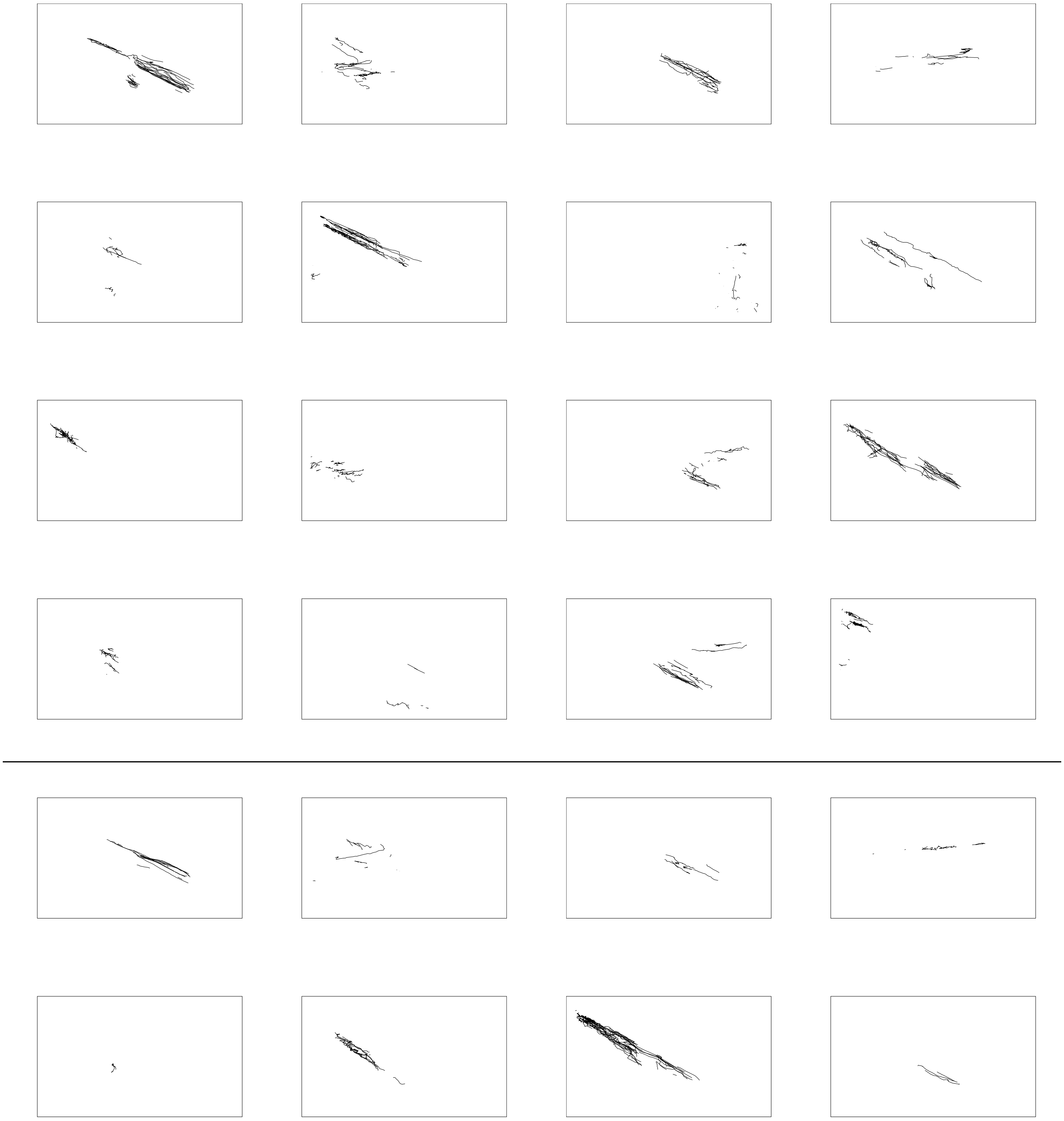

Fig. 4. Each image represents the trajectories assigned to each cluster (see Figure 1 and 2 to locate the trajectories in the intersection). The conflicting adapted clusters are displayed in the last two rows, underneath the horizontal line.

example, the traffic conflict of sequence 4 is detected for the non-adapted HMMs and for $\alpha \leq 0.15$, but not anymore when the adaptation increases. Such overfitting should be avoided.

The lack of data and ground truth make this evaluation difficult, but these results demonstrate the possibility of detecting traffic conflicts among all interactions with the presented method, and even generalize to some unseen data.

The system can be made more or less specific according to the application. Additional experimental material is available online[3].

VI. Conclusion and Future Work

In this paper a new semi-supervised approach to detect traffic conflicts is presented, based on the unsupervised general clustering of trajectories, and their adaptation with a few training traffic conflict instances. The potential of such a technique to discriminate traffic conflicts among other interactions is demonstrated on real world data. It is hoped that such methods can make significant improvements for road safety diagnosis.

Improvements to the method can be made at different levels. The use of simulated traffic data is contemplated to evaluate on a varied set of controlled situations how generic the technique is. The method can be refined by using the labeled data to guide the clustering process and by investigating more complex DBNs. K-means clustering is only guaranteed to converge to a local optimum, and the results are highly dependent on the initialization of the algorithm. Other clustering techniques such as mean shift clustering, as well as combining the clustering results can make the process more robust.

Acknowledgement

The authors would like to thank Neeraj Kanhere and Stan Birchfield from Clemson University for sharing their feature-based tracking code with us.

References

[1] G. R. Brown, "Traffic conflict for road user safety studies," *Canadian Journal of Civil Engineering*, vol. 21, pp. 1–15, 1994.

[2] T. Sayed, G. R. Brown, and F. Navin, "Simulation of Traffic Conflicts at Unsignalised Intersections with TSC-Sim," *Accident Analysis & Prevention*, vol. 26, no. 5, pp. 593–607, 1994.

[3] T. Sayed and S. Zein, "Traffic conflict standards for intersections," *Transportation Planning and Technology*, vol. 22, pp. 309–323, 1999.

[4] S. Messelodi and C. M. Modena, "A computer vision system for traffic accident risk measurement: A case study," ITC, Tech. Rep. ITC-irst T05-06-07, 2005.

[5] W. Hu, X. Xiao, D. Xie, T. Tan, and S. Maybank, "Traffic accident prediction using 3d model based vehicle tracking," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 3, pp. 677–694, May 2004.

[6] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi, "Traffic monitoring and accident detection at intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 2, pp. 108–118, June 2000.

[7] S. Atev, H. Arumugam, O. Masoud, R. Janardan, and N. P. Papanikolopoulos, "A vision-based approach to collision prediction at traffic intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 4, pp. 416– 423, Dec. 2005.

[8] S. T. Birchfield, "KLT: An Implementation of the Kanade-Lucas-Tomasi Feature Tracker," http://www.ces.clemson.edu/~stb/klt/.

[9] N. K. Kanhere, S. J. Pundlik, and S. T. Birchfield, "Vehicle segmentation and tracking from a low-angle off-axis camera," in *IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, California, June 2005.

[10] T. G. Dietterich, *Structural, Syntactic, and Statistical Pattern Recognition*, ser. Lecture Notes in Computer Science. Springer-Verlag, 2002, vol. 2396, ch. Machine Learning for Sequential Data: A Review, pp. 15–30.

[11] K. Murphy, "Dynamic bayesian networks: Representation, inference and learning," Ph.D. dissertation, UC Berkeley, Computer Science Division, July 2002.

[12] X. Zhu, "Semi-supervised learning literature survey," Computer Sciences, University of Wisconsin-Madison, Tech. Rep. 1530, 2005, http://www.cs.wisc.edu/~jerryzhu/pub/ssl_survey.pdf.

[13] S. Atev, O. Masoud, R. Janardan, and N. Papanikolopoulos, "A collision prediction system for traffic intersections," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, Aug. 2005, pp. 169–174.

[14] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan, "Semi-supervised adapted hmms for unusual event detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, June 2005.

[15] R. Xu and D. Wunsch, "Survey of clustering algorithms," *Neural Networks, IEEE Transactions on*, vol. 16, no. 3, pp. 645–678, May 2005.

[16] D. Buzan, S. Sclaroff, and G. Kollios, "Extraction and clustering of motion trajectories in video," in *Proceedings of the 17th International Conference on Pattern Recognition*, vol. 2, Aug. 2004, pp. 521–524.

[17] S. Khalid and A. Naftel, "Classifying spatiotemporal object trajectories using unsupervised learning of basis function coefficients," in *VSSN '05: Proceedings of the third ACM international workshop on Video surveillance & sensor networks*. New York, NY, USA: ACM Press, 2005, pp. 45–52.

[18] S. Kamijo, K. Ikeuchi, and M. Sakauchi, "Event recognitions from traffic images based on spatio-temporal markov random field model," in 8^{th} *World Congress on ITS*, Sydney, Sept. 2001, CD-ROM.

[19] T. Huang, D. Koller, J. Malik, G. H. Ogasawara, B. Rao, S. J. Russell, and J. Weber, "Automatic symbolic traffic scene analysis using belief networks," in *National Conference on Artificial Intelligence*, 1994, pp. 966–972. [Online]. Available: citeseer.csail.mit.edu/huang94automatic.html

[20] J. Alon, S. Sclaroff, G. Kollios, and V. Pavlovic, "Discovering clusters in motion time-series data," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2003.

[21] I. Cadez, S. Gaffney, and P. Smyth, "A general probabilistic framework for clustering individuals," University of California, Irvine, Tech. Rep. UCI-ICS 00-09, Mar. 2000.

[22] T. Oates, L. Firoiu, and P. Cohen, "Clustering time series with hidden markov models and dynamic time warping," in *Proceedings of the IJCAI-99 Workshop on Neural, Symbolic and Reinforcement Learning Methods for Sequence Learning*, 1999, pp. 17–21.

[23] C. Li and G. Biswas, "A bayesian approach to temporal data clustering using hidden markov models," in *International Conference on Machine Learning*, Stanford, 2000, pp. 543–550.

[24] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2004, pp. 819–826.

[25] N. Johnson and D. Hogg, "Learning the distribution of object trajectories for event recognition," *Image and Vision Computing*, vol. 14, no. 8, pp. 609–615, Aug. 1996, http://www.sciencedirect.com/science/article/B6V09-3VVCMHP-1B/2/4f0698a2bcc8a1e2d24a805a790b6533.

[26] M. Brand and V. Kettnaker, "Discovery and segmentation of activities in video," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 22, no. 8, pp. 844–851, Aug. 2000.

[27] C. Stauffer and E. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[28] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[29] J. Bilmes, "A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," University of Berkeley, Tech. Rep. ICSI-TR-97-021, 1997. [Online]. Available: citeseer.ist.psu.edu/bilmes98gentle.html

[3] `http://www.confins.net/saunier/`