Anomalous Video Event Detection Using Spatiotemporal Context

Fan Jiang[a,*], Junsong Yuan[c], Sotirios A. Tsaftaris[a,b], Aggelos K. Katsaggelos[a]

[a] *Department of Electrical Engineering and Computer Science, Northwestern University*
2145 Sheridan Rd, Evanston, IL 60208, USA
[b] *Department of Radiology, Northwestern University*
737 N Michigan Ave, Chicago, IL 60611, USA
[c] *School of Electrical and Electronics Engineering, Nanyang Technological University*
50 Nanyang Ave, Singapore 639798

Abstract

Compared to other anomalous video event detection approaches that analyze object trajectories only, we propose a context-aware method to detect anomalies. By tracking all moving objects in the video, three different levels of spatiotemporal contexts are considered, i.e., point anomaly of a video object, sequential anomaly of an object trajectory, and co-occurrence anomaly of multiple video objects. A hierarchical data mining approach is proposed. At each level, frequency based analysis is performed to automatically discover regular rules of normal events. Events deviating from these rules are identified as anomalies. The proposed method is computationally efficient and can infer complex rules. Experiments on real traffic video validate that the detected video anomalies are hazardous or illegal according to traffic

*Corresponding author (phone: 224-392-2622, fax: 847-491-4455)
Email addresses: fanjiang2008@u.northwestern.edu (Fan Jiang), jsyuan@ntu.edu.sg (Junsong Yuan), s-tsaftaris@northwestern.edu (Sotirios A. Tsaftaris), aggk@eecs.northwestern.edu (Aggelos K. Katsaggelos)

regulations.

Keywords: Video surveillance, anomaly detection, data mining, clustering, context

---

1. Introduction

Discovery of suspicious or anomalous events from video streams is an interesting yet difficult problem for many video surveillance applications. By automatically finding suspicious events, it significantly reduces the cost to label and annotate the video streams of hundreds of thousands of hours. In many scenarios, the video camera is fixed and the site being monitored is mainly static. By modeling the statistics of the background and the appearance and dynamics of the foreground (objects such as a person, car, airplane), various features of the objects, such as location and motion at different times, can be extracted from the video data. These object features are useful in characterizing video events. For instance, a video event can be defined by the motion trajectory of any single object [1–7].

Anomalous video event detection is a challenging problem in that it is difficult to define anomaly in an explicit manner. It is possible that we may need to identify an anomalous event when it appears, despite the fact that it had never occurred before. The more practical approach is to detect normal events first (as they follow some regular rules) and treat the rest as anomalies. In many cases, however, *a priori* knowledge of regular rules is lacking and no training data for normal video events are available. Therefore, there is a need for an unsupervised approach to automatically mine these rules directly from unlabeled data.

Clustering-based approaches have been recently investigated to address this problem [1–3, 5, 7, 8]. This approach is based on the fact that normal events appear frequently and dominate the data, while anomalies are different from the commonality and appear rarely. For instance, a running person can indicate an anomalous event if most people in the crowd are walking; similarly, a car moving against the direction of most other moving vehicles can indicate an anomalous event too. Therefore, unsupervised clustering can be performed on all video events. Those events clustered into dominant (e.g., large) groups can be identified as normal, representing the regular rules. Those that cannot be explained by the regular rules (e.g., outliers distant from all cluster centers) are defined as anomaly.

Despite the success of clustering-based approaches for anomaly detection, there exist several limitations. Most clustering approaches consider a video event as the motion trajectory of one single object. However, this definition ignores important spatial and temporal contextual information. On one hand, video anomaly may not correspond to the whole trajectory, only to a part of it. On the other hand, anomaly can arise due to the inappropriate interactions among multiple objects (i.e., multiple trajectories), even though their individual behaviors are normal. Thus, anomaly detection based on trajectory clustering can cause miss detections.

Instead of relying solely on trajectories, we define video events at different semantic levels considering both spatial and temporal context. At each level, frequency based analysis is performed. Events appearing with high frequency are automatically discovered and declared to be an explicitly description of the regular rules. The events deviating from these rules are detected as

anomalies. We test the proposed approach on real traffic videos, where vehicles have been detected and tracked. The task is to discover anomalous events from a collection of movement trajectories of vehicles. The results show that our approach can automatically infer regular rules of traffic motion of the specific scene (corresponding to the real traffic rules) and detect anomalous events at three levels: motion of one vehicle at any specific time, motion of one vehicle within a time range, and co-occurrence of multiple vehicles. Most of the detected video anomalies are proved to be hazardous or illegal, according to vehicular traffic regulations.

The rest of this paper is organized as follows. Section 2 provides an overview of the recent literature. Sections 3, 4 and 5 describe our approach in discovering anomalous video events at different semantic levels. Experimental results are presented in Section 6, and we conclude the paper in Section 7.

2. Related Work

Many approaches of video event analysis are based on the object trajectories extracted from video. Due to the lack of *a priori* knowledge of normal events, unsupervised clustering is performed on all trajectories and dominant trajectory clusters are identified and modeled as normal event patterns (i.e., regular rules). Then anomalous events can be detected from those trajectories not fitting the normal models. Specifically, there are many different representations of an object trajectory given, for example, by a sequence of muti-dimentional features [1], the curvature feature of the trajectory [9], a linear dynamical system [10], dynamic Bayesian networks [7, 11–13], the

motion histogram [14], and the multivariate probability density function of spatiotemporal variables [15]. Considering trajectory clustering, the classical k-means algorithm is applicable if each trajectory is resampled to a fixed length [12] and the number of clusters is estimated using the approach in [4]. The spectral clustering algorithm is another popular choice [2, 5, 13] because the number of clusters can be well determined by performing eigenvalue decomposition of the affinity matrix of all trajectories. Alternatively, a sequential grouping method is used in [3], where each trajectory is sequentially taken from the database and either matched to an existing group or used to initialize a new group. Other algorithms used for trajectory clustering include hierarchical clustering [7], mixture model [11, 14], mean shift [6], SVM [8], and kernel density estimation [15].

In some noisy video data, however, the object trajectory cannot be estimated accurately. Approaches are proposed to represent video events based on features at the pixel-level or at local spatio-temporal patches. Based on these representations, normal events are discovered by capturing and modeling the dominant motion of objects involved in the video stream. For example, Zhong et al. [16] used spatiotemporal gradients of all pixels to represent video motion and detected video anomalies by spectral clustering on this gradient field. Boiman and Irani [17] used spatial-temporal patches for event modeling. They considered video normality as being composed from large chunks of spatial-temporal patches. Hamid et al. [18] introduced a representation of activities as bags of event n-grams, where they analyzed the global structural information of activities using local event statistics. They detected anomalous events based on discovering regular sub-classes of normal events.

Wang et al. [19] represented video events as distributions over low-level visual features on a pixel basis and used hierarchical Bayesian models for event clustering. In our previous work [20] we proposed a representation of crowd motion in video using moving blobs and the spatial relationship among blobs, based on which anomalous motion or interaction of pedestrians is detected.

Despite the many different representations of video event, spatial and temporal contextual information is not typically used, which limits the power of video anomaly detection. By considering spatial context, an anomalous video event may include not only a single agent (e.g., a moving object or an image patch), but also multiple spatially related agents and their interactions. By considering temporal context, an anomalous video event may include behaviors at multiple times, i.e., having an arbitrary length of time. Many of the existing works fail to provide any modeling of such contextual information. One of the attempts to model spatiotemporal context for video event analysis is the work of Oliver et al. [21], where a coupled HMM is used to model the interaction between two object trajectories. Galata et al. [22] used variable length Markov models to represent temporal dependency of atomic behavior components. Yao et al. [23] represented the trajectories of multiple objects by a 3D graph which is augmented by a number of spatiotemporal relations (links) between moving and static objects in the scene. The works in [24, 25] learned co-occurrence statistics using a Markov random field model for normal events across space-time. Wang et al. [19] used hierarchical Bayesian models to connect three elements in visual surveillance: low-level visual features, simple atomic activities, and interactions. Based on this modeling a summary of typical atomic activities and interactions occur-

ring in the scene was provided and video anomalies were detected at different levels.

Similarly to [19], we propose a hierarchical representation of video events. Instead of intuitively considering two levels of video events (i.e., atomic activities and interactions) as in [19], we define three levels of events based on different spatial and temporal context. In order to detect anomalous video events at different levels, we perform frequency based analysis which is a bottom-up method, differing from the top-down method (a generative model) used in [19]. In addition, by utilizing the object tracking information, our approach can detect anomalous events associated with specific object(s) at specific time(s), instead of image pixels or patches as in [19].

3. Point anomaly detection

In a video scenario with moving objects (e.g., vehicle, humans), the most easily observed activity is the instant behavior of any single object i at a specific time t, which we categorize as an *atomic event* $e_a(i,t)$. Typically, an atomic event describes the location, moving direction, and velocity of the object at each video frame. It is the basic unit for describing more complicated events and interactions.

For any specific video scenario, the instant motion of a single object usually follows certain rules. For example, road traffic in a certain lane has to move in a determined direction, and the traffic waiting for red lights must stop at a certain location. As most atomic events follow some regular motion rules, we can detect normal and anomalous ones based on their frequency of appearance. A simple yet effective way of achieving the above is to represent

each atomic event as a discrete feature vector, compute the histogram of all feature vectors, and use a threshold to identify bins of lower probability. Those atomic events with low frequency are declared point anomalies (following a similar definition given in [26]), because these anomalies consider no contextual information. After this step we can readily detect some obvious anomalies from the video, and exclude them from subsequent analysis.

4. Sequential anomaly detection

A video anomaly may not only consist of instantaneous behavior, but may also be characterized by the ordering or transition of instantaneous behaviors. For example, in a traffic scenario, two atomic events, such as entering an intersection from a straight-only lane and making a left turn within the intersection, can be normal. But their combination is anomalous (illegal). In order to exploit this temporal context, we define a *sequential event* $e_s(i)$ as a sequence of atomic events associated with the trajectory of an object i. Note that the same atomic event appearing continuously is regarded as only one item in the sequence. For example, $e_s(i)$ is represented by the sequence $\big(e_a(i,1), e_a(i,2), e_a(i,4), \cdots\big)$, if $e_a(i,3) = e_a(i,2)$.

Similarly to point anomaly detection, an anomalous sequential event can be identified by finding sequences that appear rarely, e.g., turning left from a straight-only lane must be rare compared to other sequential events. However, a sequential anomaly may last an arbitrary length of time (possibly only a part of the complete sequence). Techniques are needed to deal specifically with the variation of time length. Another difficulty is the effect of the noise when counting the frequency of similar sequences. For example, if each

atomic event is denoted by a number 1, 2, 3, $\cdots$, the sequence of (1, 6, 2, 3, 4, 5) might be counted as an appearance of the sequence (1, 2, 3, 4, 5), because they are almost the same except for the insertion of atomic event 6. Thus, small variations must be allowed for.

To accommodate this design constraint, we adopt the technique of frequent subsequence mining [27]. Given two sequential events defined in our work, $e_s(i) = \big(e_a(i,1), e_a(i,2), \cdots\big)$ and $e_s(j) = \big(e_a(j,1), e_a(j,2), \cdots\big)$, the subsequence relationship between them is defined as follows: $e_s(i)$ is a subsequence of $e_s(j)$, if and only if a monotonically-increasing index mapping I for each element in $e_s(i)$ can be established, such that each element $e_a(i,k)$ is a subset of $e_a(j, I(k))$. That is, a sequence is a subsequence of another, if it can be matched with arbitrary long gaps but preserving the order, such that the matched elements satisfy a subset relationship. This accommodates the design constraint of allowing for small variations that can be due to the presence of noise.

In practice, we apply the CloSpan algorithm by Yan et al. [27] on all sequential events collected from the video. It automatically discovers frequent subsequences (instead of the complete sequences) with their frequency above a given threshold. In addition, this algorithm ensures the discovered subsequences contain no super-sequence with the same support (i.e., occurrence frequency). Therefore, the resulting subsequences include all typical repetitive patterns of the collected sequential events, and are thus regarded as patterns of normal sequential events.

Based on these normal patterns, we can classify every sequential event and detect anomalous parts. To compare the similarity between two sequences,

we use the edit distance [28]. In our case, the edit distance between two sequences is given by the minimum number of operations (insertion, deletion, or substitution of an atomic event) needed to transform one sequence into the other. Therefore, any sequential event is classified to the normal pattern with the minimum edit distance. Consequently, those atomic events within a sequence, which need to be deleted to match the normal patterns, are identified as anomalies. Video anomaly detected at this level considers the temporal context of an object trajectory and is referred to as sequential anomaly. Note that the sequential anomaly is not necessarily a complete sequence, but can be any part of a sequence with arbitrary time length.

5. Co-occurrence anomaly detection

The highest level of anomaly arises from the co-occurrence of multiple objects. For example, in a traffic scenario, turning left and going straight within the intersection are both normal events when considered independently; however, making a left turn in front of incoming traffic is illegal and thus anomalous. This co-occurrence anomaly usually happens in the area with multiple objects and intensive interactions, e.g., within a road intersection. In order to detect this kind of anomaly, we first define a *co-occurrence event* $e_c^A(t)$ as an instant event at time t for a specific area A of a video frame. As every object appearing in this area at any time instance has a label of atomic event pattern and sequential event pattern (anomalies are excluded), a co-occurrence event can be represented as an itemset, with each item corresponding to a label. Possible definitions include an itemset of atomic event labels, i.e., $\{e_a(i,t)|$ all i appearing in area A at $t\}$, and an

itemset of sequential event labels, i.e., $\{e_s(i)|$ all i appearing in area A at $t\}$.

Similarly to point and sequential anomaly detection, an anomalous co-occurrence event is characterized by its rareness of appearance compared to other co-occurrence events. In order to find normal patterns of co-occurrences and detect anomalies, we apply the frequent itemset mining algorithm [29, 30] on all co-occurrence events collected from the video, treating each co-occurrence event as a transaction. This algorithm discovers frequent subsets of co-occurrences (frequency above a given threshold) and also ensures the discovered subsets contain no superset with the same support. The resulting subsets include all typical repetitive patterns of the collected co-occurrence events, and are thus regarded as patterns of normal co-occurrence events.

Based on these normal patterns, we can classify every co-occurrence event and detect the anomalous parts. A simple approach is to classify each co-occurrence event to the normal pattern with maximal overlapping items. Nevertheless, it neglects the temporal constraint of co-occurrence events through video stream. For example, in a traffic video of a road intersection, as a result of the traffic light signaling, only a few combinations of driving behaviors are allowed at one time. New behaviors appear only at the time when traffic lights change. In other words, normal co-occurrence events here correspond to a few traffic states and there exist specific ways of transitioning among them. Therefore, in order to classify co-occurrences at every time to a normal pattern (state), we need to consider this temporal constraint.

Based on the above observation, the co-occurrences at all times can be considered as an observation sequence Y generated from a hidden Markov

model (HMM), where $Y = \left(e_c^A(1), e_c^A(2), \cdots\right)$. The hidden states correspond to normal co-occurrence events discovered previously by frequent itemset mining. Therefore, in order to classify every co-occurrence event to one of the normal co-occurrence patterns, we need to determine the most likely sequence of hidden states that led to the observations in Y. Actually, co-occurrence classification becomes an HMM decoding problem.

First we need to determine the parameters of the HMM. We denote a_{ij} as the transition probability from state i to state j, and $b_j(t)$ as the probability of state j emitting a co-occurrence $e_c^A(t)$. Note that $b_j(t)$ has a discrete probability distribution with infinite number of observed values, because $e_c^A(t)$ may consist of an arbitrary number of items. Due to this complexity of $b_j(t)$, the conventional forward-backward algorithm may not be applicable. In a forward algorithm, in order to calculate the forward probability $\alpha_j(t)$ (the probability that the HMM in state j at step t having generated the first t observations), $b_j(t)$ needs to be specified.

To address this issue, we propose a special modeling of the distribution of $b_j(t)$, based on which the Viterbi algorithm can be used iteratively to solve the HMM decoding problem. This solution is based on the observation that the Viterbi algorithm does not rely on the exact value of $\alpha_j(t)$ but on the comparison among all $\alpha_j(t)$'s for a fixed t, because it only needs to choose one path at each step. Specifically, at each time t, the Viterbi algorithm chooses for each state j a transition path, i.e., state i at step $t-1$, as

$$i = \arg\max_i \Big(\alpha_i(t-1)a_{ij}\Big). \tag{1}$$

If $\alpha_i(t-1)$ is comparable for different i, this path can be successfully chosen and the HMM decoding problem can be finally solved.

As we all known, $\alpha_j(t)$ is calculated as

$$\alpha_j(1) = b_j(1), \tag{2}$$

$$\alpha_j(2) = b_j(2) \sum_i \alpha_j(1) a_{ij}, \tag{3}$$

$$\alpha_j(3) = b_j(3) \sum_i \alpha_j(2) a_{ij}, \tag{4}$$

$$\cdots$$

$$\alpha_j(t) = b_j(t) \sum_i \alpha_j(t-1) a_{ij}. \tag{5}$$

Therefore, instead of specifying the exact value of $b_j(t)$, we may just model the relationship among $b_j(t)$'s for different j. Specifically, we assume that for any co-occurrence $e_c^A(t)$, the probabilities of it emitted from state i or j satisfy the relationship

$$\frac{b_i(t)}{b_j(t)} = \frac{m_i(t)}{m_j(t)}, \tag{6}$$

where $m_j(t)$ is the number of items in $e_c^A(t)$ that belong to pattern j. In other words, the emission probability is proportional to the number of items shared by the emission itemset and the state itemset. For example, $e_c^A(t) = \{1, 1, 2, 2, 2, 3, 4, 5, 5\}$ ($1, 2, 3, \cdots$ are different item labels), state $i = \{1, 2, 3\}$, and state $j = \{3, 4, 5\}$. We have $b_i(t)/b_j(t) = 6/4$, because the items 1, 2 and 3 appear in $e_c^A(t)$ 6 times in total and the items 3, 4 and 5 appear in $e_c^A(t)$ 4 times in total. Alternatively, we can express $b_j(t)$ for any state j as

$$b_j(t) = c(t) \cdot m_j(t), \tag{7}$$

where $c(t)$ is the same constant for every state. Substituting (7) into (2)-(5),

we have

$$\alpha_j(1) = c(1) \cdot m_j(1), \tag{8}$$

$$\alpha_j(2) = c(1)c(2) \cdot m_j(2) \sum_i \big(m_i(1)a_{ij}\big), \tag{9}$$

$$\alpha_j(3) = c(1)c(2)c(3) \cdot m_j(3) \sum_i \Big(m_i(2)a_{ij} \sum_k \big(m_k(1)a_{ki}\big)\Big), \tag{10}$$

$$\cdots$$

$$\alpha_j(t) = \Big(\prod_{i=1}^{t} c(i)\Big) \cdot f\Big(\{m_j(i)\}_{i=1}^{t}, \{a_{ij}\}\Big) \tag{11}$$

Note that the $c(t)$ term is constant and $f(\cdot)$ is only related to $\{m_j(i)\}$ and $\{a_{ij}\}$. Therefore, $\alpha_j(t)$'s can be compared for different j at any time t without knowing the exact value of $b_j(t)$. Based on this modeling, the Viterbi path can be determined by (1).

Actually, we use an iterative approach to determine $\{a_{ij}\}$ and to decode states, as shown in Algorithm 1.

Algorithm 1

1: Set initial $\{a_{ij}\}$ to uniform distribution;

2: Decode states by Viterbi algorithm based on (1) and (11);

3: Estimate $\{a_{ij}\}$ by taking the ratio between the number of transitions from state i to state j and the total number of any transitions from state i. Go to step 2 to recalculate $\{a_{ij}\}$ until the error of $\{a_{ij}\}$ is small enough (convergence is reached).

Once each co-occurrence event is associated with one of the states (normal co-occurrence patterns), anomalies can be detected by figuring out those items that are different from its corresponding normal co-occurrence pattern.

Specifically, if the co-occurrence event $e_c^A(t)$ is classified to the pattern j, all the items in $e_c^A(t)$ that are not included in the itemset of pattern j are identified as anomalies (co-occurrence anomalies).

6. Experimental results

In this section, we present an experimental study in order to evaluate our approach. As a general mining approach, the proposed 3-level anomaly detection is applicable to many different scenarios. These scenarios should include a large amount of object motion. The normal motions, both the motion of a single object and the motion of multiple objects, follow some intrinsic but unknown rules. Most motions follow this rule while few anomalies do not. Our task is to automatically mine these rules of normal motion from all the data (no prior knowledge, training data, etc.) and to detect any anomalous motions breaking rules. One good example is the traffic motion at an intersection guided by traffic lights.

6.1. Data

As a case study, we have selected a surveillance video monitoring traffic for a long time at a road intersection. This video is taken from a large database of aerial traffic videos from the Next Generation Simulation (NGSIM) project (http://ngsim.camsys.com/). One example frame is shown in Fig. 1. This is an one-hour-long video monitoring a 4-way intersection in Los Angeles, California. Each of the roads is a two-way road with multiple lanes (some with right turn or left turn lanes). All moving traffic in this area is controlled by traffic lights within the intersection. Thus, the underlying rule of normal motion is the legal motion directed by the traffic lights. Detailed trajectory

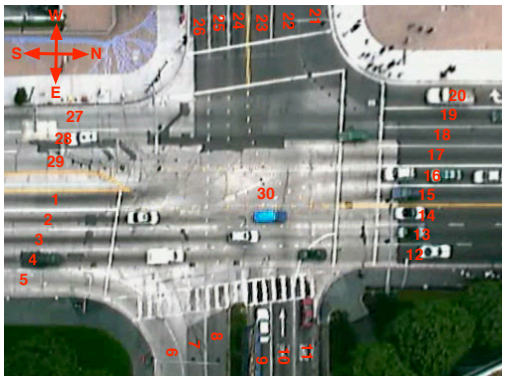


Figure 1: Example frame of video monitoring traffic at road intersection (all lanes are numbered from 1 to 29 and the intersection area is numbered as 30).

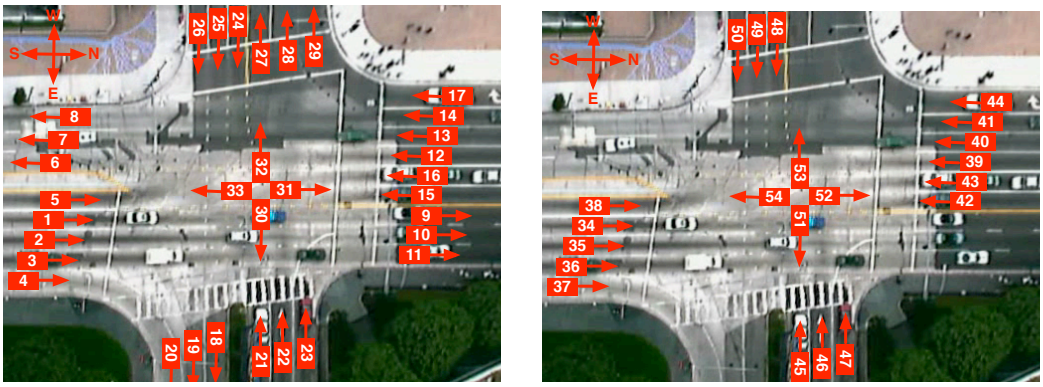
information for every vehicle is available. However, with the information of traffic signaling unknown, our goal is to discover traffic rules followed by most vehicles in this area and to detect anomalies at different levels.

6.2. Point anomaly detection

For the point anomaly detection, we represent each atomic event by three discrete features, i.e., the position of the vehicle, the driving direction, and the velocity. Every feature is quantized to discrete values. In our experiment, the vehicle position is represented by the specific lane or intersection it occupies (although other positional features are applicable, the lane information for each vehicle at every specific time is available directly from this database). As shown in Fig. 1, all lanes are numbered from 1 to 29. The intersection area is numbered as 30. The driving direction has 4 possible values (north, south, west, east). The velocity is discretized to either moving ($v > 0$) or stopping ($v \approx 0$).

A 3-D histogram for all the atomic events throughout the video is es-

tablished. By applying a threshold (10% of the average bin height in our experiments), we detect 54 frequent (normal) behaviors, as shown and numbered in Fig. 2. Specifically, Fig. 2(a) shows moving normal events ($v > 0$),



(a) (b)

Figure 2: Frequent atomic events: (a) moving normal events ($v > 0$) with the red squares referring to vehicle positions and the arrows indicating driving directions; (b) stopping normal events ($v \approx 0$) with the arrows showing the facing directions of the vehicles.

with the red squares referring to vehicle positions and the arrows indicating driving directions. Figure 2(b) shows stopping normal events ($v \approx 0$), with the arrows showing the facing directions of the vehicles. It is observed that these 54 normal atomic events include all the legal driving directions allowed in every lane. Consequently, the atomic events that do not fall into any of these normal ones are illegal driving situations and are thus detected as anomalies. Two examples are shown in Fig. 3, where tracked vehicles are indicated by green numbers. Figure 3(a) shows a vehicle (indicated by a red ellipse) moving eastwards, since it moves sharply from lane 4 to lane 5 (see Fig. 1). This anomalous movement is due to the fact that the vehicle is intended to make a right turn at the intersection, but did not decide to change

(a) (b)

Figure 3: Example results of point anomaly (green label indicates ID of each vehicle) : (a) shows a vehicle (indicated by a red ellipse) moving eastwards, since it moves sharply from lane 4 to lane 5 (see Fig. 1); (b) shows one vehicle stopping in lane 12 (see Fig. 1) right after leaving the intersection.

lanes until the very last moment. Figure 3(b) shows one vehicle stopping in lane 12 (see Fig. 1) right after leaving the intersection. Both of them are disruptive behaviors for normal traffic, as they may block subsequent traffic.

6.3. Sequential anomaly detection

Based on the 54 normal instant behaviors we identified, all the point anomalies can be excluded from the database. Then, sequential anomaly can be detected from the remaining data. For each vehicle, from the time it appears in the video to the time it disappears, all of its instant events are concatenated into a sequence. The sequence of events encodes information on the temporal relationship of atomic events. For example, most of the vehicles starting from atomic event 1 in Fig. 2(a) are going to proceed with atomic event 31, followed by event 9, because vehicles are going straight when starting with atomic event 1. The sequence (1,31,9) should appear

frequently (possibly as subsequence) within all the sequences collected from the video. On the other hand, the sequence (1,31,10) appears rarely if any, because few vehicles change lane within the intersection when going straight (this is actually illegal).

Applying frequent subsequence mining on all sequences (the threshold is set to 1% of the total sequence number), we detect 44 frequent (normal) sequential patterns, with some of them shown in Fig. 4. It is observed that

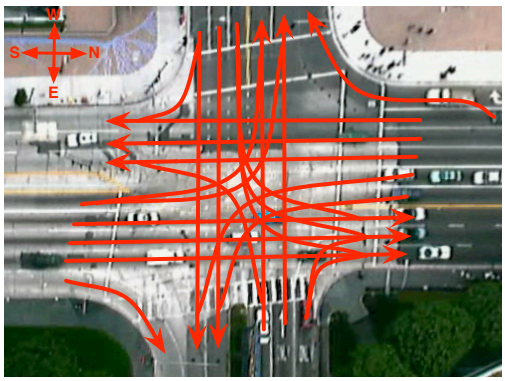


Figure 4: Frequent sequential events indicated by red paths.

all possible traveling routes permitted in this area are included. After that, we classify every sequential behavior to one of the normal patterns with the minimal edit distance. The anomalous part of the sequential behavior is detected as those atomic behaviors which need to be deleted so as to match the normal pattern. Two examples are shown in Fig. 5, with the anomalous part shown by a red dashed line. Figure 5(a) shows a vehicle changing lane within the intersection. Figure 5(b) shows a vehicle making a left turn from a no-turn lane. Both of them are illegal behaviors.

(a) (b)

Figure 5: Example results of sequential anomaly (the anomalous part is shown in red dashed line) : (a) shows a vehicle changing lane within the intersection; (b) shows a vehicle making a left turn from a no-turn lane. Both of them are illegal behaviors.

6.4. Co-occurrence anomaly detection

Finally, we detect co-occurrence anomaly. We constrain our co-occurrence analysis to the region within the intersection (i.e., region 30 in Fig. 1), because this region most of the time has multiple vehicles and intensive interaction. In this experiment, a co-occurrence event is defined as an itemset of intersection-passing types of all vehicles within the intersection at this moment. Intersection-passing types are clusters of normal sequential patterns with adjacent starting and ending atomic behaviors. For example, in Fig. 2, the sequential patterns (1, 31, 9), (2, 31, 10), (3, 31, 11) fall in the same intersection-passing type south-to-north, and the sequential patterns (15, 33, 30, 18), (16, 33, 30, 19) fall in the same intersection-passing type north-to-east. Clustering all normal sequential patterns results in 12 intersection-passing types, as shown in Fig. 6.

By applying frequent itemset mining on all co-occurrences (the threshold

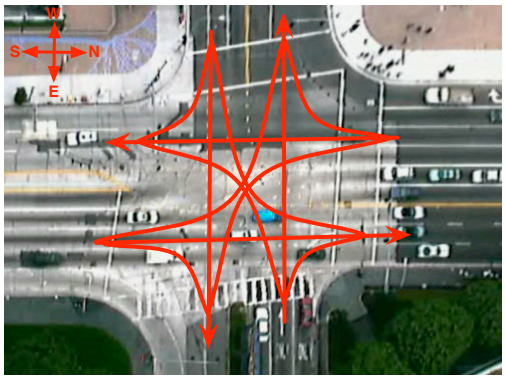


Figure 6: Intersection-passing types indicated by red routes.

is set to 1% of the total co-occurrence number), we detect a few frequent co-occurrence events. It is observed that several frequent co-occurrences have large correlation because they share very similar itemset. Thus we further cluster them into groups using spectral clustering, where pointwise mutual information [31] is used as the similarity measure. Specifically, for any two frequent co-occurrences (subsets) i and j, the similarity $s(i, j)$ is defined as

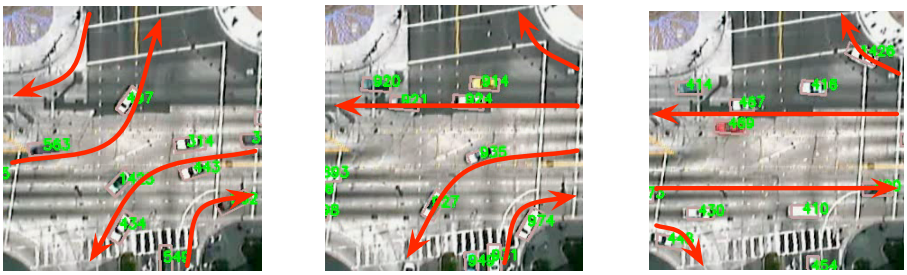
$$s(i, j) = \log \frac{p(i, j)}{p(i)p(j)}, \tag{12}$$

and is further estimated by

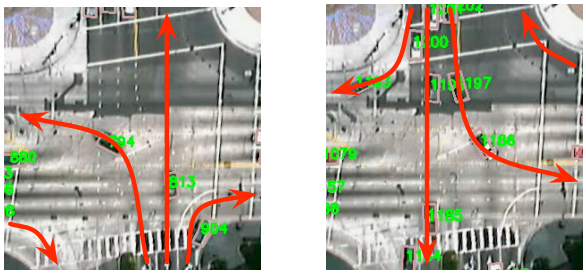
$$s(i, j) = \log \frac{(\#\text{ of } i, j \text{ appearing in the same co-occurrence})}{(\#\text{ of } i\text{'s appearances}) \cdot (\#\text{ of } j\text{'s appearances})}. \tag{13}$$

The number of clusters is determined by the eigengap heuristic, i.e., to choose the number k such that all eigenvalues $1, \cdots, k$ are very small, but $k + 1$ is relatively large. Finally, we end up with 5 groups, actually corresponding to the 5 states generated from the traffic light signals. Figure 7 depicts the driving directions allowed for each state.

Subsequently, we label the state for every co-occurrence event all through the video by the proposed iteration approach and detect all co-occurrence



(a) State 1 (b) State 2 (c) State 3



(d) State 4 (e) State 5

Figure 7: 5 normal co-occurrence patterns (states): red routes indicate the driving directions allowed for each state

anomalies. Figure 8 shows the convergence process of our iterative approach in Algorithm 1. $P(Y)$, the probability of the whole sequence Y generated from the HMM keeps increasing, while the error of transition probability keeps decreasing, until they both converge. Figure 9 shows two examples of detected co-occurrence anomalies, with the anomalous part shown by a red dashed line. Figure 9(a) shows a vehicle turning right while there is left-turning traffic going to the same lane. Figure 9(b) shows a vehicle turning left in front of incoming traffic. Both of them are illegal behaviors.

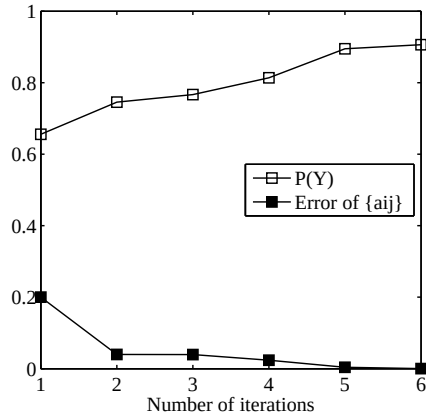


Figure 8: Convergence of iteration

6.5. Evaluation and comparison

For the three types of anomaly detection, determining the threshold is an important issue. To further test the robustness of our approach, we vary the threshold and plot ROC curves. The ground truth is acquired by manually labeling all the events. From Fig. 10 we observe that our detection performs well when the threshold is properly set, with a typical detection rate above 90% for point anomaly, above 80% for sequential anomaly, and above 70% for co-occurrence anomaly. Note that co-occurrence anomaly detection is comparably more sensitive to threshold. Actually, it performs well as long as the 5 traffic states (shown in Fig. 7) are discovered correctly, i.e., a proper threshold is selected for frequent itemset mining. Otherwise, the HMM decoding based on the incorrect states would make the results of co-occurrence labeling even worse.

Finally, we compare the proposed approach to some existing approaches applicable to the same task as ours. For sequential anomaly detection, one option is to cluster all trajectories based on the (x, y) coordinates at ev-

(a) (b)

Figure 9: Example results of co-occurrence anomaly (the anomalous part is shown in red dashed line) : (a) shows two vehicles turning into the same lane; (b) shows a vehicle turning left in front of incoming traffic. Both of them are illegal behaviors.

ery specific time and detect outliers as anomalies. Specifically, we perform spectral clustering on all vehicle trajectories in the video using a distance measure based on dynamic Bayesian network (DBN) [13]. This approach ends up with 12 clusters which are the same as the intersection-passing types shown in Fig. 6 (this is because trajectories sharing similar starting/ending positions are likely to be generated from the same DBN). Obviously, this approach fails to detect any outliers (anomalies) based on the clustering results. For example, the anomalous trajectories shown in Fig. 5(a)(b) can not be detected, because they are classified as intersection-passing types north-to-south and east-to-south, respectively. Actually, this failure is due to an improper generative model (DBN) used in this scenario, as DBN here encodes the main direction of motion but neglects the deviation of trajectories. In contrast, the sequential anomaly detection approach proposed in this paper is bottom-up and data-driven, not relying on specific model selection.

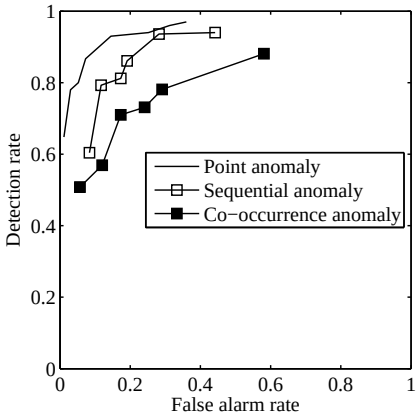
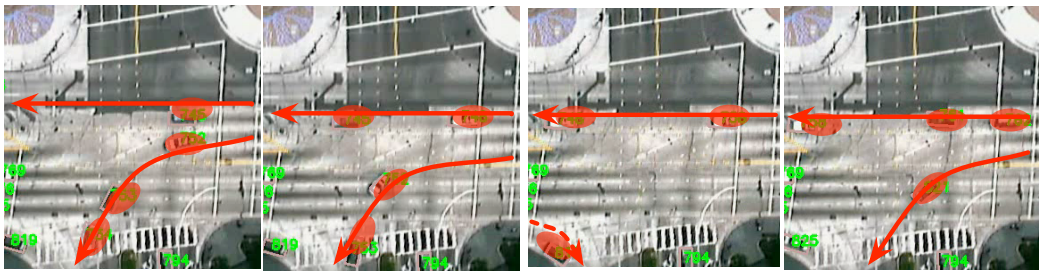


Figure 10: ROC curves

For co-occurrence anomaly detection, we cannot identify any existing method that can be used to accomplish the same task. However, in the part of co-occurrence event classification, our HMM decoding approach can be compared with a nearest neighbor classification approach. After frequent itemset mining, we have available the normal patterns of co-occurrence events, shown in Fig. 7. Nearest neighbor classification classifies each co-occurrence event to the normal pattern with maximal overlapping items, without considering the temporal consistency of patterns/states. By optimizing the threshold, this approach achieves around 60% detection rate and 30% false alarm rate, which are worse than the results presented for our approach. Figures 11(a)-(d) show the intersection area of 4 consecutive video frames (t from 930 to 933). Vehicles within the intersection are indicated by red ellipsis and their moving routes are shown in red arrows. The classification results of the 4 co-occurrence events by the two approaches, i.e., nearest neighbor classification and the HMM decoding approach, are both shown in Tab. 1. It is observed that the HMM decoding approach correctly classifies all 4 time consecutive



(a) t = 930 (b) t = 931 (c) t = 932 (d) t = 933

Figure 11: Intersection area of 4 consecutive video frames (t from 930 to 933): vehicles within the intersection are indicated by red ellipsis and their moving routes are shown in red arrows.

Table 1: Comparison results of nearest neighbor classification and the HMM decoding approach for the example shown in Fig. 11 (states are shown in Fig. 7).

t	930	931	932	933
Ground truth	State 2	State 2	State 2	State 2
HMM decoding	State 2	State 2	State 2	State 2
Nearest neighbor classification	State 2	State 2	State 3	State 2

co-occurrences to state 2 (see Fig. 7(b)), while nearest neighbor classification mistakenly classifies the co-occurrence at time 932 to state 3 (see Fig. 7(c)). As can be seen in Fig. 11(c), the co-occurrence at time 932 has no vehicle to match to the left-turning route. Instead, there is an anomaly of right-turning vehicle (shown by a red dashed line) that matches state 3 (see Fig. 7(c)). Therefore, nearest neighbor classification gives the incorrect result for this co-occurrence. In contrast, the HMM decoding approach is able to avoid this error by considering the temporal consistency of states.

7. Conclusion

With no prior knowledge about anomalous behaviors in a specific video scenario, it is necessary to follow an unsupervised approach in automatically detecting video anomalies from the data. Our approach is data-driven and applicable to many complex video scenes. The major contribution of this approach is analyzing video events at different levels considering both spatial and temporal context. In detail, considering spatial context, we analyze events of a single object and events of multiple spatially related objects. Considering temporal context, we analyze both short time (instant) events and long time events (including several short time events and their transitions). Accordingly, anomalous video events can be detected at different levels. In fact, we can categorize video anomaly in to 4 types, according to different spatiotemporal context considered, as shown in Tab. 2. We have investi-

Table 2: Anomaly categorization by spatial and temporal context

Context	Single object	Multiple objects
Short time	Point anomaly	Co-occurrence anomaly
Long time	Sequential anomaly	Interaction anomaly

gated the detection of point anomaly, sequential anomaly, and co-occurrence anomaly. The detection of interaction anomaly involves multiple objects with complicated temporal logic and will be our future work. Furthermore, we will consider how to incrementally update the current models as new video observations stream in, so that the model can efficiently adapt to visual contextual changes over a long period of time, as in [25].

Acknowledgement

This work was partially funded by the US Department of Transportation via the Center for the Commercialization of Innovative Transportation Technology at Northwestern University.

References

[1] C. Stauffer, W. E. L. Grimson, Learning patterns of activity using real-time tracking, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 747–757.

[2] F. Porikli, T. Haga, Event detection by eigenvector decomposition using object and frame features, in: Proc. IEEE Conf. on Comput. Vision and Pattern Recognition Workshops, Vol. 7, 2004, pp. 114–124.

[3] D. Makris, T. Ellis, Learning semantic scene models from observing activity in visual surveillance, IEEE Trans. Syst., Man, Cybern. B 35 (3) (2005) 397–408.

[4] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, S. Maybank, A system for learning statistical motion patterns, IEEE Trans. Pattern Anal. Mach. Intell. 28 (9) (2006) 1450–1464.

[5] Y. Zhou, S. Yan, T. S. Huang, Detecting anomaly in videos from trajectory similarity analysis, in: Proc. IEEE Int'l Conf. on Multimedia and Expo, 2007, pp. 1087–1090.

[6] N. Anjum, A. Cavallaro, Multifeature object trajectory clustering for video analysis, IEEE Trans. Circuits Syst. Video Technol. 18 (11) (2008) 1555–1564.

[7] F. Jiang, Y. Wu, A. K. Katsaggelos, A dynamic hierarchical clustering method for trajectory-based unusual video event detection, IEEE Trans. Image Process. 18 (4) (2009) 907–913.

[8] C. Piciarelli, C. Micheloni, G. L. Foresti, Trajectory-based anomalous event detection, IEEE Trans. Circuits Syst. Video Technol. 18 (11) (2008) 1544–1554.

[9] C. Rao, A. Yilmaz, M. Shah, View-invariant representation and recognition of actions, Int'l Journal of Comput. Vision 50 (2) (2002) 203–226.

[10] N. Cuntoor, R. Chellappa, Epitomic representation of human activities, in: Proc. IEEE Conf. on Comput. Vision and Pattern Recognition, 2007, pp. 1–8.

[11] F. I. Bashir, A. A. Khokhar, D. Schonfeld, Object trajectory-based activity classification and recognition using hidden markov models, IEEE Trans. Image Process. 16 (7) (2007) 1912–1919.

[12] B. T. Morris, M. M. Trivedi, Learning, modeling, and classification of vehicle track patterns from live video, IEEE Trans. Intell. Transp. Syst. 9 (3) (2008) 425–437.

[13] T. Xiang, S. Gong, Video behavior profiling for anomaly detection, IEEE Trans. Pattern Anal. Mach. Intell. 30 (5) (2008) 893–908.

[14] C. R. Jung, L. Hennemann, S. R. Musse, Event detection using trajectory clustering and 4-d histograms, IEEE Trans. Circuits Syst. Video Technol. 18 (11) (2008) 1565–1575.

[15] I. Saleemi, K. Shafique, M. Shah, Probabilistic modeling of scene dynamics for applications in visual surveillance, IEEE Trans. Pattern Anal. Mach. Intell. 31 (8) (2009) 1472–1485.

[16] H. Zhong, J. Shi, M. Visontai, Detecting unusual activity in video, in: Proc. IEEE Conf. on Comput. Vision and Pattern Recognition, Vol. 2, 2004, pp. 819–826.

[17] O. Boiman, M. Irani, Detecting irregularities in images and in video, in: Proc. IEEE Int'l Conf. on Comput. Vision, Vol. 1, 2005, pp. 462–469.

[18] R. Hamid, A. Johnson, S. Batta, A. Bobick, C. Isbell, G. Coleman, Detection and explanation of anomalous activities: Representing activities as bags of event n-grams, in: Proc. IEEE Conf. on Comput. Vision and Pattern Recognition, Vol. 1, 2005, pp. 1031–1038.

[19] X. Wang, X. Ma, W. E. L. Grimson, Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models, IEEE Trans. Pattern Anal. Mach. Intell. 31 (3) (2009) 539–555.

[20] F. Jiang, Y. Wu, A. K. Katsaggelos, Detecting contextual anomalies of crowd motion in surveillance video, in: Proc. IEEE Int'l Conf. on Image Process., 2009, pp. 1117–1120.

[21] N. M. Oliver, B. Rosario, A. P. Pentland, A Bayesian computer vision

system for modeling human interactions, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 831–843.

[22] A. Galata, N. Johnson, D. C. Hogg, Learning variable-length markov models of behavior, Comput. Vision and Image Understanding 81 (3) (2001) 398–413.

[23] B. Yao, L. Wang, S. Zhu, Learning a scene contextual model for tracking and abnormality detection, in: Proc. IEEE Conf. on Comput. Vision and Pattern Recognition Workshops, 2008, pp. 1–8.

[24] Y. Benezeth, P. M. Jodoin, V. Saligrama, C. Rosenberger, Abnormal events detection based on spatio-temporal co-occurences, in: Proc. IEEE Conf. on Comput. Vision and Pattern Recognition, 2009, pp. 2458–2465.

[25] J. Kim, K. Grauman, Observe locally, infer globally: A space-time mrf for detecting abnormal activities with incremental updates, in: Proc. IEEE Conf. on Comput. Vision and Pattern Recognition, 2009, pp. 2921–2928.

[26] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM Comput. Surv. 41 (3) (2009) 1–58.

[27] X. Yan, J. Han, R. Afshar, Clospan: Mining closed sequential patterns in large datasets, in: Proc. IEEE Int'l Conf. on Data Mining, 2003.

[28] V. I. Levenshtein, Binary codes capable of correcting deletions, insertions, and reversals, Soviet Physics Doklady 10 (8) (1966) 707–710.

[29] T. Uno, M. Kiyomi, H. Arimura, Lcm ver.2: Efficient mining algorithms for frequent/closed/maximal itemsets, in: Proc. IEEE Conf. on Data Mining Workshop on FIMI, 2004.

[30] J. Yuan, Y. Wu, M. Yang, From frequent itemsets to semantically meaningful visual patterns, in: Proc. ACM SIGKDD Conf. on Knowl. Discovery and Data Mining, 2007, pp. 864–873.

[31] T. M. Cover, J. A. Thomas, Elements of Information Theory, John Wiley & Sons, Inc., 1991.