

Anomalous Event Detection in Traffic Video Based on Sequential Temporal Patterns of Spatial Interval Events

Ashok Kumar P.M¹ and Vaidehi.V²

^{1,2}Department of Electronics Engineering & AU-KBC Research Centre, MIT Campus.

Anna University, Chennai-600 044, India.

[E-mail: profpmashok@gmail.com]

[E-mail: vaidehi@annauniv.edu]

*Corresponding author: Ashok Kumar P.M

*Received July 28, 2014; revised October 31, 2014; accepted November 20, 2014;
published January 31, 2015*

Abstract

Detection of anomalous events from video streams is a challenging problem in many video surveillance applications. One such application that has received significant attention from the computer vision community is traffic video surveillance. In this paper, a Lossy Count based Sequential Temporal Pattern mining approach (LC-STP) is proposed for detecting spatio-temporal abnormal events (such as a traffic violation at junction) from sequences of video streams. The proposed approach relies mainly on spatial abstractions of each object, mining frequent temporal patterns in a sequence of video frames to form a regular temporal pattern. In order to detect each object in every frame, the input video is first pre-processed by applying Gaussian Mixture Models. After the detection of foreground objects, the tracking is carried out using block motion estimation by the three-step search method. The primitive events of the object are represented by assigning spatial and temporal symbols corresponding to their location and time information. These primitive events are analyzed to form a temporal pattern in a sequence of video frames, representing temporal relation between various object's primitive events. This is repeated for each window of sequences, and the support for temporal sequence is obtained based on LC-STP to discover regular patterns of normal events. Events deviating from these patterns are identified as anomalies. Unlike the traditional frequent item set mining methods, the proposed method generates maximal frequent patterns without candidate generation. Furthermore, experimental results show that the proposed method performs well and can detect video anomalies in real traffic video data.

Keywords: Gaussian Mixture Models, Block Motion Estimation, Primitive Events, Lossy Count based Approach, Sequential Temporal Patterns.

1. Introduction

Recently, there has been an increasing demand for surveillance cameras in public places for addressing security, safety and monitoring issues. Thus, there is a need for automatic detection of anomalous or abnormal events in surveillance videos. However, robust and accurate detection of abnormal events in a video still remains a difficult problem for the majority of computer vision applications due to its size, unrestricted flow of video streams and complex composite events involving multiple objects interactions, e.g., jay-walking (people cross the road while vehicles pass by) and vehicle overtaking [2,3,8, 11,16,25,32].

Most of the works in this field are based on modeling the statistical features of the background, the appearance of the foreground objects (person, car, bicycle, etc.) and foreground dynamics (such as location and motion at different times). These object features are used in characterizing video events. However, these approaches [2, 8] ignore important spatial and temporal contextual information.

Instead of relying solely on trajectories, the primitive events of the object are defined as the basic units (along with spatial and temporal symbols) for describing more complicated activities and interactions. In traffic applications, these multiple primitive events usually persist for an interval of time, and the main goal is to detect anomalous events with complex temporal relationship between the primitive events. First, the temporal patterns involving multiple spatial primitive events with high frequency are calculated. These regular temporal patterns describe various scenarios in traffic video sequences such as when a car stops to wait for a pedestrian passing by, when cars move at the same time in the same lane or different lanes, etc.

To the best of our knowledge, there have been no efficient methods developed for mining frequent time interval-based patterns (referring as temporal patterns) from sequences of video data. Traditional sequential mining approaches cannot be applied for detecting anomalous events in surveillance videos as they can handle only instantaneous events and static databases. Moreover, they are not capable of detecting anomalous events based on event intervals and stream of video data.

This paper proposes the Lossy Count based approach for Sequential Temporal Patterns (LC-STP), in which primitive events of each object are analyzed to form a regular expression. Regular expression represents temporal relation between various objects' primitive events in a sequence of frames. The support for this temporal sequence is calculated for finding regular patterns of normal events. Events that deviate from these patterns are identified as anomalies.

The proposed approach is applied 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 the proposed approach can automatically infer regular patterns of traffic motion in the training phase and detect spatio-temporal anomalous events due to multiple objects over different regions in the testing phase.

The main contribution of this work includes:

- (a) A formal definition of the problem of mining frequent temporal arrangements of primitive event intervals in a sequence of frames,
- (b) A novel notation for representing timing information associated with the spatial primitive events,
- (c) An efficient algorithm for mining maximal frequent temporal patterns in a sequence of

- frames, and
- (d) An extensive experimental evaluation of these techniques using real traffic datasets.

The rest of this paper is organized as follows: Section 2 describes the related works. The problem definition and formulation, notations used in this paper are defined in Section 3. Section 4 describes the proposed work, which includes an algorithm for finding both regular and irregular temporal patterns between various objects in a video. In Section 5, we demonstrate the effectiveness of the proposed algorithm through a series of experiments. In Section 6, the conclusions and further research are discussed.

2. Related Works

The abnormal event detection in a video sequence has been an active area. Some of the works as in Dong et al. [3] use directional motion behavior descriptor to compare with normal behavior descriptor. In addition to motion, the foreground region area, shape factors and pixel velocity vector are used as features [11] for simple classifier to determine the objects' normal or abnormal states. In the work of Chen et al. [2], a support vector machine (SVM) was selected as a classifier for features such as incident point velocity, downstream and upstream velocities, incident point occupancy rate, and upstream and downstream occupancy rates. Instead of motion features, the entire trajectory information was used for SVM [25]. In some works [32], Kalman filter and motion models were used for tracking, and any deviation in this resulted in an anomaly.

Most of the works are based on trajectory information [4,22,31,38] and their corresponding modeling techniques used are hidden Markov model (HMM) [4,31], coupled HMM [22], and 3-D graphs [38]. While in Gilbert et al. [5], 2D corners were grouped spatially and temporally using a hierarchical process to learn descriptive and distinctive features for action recognition.

Some works reveal spatiotemporal dependencies of moving agents in complex dynamic scenes, such as the right way between different lanes or typical traffic light sequences, using the Markov random field model, the topic model, and the dependent Dirichlet processes [12,13]. Wang et al. [34] have used hierarchical Bayesian models to connect three elements in visual surveillance: low-level visual features, simple atomic activities, and interactions. Thus, a summary of typical atomic activities and interactions occurring in the scene was provided, and video anomalies were detected at different levels.

Though there are several schemes for detecting anomalies in the video, there is a need to address the primitive event intervals based anomalies. Thus, this paper proposes a representation of video frames in terms of primitive events. In order to detect anomalous video events based on temporal context, frequent temporal pattern mining based approach is performed on a stream of video sequences.

3. Problem Definition

In general, traffic video sequences contain time interval based spatial events, which exist over a sequence of continuous frames. Each sequence id is represented as the set of consecutive transactions, i.e., frames in our case. Each frame is represented as a single transaction, and each transaction will have a set of primitive events. Since the time of conclusion of the primitive events is not known, direct application of traditional sequential mining techniques becomes difficult. Thus, we have introduced novel notations for

preserving the timing information associated with each spatial primitive event. After summarizing, these data are grouped into sequences. From such a list of sequences, the application of proposed LC-STP algorithm leads to the discovery of maximal patterns. The following definitions formally describe this mining problem.

Definition 3.1 Spatial Primitive Event:

Each event corresponds to the individual object's id, abstract location and its time point with respect to each frame. Thus, the event e is defined by three related attributes: its abstract spatial location, the object id number and its time point. Accordingly, the following notations are defined for a primitive event e :

- (1) $e.loc$ refers to the abstract spatial location.
- (2) $e.id$ refers to the object id number in $e.loc$ attribute.
- (3) $e.time$ refers to the notation of the object's temporal state, i.e., start or end or continue time state as $+, -, *$ respectively.

The temporal state and object id information are represented as superscript and subscript to the spatial location respectively.

Example 1. Assume that in a particular frame of a video sequence, there are three moving objects: O_1, O_2, O_3 . O_1 just started in location A, O_2 continued the movement from earlier frames in location B and O_3 finished its movement in location C. Based on the definition 3.1, the event representation for those objects O_1, O_2, O_3 will be in A_1^+, B_2^*, C_3^- respectively.

Definition 3.2 Transaction:

Consider each frame as a single transaction. A transaction is a set of primitive events such as $T_i = \langle e_1, e_2, \dots, e_n \rangle$, where e_n is a spatial primitive event with respect to each object 'n' for that frame 'i'.

Example 2. Consider the previous example 1. If all the events corresponding to different objects occur in a particular frame, based on definition 3.2, then transaction T is represented as $\langle A_1^+, B_2^*, C_3^- \rangle$.

Definition 3.3 Sequence of Transactions:

A sequence of transactions, seq-id is represented by a sequence of frames $seq-id = \langle T_1, T_2, \dots, T_n \rangle$, where $ts(T_i) < ts(T_j)$ for $i < j$. The $ts(T_i)$ means the time-stamp at which T_i has been issued.

Example 3. Consider the sequence of frames in which Object O_1 in location A starts at frame 1 and finishes at frame 3 at location B, object O_2 starts at frame 2 and finishes at location B of frame 3, and object O_3 starts at frame 2, location C, and still continues at frame 3 at location B. Based on the definitions 3.1, 3.2, 3.3, Seq-id₁ is represented as $\{ \langle A_1^+ \rangle, \langle A_1^+, B_2^+, C_3^+ \rangle, \langle B_1^-, B_2^-, B_3^- \rangle \}$.

Definition 3.4 Event interval temporal relation:

Allen's temporal logic describes the 13 possible relations for any pair of state intervals. Out of 13 relations, seven relations are: before, meets, overlaps, is-finished-by, contains,

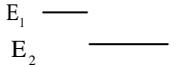
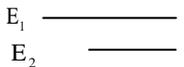
starts, equals and other six relations are simply their inverses. Allen's relations have been used by most works on mining time interval data [7,22,24,29,35]. The use of all the Allen's relations is not appropriate in our case of traffic videos. All the moving objects are dependent only on the traffic control signal and they are mutually independent. Thus, the two relations "finished-by (E_1, E_2)" and "starts (E_1, E_2)" are not considered for our traffic video sequences. Therefore, in this paper only five temporal relations (as shown in **Table 1**):

Before (E_1, E_2), Meets (E_1, E_2), Overlaps (E_1, E_2), Contains (E_1, E_2), Equals (E_1, E_2) are used and defined as follows:

Given two event intervals E_1 and E_2 :

- (1) E_1 Before E_2 , if $E_1^+ < E_2^- < E_2^+$, which is the same as Allen's before relation and is denoted as $E_1 \rightarrow E_2$.
- (2) E_1 Meets E_2 , if $E_1^+ < E_1^- = E_2^+ < E_2^-$ i.e., E_1 starts before E_2 and end time of E_1 equals to start time of E_2 . It is denoted as $E_1 E_2$.
- (3) E_1 Overlaps E_2 , if $E_1^+ < (E_1^*, E_2^+) < (E_1^-, E_2^-) < E_2^-$ i.e., E_1 starts before E_2 and end time of E_1 is in between start time and end times of E_2 . It is denoted as $E_1 > E_2$.
- (4) E_1 Contains E_2 , if $E_1^+ < (E_1^*, E_2^+) < (E_1^-, E_2^-) < E_1^-$ i.e., E_1 starts before E_2 . Start time and end times of E_2 is in between start time and end time of E_1 . It is denoted as E_1 / E_2 .
- (5) E_1 Equals E_2 , if $(E_1^+, E_2^+) < (E_1^*, E_2^*) < (E_1^-, E_2^-) < (E_1^-, E_2^-)$ i.e., Start time and end times of E_2 is equal to start time and end time of E_1 . It is denoted as $E_1 = E_2$.

Table 1. Temporal Pattern Expression Representation and Notation

Sl.no	Temporal Relation	Pictorial Example T_1 T_2 T_3 T_4	Sequence of Transaction	Temporal Interval Pattern notation
1.	E_1 before E_2		$\langle E_1^+ \times E_1^- \times E_2^+ \times E_2^- \rangle$	$E_1 \rightarrow E_2$
2.	E_1 meets E_2		$\langle E_1^+ \times E_1^-, E_2^+ \times E_2^+ \times E_2^- \rangle$	$E_1 \sqcap E_2$
3.	E_1 overlaps E_2		$\langle E_1^+ \times E_1^-, E_2^+ \times E_1^-, E_2^+ \times E_2^- \rangle$	$E_1 > E_2$
4.	E_1 is finished by E_2		$\langle E_1^+ \times E_1^-, E_2^+ \times E_1^-, E_2^+ \times E_1^-, E_2^- \rangle$	--
5.	E_1 contains E_2		$\langle E_1^+ \times E_1^+, E_2^+ \times E_1^+, E_2^+ \times E_1^- \rangle$	E_1 / E_2
6.	E_1 starts E_2		$\langle E_1^+, E_2^+ \times E_1^+, E_2^+ \times E_1^-, E_2^+ \times E_2^- \rangle$	--
7.	E_1 equals E_2		$\langle E_1^+, E_2^+ \times E_1^+, E_2^+ \times E_1^+, E_2^+ \times E_1^-, E_2^- \rangle$	$E_1 = E_2$

Definition 3.5 Temporal Patterns:

In order to obtain temporal descriptions of the data, we combine basic spatial events along with temporal relations to form temporal patterns. There are two kinds of temporal patterns that exist in a traffic video. The first is called an ‘‘intra-event pattern’’, which represents the frequent temporal relation that exists within each spatial region. The other kind of multi-label temporal pattern is an ‘‘inter-event pattern’’, which represents the frequent temporal relation that exists between each pair of spatial region. The regular temporal patterns are represented as per the notations defined in 3.4.

Definition 3.6 Temporal Relation b/w intra-primitive events:

The temporal relation between intra-events is defined as the temporal pattern that exists between each pair of spatial primitive events within the same region. The temporal relation is calculated for each pair of events based on definition 3.4. The intra-temporal relation for each sequence of frames is constructed with the help of $\{ >, =, |, \square \}$ operators, and its count is maintained and is given as

$TR_i : \{S_1 - (R_1), S_2 - (R_2), \dots\}$, in i^{th} sequence, for all possible set regions S. The same procedure is repeated for the next sequence of frames.

Example 4.

1. $TR_1 : \{A - (>, \rightarrow, =) \}$ - semantic interpretation: The intra-temporal relations found in region A are {Overlaps, Before, Equals}.
2. $TR_2 : \{B - (>, =) \}$ - semantic interpretation: The intra-temporal relations found in region B are {Overlaps, Equals}.

Definition 3.7 Relation B/W inter-primitive events:

The temporal relation between inter-primitive events is defined as the temporal pattern that exists between each pair of spatial primitive events belonging to different regions. The temporal relation is calculated for each pair of regions based on definition 3.4 and is given as $\{ S_1 (R_1) S_2, S_1 (R_1) S_3, \dots \}$, containing temporal operators R that exists in between regions.

Example 5.

1. $TR_1 : A - \{>, =\} - C$ semantic interpretation: The inter-temporal relation found between objects moving in regions A and C are { Equals, Overlaps }.
2. $TR_2 : A - \{>, =\} - B$, semantic interpretation: The inter-temporal relation found between objects, moving in regions A, B are { Equals, Overlaps }.

Based on the above definitions, the problem of mining frequent temporal arrangements can be formulated as shown below.

Given a stream of e-sequence of window size w and a support threshold min-sup, the task is to find frequent intra-event and inter-event patterns of the form

$F_{intra} = \{S_1 - (R_1), S_2 - (R_2), \dots\}$, for all possible set regions S containing temporal operators that exists within that region, and

$F_{inter} = \{ S_1 - (R_1) - S_2, S_1 - (R_1) - S_3 \dots \}$, for all possible combinations between the different set regions, where R_i is a set containing temporal operators that exists between regions.

4. Detection of Abnormal Temporal pattern Framework

In this section, a Lossy Count based Sequential Temporal Pattern mining approach (LC-STP) is proposed for detecting abnormal patterns in traffic scenes based on the spatio-temporal context. There are two phases in this method. In the training phase, a portion of normal surveillance video frames is selected as a training sample to generate frequently occurring temporal patterns (regular events) with the interval based spatial events. Here, the class labels are not assigned manually, and the regular temporal patterns are found out by the proposed LC-STP algorithm. In the testing phase, the incoming temporal patterns for each sequence of frames are compared with the stored regular temporal patterns. The proposed framework (as shown in Fig. 1) consists of object detection and tracking, data stream conversion, novel frequent temporal pattern mining and temporal pattern matching.

4.1 Object detection and Tracking

In this paper, our emphasis is mainly on event modeling tasks. Thus, the state of art techniques such as Gaussian mixture models (GMM) [9] are used for object detection and morphological operations are applied for noise elimination to get the resultant blob. The main reason for choosing GMM is its ability to deal with lighting changes, repetitive motions of scene elements.

Object tracking is done in two steps (i.e., tracking by detection technique):

1. Detecting moving objects in first frame, and
2. Associating the detections corresponding to the same object over time.

Our simple tracking method works as follows:

1. Input: The input video is converted into a sequence of frames $(f_1, f_2, f_3, \dots, f_n)$.
2. Initialization: Let f_0 be the first frame for which detection is available with GMM [9]. A new track set T_j^0 for each blob detection j is created.
3. Iteration: Loop over frames i from f_1 to f_n ,
4. Object detection and track assignment: Perform block motion estimation using the three step search algorithm [26] based on the block matching criteria, i.e., mean square error (MSE).

$$MSE(d_1, d_2) = \frac{1}{N_1 \times N_2} \sum_{(n_1, n_2) \in B} [S(n_1, n_2, k) - S(n_1 + d_1, n_2 + d_2, k + 1)]^2 \quad (1)$$

where, B is $N_1 \times N_2$ block of pixels, and $s(x, y, k)$ denotes a pixel location at (x, y) in frame k , (d_1, d_2) is the displacement of the center pixel of the block.

5. Delete lost tracks: Each track keeps count of the number of consecutive frames, where it remained unassigned. If the count exceeds a specified threshold, (i.e., object might have left out of that frame), then we delete those tracks.
6. Create new tracks: New blobs are introduced by applying GMM [9] at every 10 frames. Noisy object detection is eliminated with the help of size and appearance.
7. Output: In our case, object's id, spatial location and name are extracted to form a formatted output for further processing in our proposed framework.
8. New and old blobs are tracked by using steps 4–7.

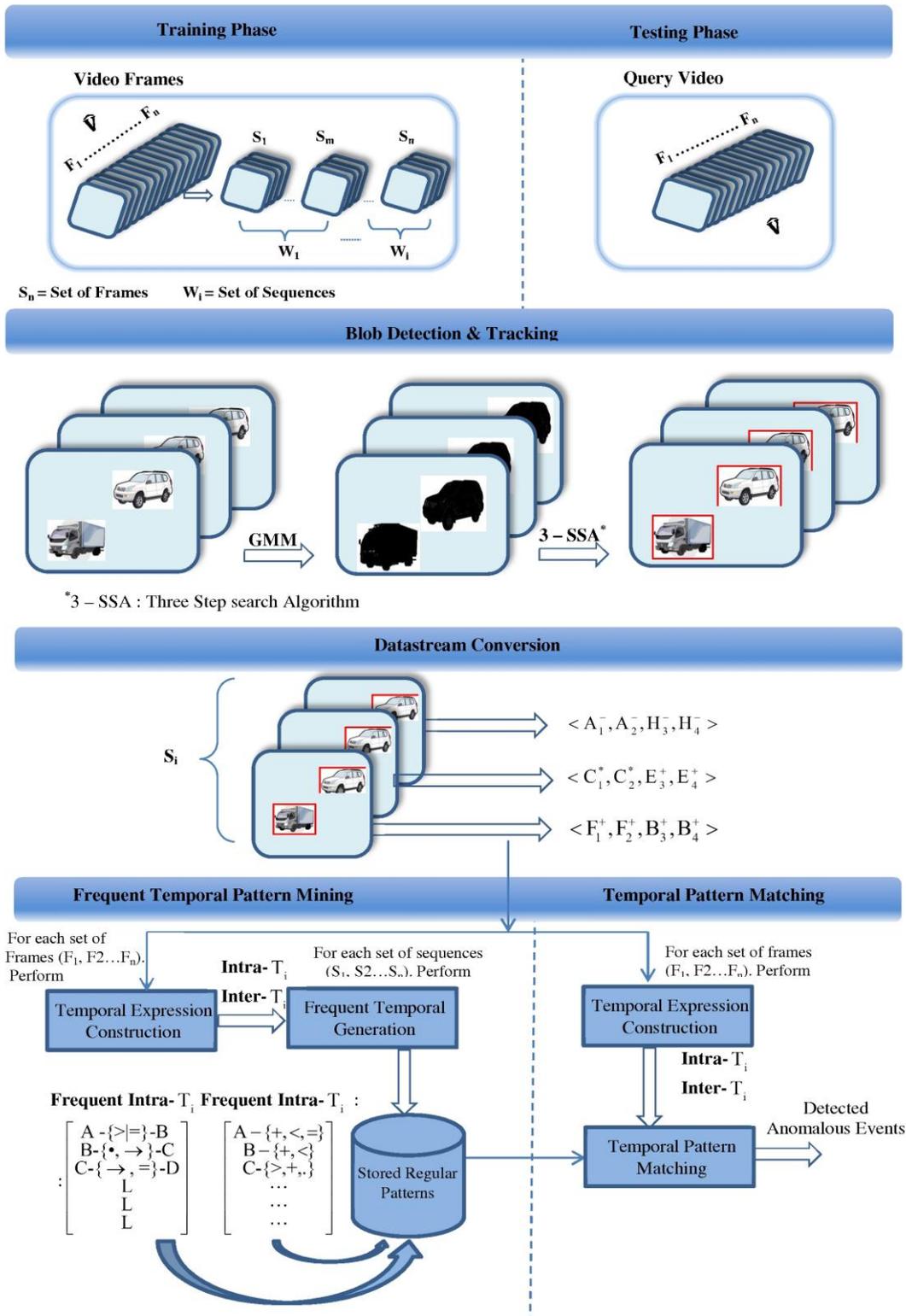


Fig. 1. Proposed framework for Abnormal Event Detection System.

4.2 Data Stream Conversion based on spatial atomic events

First, the static background frame of a traffic video is divided into segments based on lanes, junctions, and zebra crossings and are labeled as per alphabetical order (A, B, C, D, E...) as shown in **Figs. 2(a) & 2(b)**. The primitive spatial events are formed for each object based on the objects-id, its abstract spatial location, and its timing information. These events are grouped for each frame as per definition 3.1 to form a transaction, $T_i = \langle e_1, e_2, \dots, e_n \rangle$, where e_n is a spatial primitive event with respect to each object n for that frame i . A sequence of transaction is formed as per definition 3.2,

seq-id = $\langle T_1, T_2, \dots, T_n \rangle$, for each set of 'n' consecutive frames. For each sequence, seq-id candidate temporal regular pattern is constructed. Frequent pattern mining is performed over a window of sequences to form regular temporal patterns.

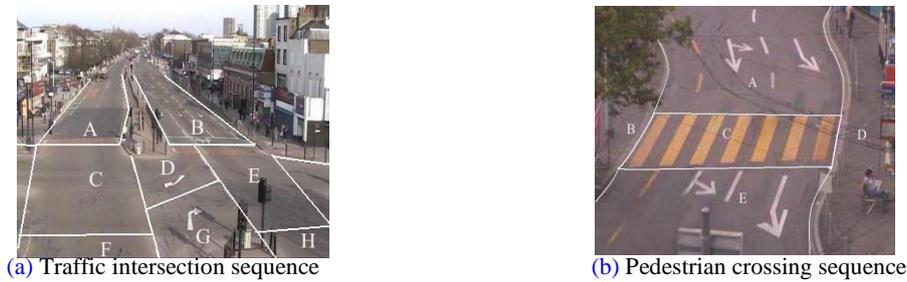


Fig. 2. Label assignment of video sequence

Example 6: In **Fig. 2(a)**, if the object with id's O_1 and O_2 is moving in a straight road from bottom left to top left in one sequence and other object with id's O_3 & O_4 is moving from top right to bottom right in another sequence of frames, then sequence of data streams are created as shown in **Table 2**.

Table 2. Data Stream Conversion

Sequence ID	Data Items
S_i	
S_1	$\langle F_1^+, F_2^+ \times C_1^*, C_2^* \times A_1^-, A_2^- \rangle$
S_2	$\langle B_3^+, B_4^+ \times E_3^*, E_4^* \times H_3^-, H_4^- \rangle$
S_3	$\langle F_1^+, F_2^+, B_3^+, B_4^+ \times C_1^*, C_2^*, E_3^*, E_4^* \times A_1^-, A_2^-, H_3^-, H_4^- \rangle$

Example 6: In **Fig. 2(b)**, consider the scenario of person-id P_1 crossing the road through the regions B, C, D. While the object with id O_2 , stopped at junction A and object id O_3 , moving at region E. Then the sequence of data streams is created as shown in **Table 3**.

Table 3. Data Stream Conversion

Sequence ID	Data Items
S_i	
S_1	$\langle B_1^+, E_2^+ \times B_1^-, E_2^-, A_3^+ \times E_2^*, A_3^*, C_4^+ \rangle$
S_2	$\langle E_1^+, C_2^+, A_3^+ \times E_1^*, C_2^*, A_3^* \times E_1^-, C_2^-, A_3^- \rangle$
S_3	$\langle E_1^+, C_2^+, A_3^+ \times E_1^-, D_2^+, C_3^+ \times D_2^-, C_3^- \rangle$

4.3 Frequent Temporal Pattern Mining

The streams of transaction (i.e., frames) with spatial primitive events are grouped to form sequences. The method of finding frequent temporal patterns in sequence of frames consists of two phases: temporal expression construction and frequent pattern generation and is presented in **Table 4**.

Using $\text{construct-temp-exp}(F_i)$, Temporal expression is constructed for each set sequences (S_1, S_2, \dots, S_n) to form candidate intra and inter-temporal patterns. The intra-temporal patterns consist of a triplet of labeled region, the temporal relation in that region and its support count. The same is repeated for every sequence, and the support count is updated if the relation already exists. If the relation is a new one, we add the expression and maintain a count.

Table 4. A Lossy Counting based algorithm for Sequential Temporal Patterns (LC-STP)

Input: Data Stream S_n , error rate ϵ , support threshold s
Output: The set D Frequent Temporal Patterns in a Sequence of frames.
1. Begin
2. Initially intra- D , inter- D is empty, $n=0$
3. For each sequence S_i , $i=0$ to L , do
4. For each frame F_i , $i=0$ to bucket width w , do
5. $T_i = \text{Construct-Temp-Exp}(F_i)$; // T_i, T_j represents current & stored temporal patterns.
6. end for
7. $T_j = \text{generate-Temp-Pattern}(T_i)$;
8. End for
9. For every n sequences,
10. Delete T_j ; $\forall T_j \in \text{inter-}D$ and $\text{intra-}D$ where $f(T_j) < (s - \epsilon)$
11. End for
12. Output T_j , $\forall T_j \in \text{inter-}D$ and $\text{intra-}D$; // Displays Temporal-Patterns.
13. End
14. Function $\text{generate-Temp-Pattern}(T_i)$
15. If T_i exists in $(T_j \in D)$ Then
16. $f(T_j) = f(T_j) + 1$;
17. //existing Temporal Pattern T_j is retained. Count of the pattern T_j is incremented.
18. end if
19. else if $T_j \supset (T_i \in D)$ Then
20. $T_j = T_i$; // T_j is replaced with T_i
21. $f(T_j) = f(T_j) + 1$; //count of the Frequent Pattern T_j is incremented.
22. end if
23. else // Store new patterns in D
24. add-entry($T_i, 1$) to D ;
25. End

```

26. Function Construct-Temp-Exp (  $F_i$  )
27.   Arrange( $F_i$ ); // To arrange spatial events in Alphabetical order
28.   For each new event  $e_i$  .loc in  $F_i$  (n)
29.   |   Create-Buffer( $e_i$  .loc),  $\forall e_i$  .loc , //Buffer,  $b_i=A,B,C,D,E,F,G,H$ . & store events
30.   |   End for
31.   |   // loop for Intra-Temporal Patterns
32.   |   For each event  $e_i$  in Buffer,  $b_i=A,\dots,H$ ;  $i=1\dots n$ 
33.   |   |   For each event  $e_j$  in Buffer,  $b_j=A\dots H$ ;  $j=1,\dots n$ 
34.   |   |   |   If(  $e_i$  .loc ==  $e_j$  .loc)
35.   |   |   |   |    $R_i = f(e_i, e_j)$ ; //As per 3.6 definition
36.   |   |   |   |   Intra-  $T_i = \text{add}(e_i$  .loc,  $R_i)$  ; // storing in temporal intra- pattern
37.   |   |   |   |   End if
38.   |   |   |   End for
39.   |   |   End for
40.   |   //loop for Inter-temporal Patterns
41.   |   For each event  $e_i$  in Buffer  $b_i=A\dots H$ ;  $i=1,\dots n$ 
42.   |   |   For each event  $e_j$  in Buffer,  $b_j=A\dots H$ ;  $j=1,\dots n$ 
43.   |   |   |   If(  $e_i$  .loc !=  $e_j$  .loc)
44.   |   |   |   |    $R_i = f(e_i, e_j)$ 
45.   |   |   |   |   Inter-  $T_i = \text{add}(e_i$  .loc,  $R_i, e_j$  .loc);
46.   |   |   |   End for
47.   |   |   End for
48.   |   End for
49. End function

```

Similarly, in inter-temporal pattern expressions are formed for all possible combinations of the region. The expression consists of four items: a pair of labeled set regions, the temporal relation between them, and its support count. The same procedure of forming inter-temporal patterns is performed for every sequence, and the support count is updated if that temporal relation already exists. If the relation is a new one, we add the relation in the expression and start maintaining the count.

Frequent temporal patterns are generated using generate-temp-pattern(T_i) for each set window of sequences (W_1, W_2, \dots, W_n). After each window sequence, infrequent patterns are removed with frequency less than min-support 's' and error threshold 'e'. This method of finding frequent temporal patterns is similar to the lossy count algorithm [20] and is presented in Table 4. This approach eliminates the need for finding sub-sets or candidate generation. Thus, specialized data structures (trie, suffix tree, etc.,) are not needed for subsets or candidate generation.

In LC-STP, the user has to supply parameters such as support 's', error 'e', bucket width 'w', and sequences length 'l'. We maintain a data structure D that stores both inter- and intra-temporal patterns in that sequence, and the frequency count is maintained for the same sequence. In the initial stage, D is empty. The temporal pattern of the first sequence is added to the initial set. The incoming sequence frames are conceptually divided into buckets of width w and are labeled with the bucket id. Therefore, each bucket consists of a set of

overlapping sequences. Each sequence (i.e., transactions sequence) is processed to form inter- and intra-temporal pattern expressions, and the frequency count of the temporal relation is updated if it is an old one or is created if it is a new one. This is performed over a set of overlapping buckets to find frequent temporal patterns, and infrequent ones are deleted if its frequency $f < (s - \epsilon)$.

In the LC-STP algorithm (as shown in **Table 4**), lines 1–14 represents the main code. We call function Construct-Temp-Exp (f_i) for every sequence to construct inter- and intra-temporal patterns and call Generate-Temp-Pattern (T_i) at the end of every sequence to calculate the count of the temporal patterns generated for that sequence. Infrequent temporal patterns are removed based on the frequency $f < (s - \epsilon)$ as shown in line 12. The threshold value is determined by the user based on the length of the training video and periodicity of the traffic rules.

The main idea is that the frequently generated patterns are always regular. Thus, infrequent ones will be the anomalous events or traffic rule breakers. In the training phase, videos with high frequency of regular traffic events are taken as input, and frequently occurring temporal patterns are found without manual labeling.

4.4 Pattern Matching

In the testing phase, object detection and tracking, data stream conversion and temporal expression construction tasks are performed to find both (intra & inter) temporal sequence patterns for each sequence (i.e., set of frames). These patterns are matched with the stored regular Intra- T_i & Inter- T_i patterns, to find the irregular patterns. If the temporal sequence pattern of each sequence is equal or a subset of the stored patterns, then it is considered as normal events. Otherwise, that sequence is considered as an abnormal event sequence.

The Brute Force algorithm [21] is used to compare the patterns to the frequently occurring patterns, i.e., match each relation by relation with stored Inter-D and Intra-D patterns.

Table 5. Abnormal detection Algorithm for Sequential Temporal Patterns using simple pattern matching method

1.	Begin
2.	For each sequence S_i , $i=0$ to L , upto bucket width w , do
3.	For each frame, F_i , $i=0$ to EOS do
4.	$T_i = \text{Construct-Temp-Exp}(F_i)$;
5.	// T_i , T_j represents current & stored of Temporal patterns.
6.	End for
7.	Compare(T_i , Intra-D, Inter-D); //compare with Stored patterns
8.	End for
9.	Go to step 1

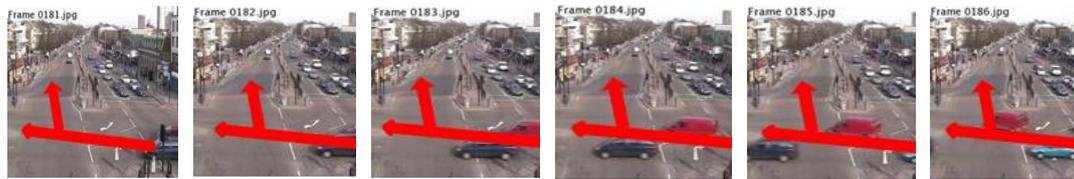
5. Results and Discussions

In this section, the experimental study and results are presented for the evaluation of the proposed approach. This approach for anomaly detection is applicable to many different

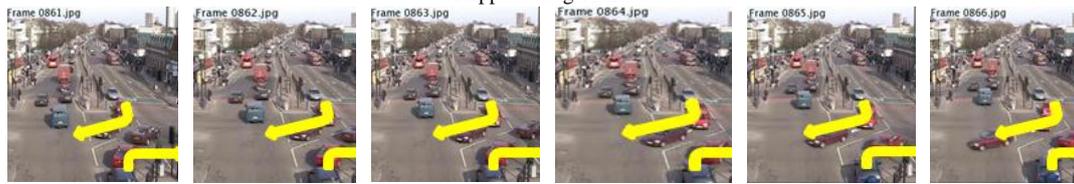
scenarios. Most motions in this scenario are normal while only a few are outliers. The task is to automatically mine these regular temporal rules of normal motion from all the data and to detect any anomalous motions that broke the rules.

5.1 Traffic intersection scenario

In order to evaluate the proposed abnormal event detection framework, a busy traffic data set containing videos of one hour length (9,000 frames) and frame size of $360 * 288$ is taken [39] as shown in Fig. 3.



(a) Regular pattern showing objects crossing from right to left side of the frame through regions E,G,D,C,A while objects stopped at region B



(b) Regular pattern showing objects moving through D, C/D, C, A and G,E regions. while objects are moving in region A



(c) Regular pattern showing objects crossing from left to right side of the frame through regions C,D,G,E/ C,D,G,E H while objects stopped at region B



(d) Regular pattern showing objects moving through F, C, A and B, E, H regions while objects stopped at region D,G

Fig. 3. Sequence showing Regular Traffic patterns

The data set contains all the regular and irregular traffic patterns. The proposed LC-STP algorithm (as shown in Table 4) is applied for determining regular patterns with 50% overlapping of bucket. We obtained 8,882 frequent temporal pattern and 118 irregular patterns (out of 9,000 frames).

This video monitors a four-road intersection. Each road consists of a two-way lane for moving objects in both directions. The entire moving traffic in this area is controlled by the traffic lights within the intersection. However, in the test video frame, only the top and right lanes are visible, but left and bottom lanes are not. Thus, the underlying rule of normal motion is the legal motion directed by the traffic lights. However, in the training phase, the goal is to discover the traffic spatial patterns with valid temporal patterns followed by most vehicles in this area and to detect anomalies within a lane and in-between lanes.

In the training phase, the proposed algorithm is applied in the offline mode to discover frequent temporal patterns. Since the motion of the object is small, only 1 frame for every 10 frames (key frame) is taken to improve the execution time. Then 10 frames are grouped into each sequence, which is then converted into temporal expression containing both intra and inter patterns. These temporal patterns are processed by the proposed algorithm (as shown in **Table 4**) to determine regular patterns. The results obtained are 8,882 frequent pattern sequences and 118 irregular patterns (out of total 9,000 frames).

Figs. 3(a–d) represent the regular patterns present in the traffic video sample. It shows some of the temporal sequence patterns obtained through the application of our proposed LC-STP algorithm. We showed only six frames in each sequence for display purposes. **Fig. 3(a)** shows the pattern of objects moving from the right lane to the top left lane, i.e., objects move through the E,D,C,A or E,G,C,A regions, when some objects stopped at the top right lane, i.e., objects region B (as per notations). The objects in other lanes are not visible (this will not affect the performance of our algorithm). **Fig. 3(b)** shows the pattern of objects crossing from the top right lane to the left lane, i.e., through B,G,C regions; some crossed from the bottom left to the right lane, i.e., from region G to E, and some moved in the region A. **Fig. 3(c)** shows some objects crossing the junction from left to right lane, i.e., through regions C,D,G,E; some crossed towards the bottom right lane, i.e., through regions C,D,G,H, and some moved in the region A. **Fig. 3(d)** shows the pattern of moving objects from bottom left to top left lane, top right to bottom right lane, i.e., through regions F,C,A and regions B,E,H respectively. While some are stopped at region D from B and some objects are stopped at G.

Figs. 4(a–c) represent abnormal patterns (anomalies) present in the traffic video sample. The results shown are abnormal events, which are obtained only when frequencies do not exceed the support and error thresholds. The displayed abnormal event in **Fig. 4(a)** shows the object crossing the junction from left to right lane illegally (through regions C,D,E), while all the objects followed the regular pattern of moving upside and downside of the frame (through regions F,C,A and B,E,H). **Fig. 4(b)** shows one such abnormal event of the objects, i.e., sudden crossing from region G to E or G to E,H, while all the other objects followed the regular pattern of moving upside and downside of the frame (through regions F,C,A and B,E,H). **Fig. 4(c)** shows another unusual event of objects moving from region D to region C or region D to region C,A, while all other objects followed the regular pattern of moving upside and downside of the frame (through regions F,C,A and B,E,H).

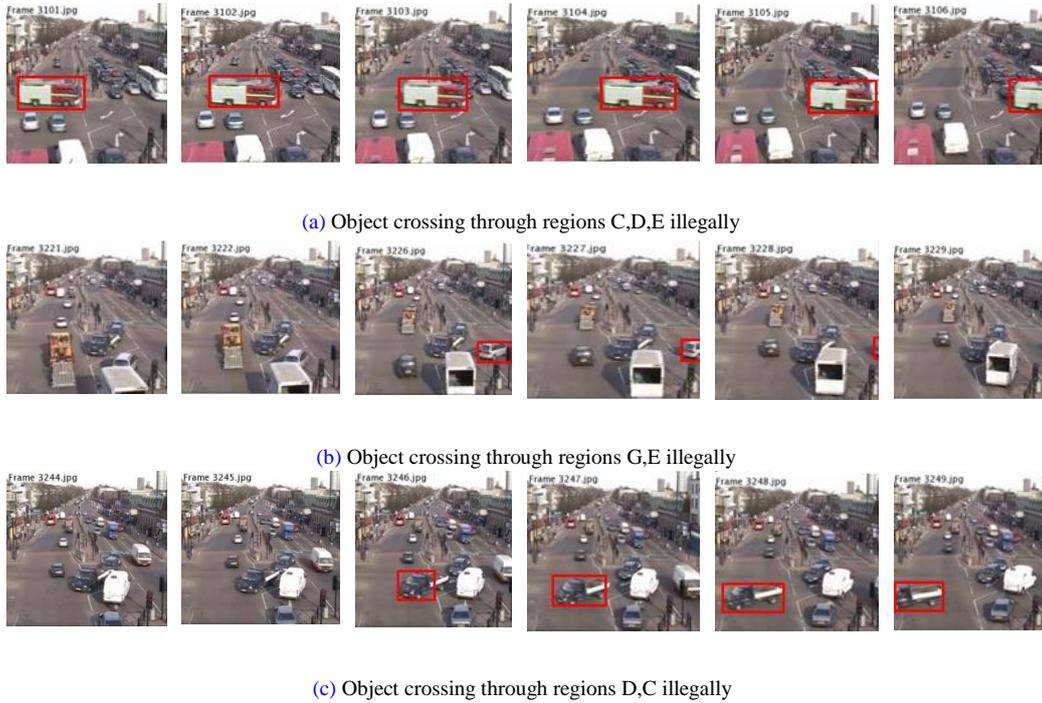


Fig. 4. Example Output Sequence showing Irregular Traffic Pattern

The proposed LC-STP algorithm detected 103 out of 118 abnormal events in the video sequence, and 12 are detected falsely as anomalies, setting the detection rate and false alarm rate of 87.2% and 10.2% respectively. Generally, the value of error rate ϵ is one by tenth of support, s . These results are tabulated with different values of ϵ , s as shown in **Table 6**.

5.2 Central Pedestrian crossing sequence:

In addition to the above scenario, the proposed work is applied on pedestrian crossing sequence videos obtained from datasets [39] to detect anomalies such as pedestrian crossing the road illegally when the vehicles are moving. **Figs. 5(a–d)** represent regular patterns present in the pedestrian crossing sequence. This video contains pedestrians crossing the road through the regions B,C,D and D,C,B, (region C is zebra crossing) and vehicles moving through regions A,C,E (one way direction). Normal pattern followed in this video is that the pedestrians cross the road when the vehicles are not in the road or do not move through zebra crossing.

This video data set consists of 900 frames. Five frames are considered for each sequence, and the proposed LC-STP algorithm are applied on each sequence. Some of the results are shown in **Figs. 5(a–d)**, which depict normal regular temporal patterns. **Fig. 5(a)** shows the pattern of pedestrians moving through the zebra crossing (through region C from either B or D to D or B). **Fig. 5(b)** shows the pattern of pedestrians crossing the road (region C) while the vehicles are moving in the region A or E.

Fig. 5(c) shows the regular pattern of objects moving through A,C,E regions. **Fig. 5(d)** shows the pattern of pedestrians crossing through regions B,C,D or D,C,B. While some object stopped at region A, others moved at region E.

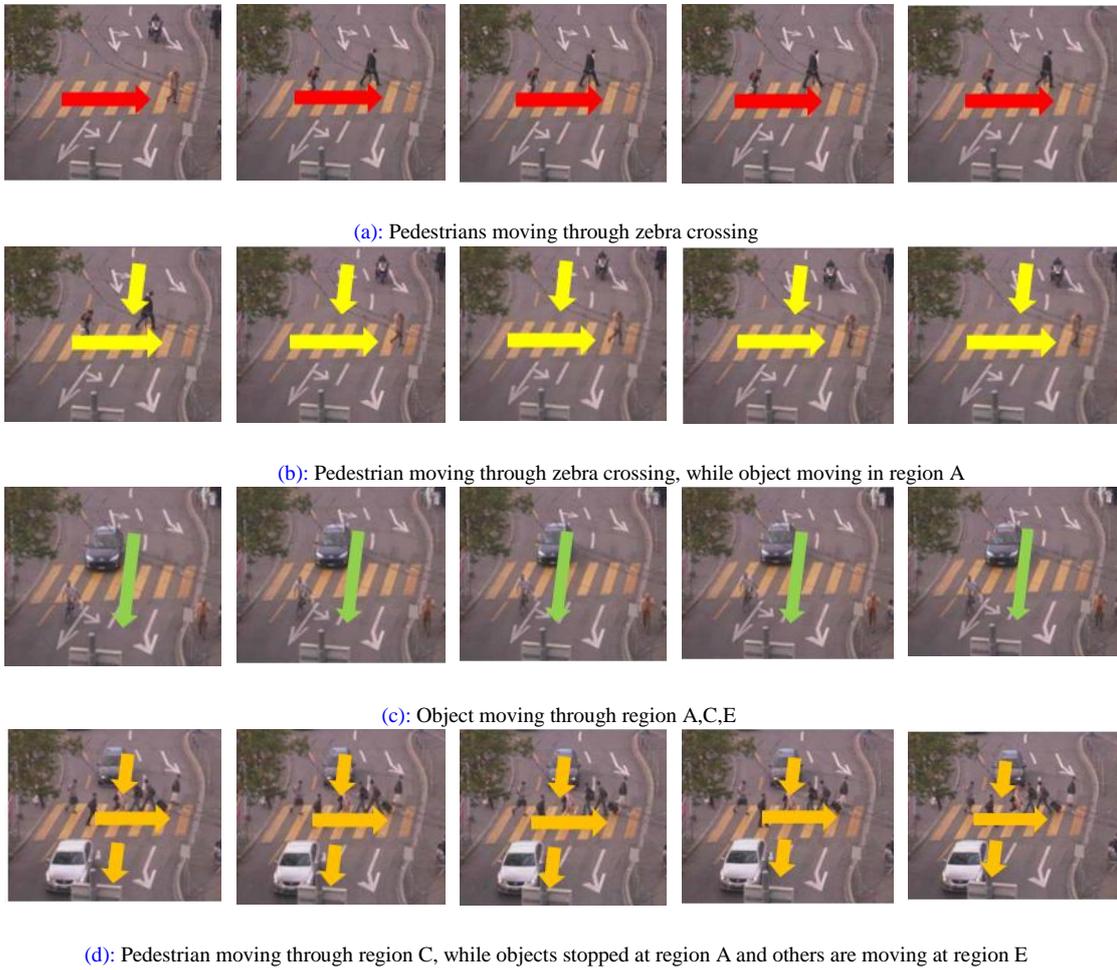


Fig. 5. Detected Pedestrian Crossing Sequence showing Regular patterns

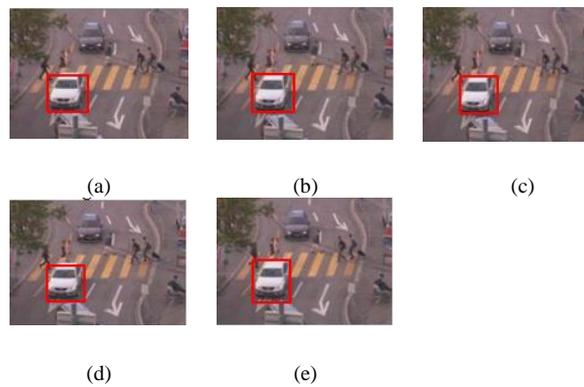


Fig. 6. People in zebra crossing, while the vehicles moving across the zebra crossing

Table 6. Statistical results of the proposed work

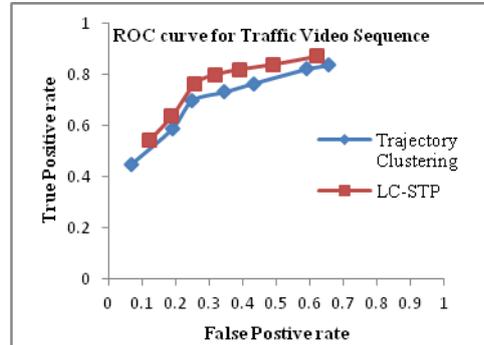
	Statistical results					
	Traffic intersection scenario [38]			Pedestrian Crossing Sequence [39]		
	$\epsilon = 0.1\%$ $s = 1.0\%$	$\epsilon = 0.05\%$ $s = 0.5\%$	$\epsilon = 0.01\%$ $s = 0.01\%$	$\epsilon = 0.1\%$ $s = 1.0\%$	$\epsilon = 0.05\%$ $s = 0.5\%$	$\epsilon = 0.01\%$ $s = 0.01\%$
Total no. of sequences	9000	9000	9000	900	900	900
Anomaly (ground truth)	118	118	118	11	11	11
Anomaly (true positive)	99	102	103	8	9	9
Anomaly (false positive)	19	16	15	3	2	2
Detection rate (%)	84%	87%	88%	73%	82%	82%
False alarm rate (%)	16%	13%	12%	27%	18%	18%

For this video data set, 11 sequences are obtained that consist of abnormal events. One of them is shown in Fig. 6, which depicts pedestrians crossing the street while the vehicle is still in the zebra crossing area (i.e., region C); the detected vehicle is shown in red box. The proposed algorithm detected 9 out of 11 abnormal events in the video sequence, and two are detected falsely as anomalies, setting the detection rate and false alarm of 81.8% and 18% respectively. These results are tabulated with different values of ϵ , s as shown in Table 6.

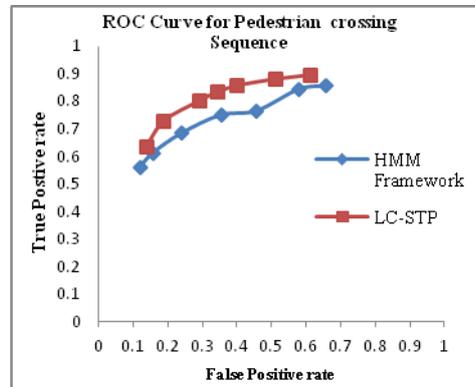
5.2 Comparison

For the comparison of the proposed LC-STP based system on anomaly detection, all the normal trajectories are extracted and clustered based on their positions. Spectral clustering is performed for all the vehicle trajectories in the traffic video sequence based on the dynamic Bayesian network [10]. The detected outliers are treated as anomalies. However, this approach fails to detect anomalies due to multiple objects as the proposed LC-STP algorithm did. For the detection of anomaly, some of the sequences are selected as abnormal, and other activities are treated as normal, for the training phase. Cross-validation is used to assess the performance of anomaly detection. Fig. 7(a) shows the ROC curves of trajectory clustering and the proposed method (LC-STP) for different values of ϵ , s in traffic video sequences.

HMM [37] is used to model the pedestrian crossing sequence. It has three states, and each state represents an event. The Gaussian mixture distribution (GM) is used to describe the state-conditional probability distributions, and the EM algorithm is used to train probability parameters. Fig. 7(b) shows the ROC curves of HMM [37] and the proposed method for different values of ϵ , s in pedestrian crossing sequences.



(a) ROC curve for traffic video sequence



(b) ROC curve for pedestrian crossing sequence

Fig. 7. Performance Comparison using ROC Curves

6. Conclusion

This paper has proposed a new approach based on frequent temporal pattern mining of interval based spatial events. The main contributions of this paper include a novel representation of spatial primitives with timing information and lossy count based sequential temporal pattern (LC-STP) approach for finding frequent temporal patterns. The proposed approach is analyzed to detect both intra- and inter-temporal spatial events. It detects anomalous events in both traffic video sequences and pedestrian crossing sequences with a high detection rate and low false alarm rate.

Experimental results show that the proposed approach is effective for abnormality detection and can be applied to many traffic video scenes, provided we should have prior knowledge of the spatial regions or lanes. However, the limitation of the current work is its dependence on the outcomes of object detection and tracking methods in videos. In future, this work will be extended to include low-level pixel features and to find semantic rules in the video to enhance the overall performance of the system.

References

- [1] Benezeth.Y, Jodoin P.M., Saligrama V, “Abnormal events detection based on spatio-temporal co-occurrences,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2458–2465, 2009. [Article \(CrossRef Link\)](#).
- [2] Chen L, Yuan Cao and Ronghua Ji, “Automatic Incident Detection Algorithm Based on Support Vector Machine,” in *Proc. of Sixth IEEE International Conference on Natural Computation*, vol. 2, pp. 864-866, 2010. [Article \(CrossRef Link\)](#).
- [3] Dong N, Zhen Jia, Jie Shao, “Traffic Abnormality Detection through Directional MotionBehaviour Map,” in *Proc. of Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 80–84, 2010. [Article \(CrossRef Link\)](#).
- [4] Galata A, Johnson N., Hogg D.C., “Learning variable-length Markov Models of behaviour,” *Computer Vision and Image Understanding* 81 (3), 398–413 , 2001. [Article \(CrossRef Link\)](#).
- [5] Gilbert. A, Illingworth. J. and Bowden, R, “Action Recognition Using Mined Hierarchical Compound Features,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 33, Issue: 5, pp.883–897, 2011. [Article \(CrossRef Link\)](#).
- [6] Harwood D, Haritaoglu I and Larry S. Davis, “W4: Real-time surveillance of people and their activities,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.22, pp. 809–830, 2000. [Article \(CrossRef Link\)](#).
- [7] Hoppner, F., “*Knowledge discovery from sequential data*,” Ph.D. thesis, Technical University Braunschweig, Germany, 2003. [Article \(CrossRef Link\)](#).
- [8] Hsu W L, Chang-Lung Tsai and Po-Lun Chang, “Automatic Traffic Monitoring Method Based on Cellular Model,” in *Proc. of Fifth IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 640-643, 2009. [Article \(CrossRef Link\)](#).
- [9] Hu W, Wang L and Tan T., “Recent developments in human motion analysis and Pattern Recognition,” in *Elsevier Journal*, vol. 36, pp. 585–601,2003. [Article \(CrossRef Link\)](#).
- [10] C.R. Jung, L. Hennemann, S.R. Musse, “Event detection using trajectory clustering and 4-d histograms,” *IEEE Transactions on Circuits and Systems for Video Technology*, 18 (11), 1565–1575, 2008. [Article \(CrossRef Link\)](#).
- [11] Kamijo S, Harada M, Sakauchi M, “Incident Detection based on Semantic Hierarchy composed of the Spatio-Temporal MRF model and Statistical Reasoning,” *IEEE International Conference on Man and Cybernetics*, vol.1 ,415-421, 2004. [Article \(CrossRef Link\)](#).
- [12] Ki Y, Lee D, “A traffic accident recording and reporting model at Intersections,” *IEEE Transactions on Intelligent TransportationSystem*, vol.8, issue. 2, pp.188-196, 2007. [Article \(CrossRef Link\)](#).
- [13] Kim. J, Grauman K., “Observe locally, infer globally: A space-time MRF for detecting abnormal activities with incremental updates,” in *Proc. of Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2921–2928, 2009. [Article \(CrossRef Link\)](#).
- [14] Kuettel.D, Breitenstein, M.D.,VanGool, L., “What’s going on? Discovering spatio-temporal dependencies in dynamic scenes,” in *Proc. of Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010. [Article \(CrossRef Link\)](#).
- [15] Leibe. B,Schindler, K.,Cornelis, N., “Coupled Object Detection and Tracking from Static Cameras and Moving Vehicles,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 30, No. 10, pp. 1683-1698 , 2008. [Article \(CrossRef Link\)](#).
- [16] Lili C, Kehuang Li, Jiapin Chen, “Abnormal Event Detection in Traffic Video Surveillance Based on Local Features,” *IEEE Transactions on Image and Signal Processing*, vol.1, pp. 362 – 366, 2011. [Article \(CrossRef Link\)](#).
- [17] Lin, C., Tai, J. and Song, K, “Traffic monitoring based on real-time image tracking,” in *Proceedings of the IEEE International Conference on Robotics & Automation*, vol.2, pp. 2091-2096, 2003. [Article \(CrossRef Link\)](#).
- [18] Loy.C.C,Tao Xiang, Shaogang Gong, “Stream-based active unusual event detection,” in *Proc. of proceedings of Springer on Computer Vision–ACCV 2010*, pp. 161-175, 2011.

- [Article \(CrossRef Link\)](#).
- [19] Loy. C.C, Xiang T., and Gong S., “From Local Temporal Correlation to Global Anomaly Detection,” in *Proc. of ECCV, International Workshop on Machine Learning for Vision-based Motion Analysis*, 2008. [Article \(CrossRef Link\)](#).
- [20] Manku G.S. and Rajeev Motwani, “Approximate Frequency Counts over Data Streams,” in *Proc. of the 28th VLDB Conference, Hong Kong, China*, 2002. [Article \(CrossRef Link\)](#).
- [21] Moerchen, F., “Algorithms for time series knowledge mining,” in *Proc. of the international conference on Knowledge Discovery and Data mining (SIGKDD)*, 668–673, 2006. [Article \(CrossRef Link\)](#).
- [22] Moskovitch, R. Shahar, Y., “Medical temporal-knowledge discovery via temporal abstraction,” in *Proc. of the American Medical Informatics Association (AMIA)*, 2009. [Article \(CrossRef Link\)](#).
- [23] Oliver.N.M, Rosario B., Pentland A.P., “A Bayesian computer vision system for modeling human interactions,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, pp no. 831–843, 2000. [Article \(CrossRef Link\)](#).
- [24] Papapetrou P., Kollios, G., Sclaroff S., “Discovering frequent arrangements of temporal intervals,” in *Proc. of the International Conference on Data Mining (ICDM)*, 2005. [Article \(CrossRef Link\)](#).
- [25] Piciarelli C., Micheloni C. and Foresti G.L, “Trajectory-Based Anomalous Event Detection,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, issue 11, pp. 1544-1554, 2008. [Article \(CrossRef Link\)](#).
- [26] Renxiang Li, Bing Zeng, and Ming L. Liou, “A new Three- step Search Algorithm for Block Motion Estimation,” *IEEE Trans, Circuits and Systems For Video Technology*, Vol 4., no 4, , pp 438-442 , 1994. [Article \(CrossRef Link\)](#).
- [27] SacchiL, Cristiana Larizza, Carlo Combi, “Data mining with Temporal Abstractions: learning rules from time series,” *Data Mining and Knowledge Discovery*, 2007. [Article \(CrossRef Link\)](#).
- [28] Schindler. K., Schindler, K., Cornelis, N., “Coupled Object Detection and Tracking from Static Cameras and Moving Vehicles,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 30, No. 10, pp. 1683-1698, 2008. [Article \(CrossRef Link\)](#).
- [29] Shan kam P, Chee fu A. W., “Discovering temporal patterns for interval-based events,” in *Proc. of the International Conference on Data Warehousing and Knowledge Discovery (DaWaK)*, 2000. [Article \(CrossRef Link\)](#).
- [30] Stauffer. C and Grimson W., “Adaptive background mixture models for real-time tracking,” in *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 246–252, 1999. [Article \(CrossRef Link\)](#).
- [31] Swears, E., Hoogs A., and Perera, A.G.A, “Learning motion patterns in surveillance video using HMM clustering,” in *Proc. of IEEE Visual Motion Computing (Motion) Workshop*, Copper Mountain/Colarado, pp.1-8, 2008. [Article \(CrossRef Link\)](#).
- [32] Veeraraghavan, H., Schrater. P. and Papanikolopoulos N, “Switching Kalman Filter-Based Approach for Tracking and Event Detection at Traffic Intersections,” in *Proc. of IEEE Conference on Control and Automation*, 1167-1172, 2005. [Article \(CrossRef Link\)](#).
- [33] Villafane R, Kien A. Hua, Duc Tran, “Knowledge discovery from series of interval events,” *Journal of Intelligent Information Systems* 15, pp 71–89, 2000. [Article \(CrossRef Link\)](#).
- [34] Wang. X, Ma Xa, Grimson W.E.L., “Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31 (3), 539–555, 2009. [Article \(CrossRef Link\)](#).
- [35] Winarko E, Roddick J. F. “Armada - an algorithm for discovering richer relative temporal association rules from interval-based data,” *Data and Knowledge Engineering* 63, pp 76–90, 2007. [Article \(CrossRef Link\)](#).
- [36] Wu S.Y., Chen Y.L. “Mining non-ambiguous temporal patterns for interval-based events,” *IEEE Transactions on Knowledge and Data Engineering* 19, pp 742–758, 2007. [Article \(CrossRef Link\)](#).

- [37] Xiaokun Li, Porikli, F.M. "A hidden Markov model framework for traffic event detection using video features," in *Proc. of IEEE Conference on Image Processing*, pp 1902-1907, 2004. [Article \(CrossRef Link\)](#)
- [38] Yao B, Wang L, Zhu S., "Learning a scene contextual model for tracking and abnormality detection," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1-8 , 2008. [Article \(CrossRef Link\)](#).
- [39] Zou Y,Guangyi Shi, Hang Shi, "Image sequences based traffic incident detection for signalled intersections using HMM," in *Proc. of international Conference on Hybrid Intelligent Systems*, 2009. [Article \(CrossRef Link\)](#).



Mr.P.M.Ashok Kumar obtained his B.Tech in ECE from JNTU Hyderabad and M.E Computer Science Engg. from Anna University, Chennai, India. Currently he is doing Ph.d in the Department of Electronics Engineering Anna University, India. He authored more than 10 articles in reputed journals, conferences. His main research interests include Image processing, Data mining, Machine learning. He is currently working on Anomaly detection, Video Traffic surveillance, Texture feature extraction, Object tracking features.



Dr. V. Vaidehi has done her BE in ECE from College of Engineering, Guindy, University of Madras, ME Applied Electronics from MIT and Ph.D Electronics Engineering from MIT. She has joined Madras Institute of Technology in 1982 after serving as Scientific Assistant in I.I.Sc, Bangalore.

She has served as Head-Computer Centre, Member Board of Studies, Member - Academic Council, Head- Electronics Engineering, Head- Computer Technology, and Head- Information Technology. Currently, she is the Professor of Electronics Department, Director of AU-KBC Research Center and Chairman of Faculty of Information and Communication Engineering.

She has executed several funded research projects; she published several research papers in reputed international journals and conferences. She has received several awards. Her areas of interest are Networks, Data Mining, and Image processing.