Online Detection of Unusual Events in Videos via Dynamic Sparse Coding

Bin Zhao
School of Computer Science
Carnegie Mellon University
binzhao@cs.cmu.edu

Li Fei-Fei
Computer Science Department
Stanford University
feifeili@cs.stanford.edu

Eric P. Xing
School of Computer Science
Carnegie Mellon University
epxing@cs.cmu.edu

November 23, 2011

Abstract

Real-time unusual event detection in video stream has been a difficult challenge due to the lack of sufficient training information, volatility of the definitions for both normality and abnormality, time constraints, and statistical limitation of the fitness of any parametric models. We propose a fully unsupervised dynamic sparse coding approach for detecting unusual events in videos based on online sparse reconstructibility of query signals from an atomically learned event dictionary, which forms a sparse coding bases. Based on an intuition that usual events in a video are more likely to be reconstructible from an event dictionary, whereas unusual events are not, our algorithm employs a principled convex optimization formulation that allows both a sparse reconstruction code, and an online dictionary to be jointly inferred and updated. Our algorithm is completely unsupervised, making no prior assumptions of what unusual events may look like and the settings of the cameras. The fact that the bases dictionary is updated in an online fashion as the algorithm observes more data, avoids any issues with concept drift. Experimental results on hours of real world surveillance video and several Youtube videos show that the proposed algorithm could reliably locate the unusual events in the video sequence, outperforming the current state-of-the-art methods.

1 Introduction

Recently, there has been growing interests in developing systems to automatically analyze video data. Of the many possible tasks, detecting unusual events from video sequence is of considerable practical importance.

As is often the case, one of the major difficulties in video analysis is the huge amount of data, while it is often true that only a small portion of video contains important information. Consequently, algorithms that could automatically detect unusual events within streaming or archival video would significantly improve the efficiency of video analysis and save valuable human attention for only the most salient contents. It should be noted that the definition of unusual events is rather subjective. In this paper, we define unusual events as those incidences that occur very rarely in the entire video sequence [1, 7, 22, 3, 21, 14].

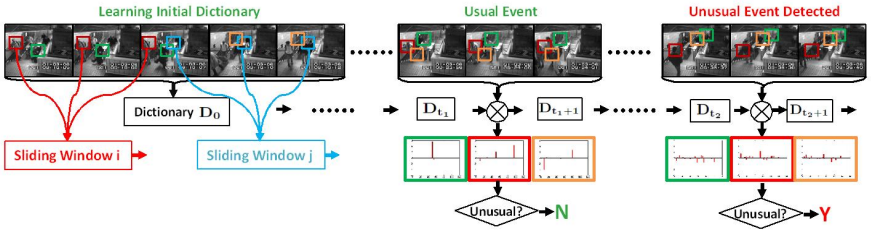


Figure 1: (Best viewed in color) Flowchart of our approach. Given an input video sequence, events are defined using sliding windows (displayed as colored boxes on the video frames). Within each sliding window, spatio-temporal interest points are detected (not shown in the figure), and a dictionary is learned using previously seen video data. For a query event, reconstruction vectors using bases in the dictionary are learned by solving a sparse coding optimization problem. Normality of the query event is then decided using these vectors. Finally, the dictionary is updated with the addition of the query event.

In this work, we provide a framework of using *sparse coding* [12] and *online reconstructibility* to detect unusual events in videos. A query video segment is projected onto a set of sparse coding bases conceptually constituting usual events, which are learned and updated realtime by the algorithm, where the reconstruction error is obtained. An unusual event in a video refers to those segments whose reconstruction errors are significantly higher than the majority of the other (usual event) segments of the video. To our knowledge, we offer the first treatment of unusual event detection in this framework. Compared to previous work that are either model-based [21, 14, 10], or clustering or saliency based [22, 8, 3], our proposed sparse coding framework is built upon a rigorous statistical principle, offering the following advantages: 1) It makes no prior assumptions of what unusual events may look like, hence no need to obtain prior models, templates, knowledge of the clusters; 2) It is completely unsupervised, leveraging only on the assumption that an unusual event is unlikely to occur in the small initial portion of a video; and 3) Our learning algorithm continues to learn and updates its bases dictionary as the algorithm observes more data, avoiding any issues with concept drift.

The rest of this paper is organized as follows. We provide a brief overview of the proposed unusual event detection approach in the remainder of this section. Section 2 provides detailed explanation of the framework, followed by a brief review of previous

works on event detection and sparse coding in Section 3. Section 4 demonstrates the effectiveness of the proposed algorithm using hours of real world surveillance video collected at a subway station and Youtube videos, followed by conclusions in Section 5.

1.1 Overview of Our Approach

Figure 1 provides a flowchart of the proposed unusual event detection approach. Specifically, given a video sequence, the proposed method employs a sliding window along both the spatial and temporal axes to define an event. As the sliding window scans along the spatial and temporal axes, the video is broken into a set of events, each represented by a group of spatio-temporal cuboids. The task of unusual event detection is therefore formulated as detecting unusual group of cuboids residing in the same sliding window. A dictionary is first learnt from the video using sparse coding and later updated in an online fashion as more data become available. Given the learned dictionary, a reconstruction weight vector is learned for each query event and a normality measure is computed from the reconstruction vectors. The proposed algorithm only needs to scan through the video once, and online updating of the learned dictionary makes the algorithm capable of handling concept drift in the video sequence. Finally, using sparse coding enables the algorithm to robustly discriminate between truly unusual events and noisy usual events.

2 Sparse Coding for Unusual Event Detection

2.1 Video Representation

The proposed unusual event detection algorithm adopts a representation based on spatio-temporal cuboids (though it should be noted that the proposed approach could be applied over a variety of video descriptors), to detecte salient points within the video and describe the local spatio-temporal patch around the detected interest points. There have been several attempts in detecting spatio-temporal interest points in video sequences [11, 5, 2]. Here, we adopt the spatio-temporal interest points detected using the method in [5], and describe each detected interest point with histogram of gradient (HoG) and histogram of optical flow (HoF). Figure 2 provides several frames from the video data used in this paper and the detected spatio-temporal interest points within these frames.

Figure 2: Example spatio-temporal interest points detected with the method in [5].

2.2 The Proposed Method

Given a video sequence, the proposed approach employs a sliding window along both the spatial and temporal axes to define an event. Consequently, as a video is represented as a set of cuboids, those cuboids residing in a sliding window define an event. As the sliding window scans along the spatial and temporal axes, the video is broken into a set of events, each represented by a group of spatio-temporal cuboids. Specifically, the video is represented as $\mathbf{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_m\}$, with each event $\mathbf{X}_i$ composed of a group of cuboids, i.e., $\mathbf{X}_i = \{\mathbf{X}_i^1, \ldots, \mathbf{X}_i^{n_i}\}$, where n_i is the total number of cuboids within the sliding window.

2.2.1 A Sparse Coding Formulation

In this work, detecting unusual events in video is formulated as a sparse coding problem. The basic idea for our approach is to represent the knowledge of usual events using the learned dictionary $\mathbf{D}$, whose columns are bases for reconstructing signals. Different from conventional settings of sparse coding, where the input signal is a vector, the input signal in unusual event detection is an event, composed of a group of cuboids $\mathbf{X}_i = \{\mathbf{X}_i^1, \ldots, \mathbf{X}_i^{n_i}\}$. Therefore, the basic unit of input signal is no longer a vector, but instead a group of vectors, with both spatial and temporal location information. In addition to sparsity of the reconstruction weight vectors, we also need to consider the relationships between these weight vectors imposed by the neighborhood structure of cuboids that define the event.

Given dictionary $\mathbf{D}$ (details about learning $\mathbf{D}$ will be provided later in this section), we define the following objective function that measures the normality of an event $\mathbf{X}_i = \{\mathbf{X}_i^1, \ldots, \mathbf{X}_i^{n_i}\}$ and a specific choice of reconstruction weight vectors $\boldsymbol{\alpha}_i = \{\boldsymbol{\alpha}_i^1, \ldots, \boldsymbol{\alpha}_i^{n_i}\}$:

$$J(\mathbf{X}_i, \boldsymbol{\alpha}_i, \mathbf{D}) = \frac{1}{2n_i} \sum_{j=1}^{n_i} ||\mathbf{X}_i^j - \mathbf{D}\boldsymbol{\alpha}_i^j||_2^2 + \frac{\lambda_1}{n_i} \sum_{j=1}^{n_i} ||\boldsymbol{\alpha}^j||_1 + \frac{\lambda_2}{2n_i^2} \sum_{j,k} \mathbf{W}_{jk} ||\boldsymbol{\alpha}_i^j - \boldsymbol{\alpha}_i^k||_2^2 \tag{1}$$

where subscripts j and k run through $\{1, \ldots, n_i\}$ and λ_1, λ_2 are regularization parameters. We now discuss in details of each term in Eq. (1).

Reconstruction error. The first term in Eq. (1) is the reconstruction error. For a usual event, this term should be small, due to the assumption that the learned dictionary represents knowledge in the previously seen video data. A small reconstruction error means the information within the newly observed event $\mathbf{X}_i$ has appeared in early part of the video, which agrees with our definition of usual events.

Sparsity regularization. The second term is the sparsity regularization. Enforcing sparsity for reconstructing usual events is necessary due to the fact that dictionary $\mathbf{D}$ is learned to maximize the sparsity[1] of reconstruction vectors for usual events in the video. On the other hand, for unusual events, although it is possible that a fairly small reconstruction error could be achieved, we would expect using a large amount of video fragments for this reconstruction, resulting in a dense reconstruction weight vector. Figure 3 presents the reconstruction weight vectors for 2 events in the video:

[1] In this paper, we define sparsity as the number of zero elements in a vector.

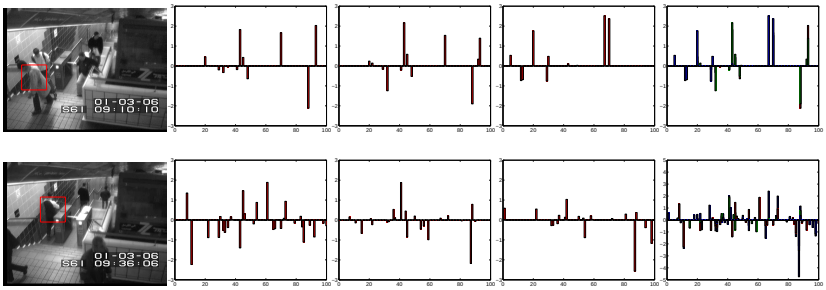


Figure 3: First row: usual event (leaving subway exit); second row: unusual event (entering subway exit). From left to right: example frame and sliding window, reconstruction vectors for 3 cuboids, plot all 3 reconstruction vectors on the same figure.

the first event is usual, and the second is unusual. Results in Figure 3 show that the reconstruction vectors for usual event are sparse, while the ones for unusual event are dense.

Smoothness regularization. The third term is the smoothness regularization, where $\mathbf{W} \in \mathbb{R}^{n_1 \times n_1}$ is the adjacency matrix of $\{\mathbf{X}_i^1, \ldots, \mathbf{X}_i^{n_i}\}$, with large value corresponding to neighboring cuboids and small value corresponding to far apart cuboids. This regularization is based on the fact that similar motions at neighboring patches are more likely to be involved in a usual event. Consequently, it should be of higher probability for similar reconstruction weight vectors being assigned to neighboring cuboids in a usual event. The adjacency matrix $\mathbf{W}$ adopted in this paper is the Gaussian RBF kernel function:

$$W_{jk} = \exp\left[-\frac{||x_j - x_k||^2}{2\sigma^2} - \frac{||y_j - y_k||^2}{2\sigma^2} - \frac{||t_j - t_k||^2}{2\tau^2}\right] \tag{2}$$

where (x_j, y_j) and t_j are spatial and temporal locations of the jth cuboid, σ and τ are variances of the Gaussian function. In the last column of Figure 3, where all 3 reconstruction vectors are plotted on the same image, usual event shows a significant amount of overlap, while the reconstruction vectors for unusual event becomes even denser.

In summary, our sparse coding scheme presented above encapsulates the following intuitions for what we would think of usual and unusual events. Given a dictionary of bases corresponding to usual events, a usual event should be reconstructible from a small number of such bases, in a way that the reconstruction weights change smoothly over space/time across actions in such events. On the other hand, an unusual event is either not reconstructible from the dictionary of usual events with small error, or, even if it is reconstructible, it would necessarily build on a combination of a large number of bases in the dictionary, and possibly in a temporal-spatially non-smooth fashion. Crucial to this technique, is the ability to learn a good dictionary of bases representing usual events, and being able to update the dictionary online to adapt to changing content of the video, which we discuss in detail in next section.

2.2.2 Optimization

The objective function $J(\mathbf{X}_i, \boldsymbol{\alpha}_i, \mathbf{D})$ of Eq. (1) measures the normality of event $\mathbf{X}_i$ with any reconstruction weight vector $\boldsymbol{\alpha}_i$ and any dictionary $\mathbf{D}$. The lower J is, the more likely an event $\mathbf{X}_i$ is normal. As both $\boldsymbol{\alpha}_i$ and $\mathbf{D}$ are latent variables introduced in the formulation, to properly measure the normality of an event $\mathbf{X}_i$, we need to adopt the optimal weight vector $\boldsymbol{\alpha}_i^*$ and dictionary $\mathbf{D}^*$ which minimize the objective function for the given event $\mathbf{X}_i$. Specifically, assume there are m events in the video defined using the sliding window, i.e., $\mathbf{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_m\}$, the optimal reconstruction weight vector $\boldsymbol{\alpha}_i^*$ and dictionary $\mathbf{D}^*$ are learned by solving the following optimization problem

$$(\boldsymbol{\alpha}_1^*, \ldots, \boldsymbol{\alpha}_m^*, \mathbf{D}^*) = \underset{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_m, \mathbf{D}}{\arg\min} \sum_{i=1}^{m} J(\mathbf{X}_i, \boldsymbol{\alpha}_i, \mathbf{D}) \tag{3}$$

subject to proper constraints discussed later. A close look into the above optimization problem reveals that the problem is convex with respect to the coefficients $\boldsymbol{\alpha} = \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_m\}$ of the sparse decomposition when the dictionary $\mathbf{D}$ is fixed, and also convex with respect to $\mathbf{D}$ when $\boldsymbol{\alpha}$ is fixed. However, it is not jointly convex with respect to $\mathbf{D}$ and $\boldsymbol{\alpha}$. A natural solution is to alternate between these two variables, minimizing one while clamping the other. We note that this alternating optimization algorithm converges to local optimum. With the learned dictionary $\mathbf{D}^*$, given a newly observed event $\mathbf{X}'$, the algorithm learns the optimal reconstruction weight vector $\boldsymbol{\alpha}'$ for this event. Consequently, $J(\mathbf{X}', \boldsymbol{\alpha}', \mathbf{D}^*)$ measures the normality of event $\mathbf{X}'$. An event $\mathbf{X}'$ is detected as unusual if its corresponding $J(\mathbf{X}', \boldsymbol{\alpha}', \mathbf{D}^*)$ is larger than certain threshold.

Learning Reconstruction Weight Vector $(\boldsymbol{\alpha})$ with Fixed $\mathbf{D}$. With dictionary $\mathbf{D}$ fixed, reconstruction weight vectors for different events are independent. Therefore, they could be optimized independently. Specifically, for event $\mathbf{X}_i = \{\mathbf{X}_i^1, \ldots, \mathbf{X}_i^{n_i}\}$, the corresponding optimization problem is as follows

$$\min_{\boldsymbol{\alpha}_i^1, \ldots, \boldsymbol{\alpha}_i^{n_i}} \quad \frac{1}{2n_i} \sum_j ||\mathbf{X}_i^j - \mathbf{D}\boldsymbol{\alpha}_i^j||_2^2 + \frac{\lambda_1}{n_i} \sum_j ||\boldsymbol{\alpha}_i^j||_1 + \frac{\lambda_2}{2n_i^2} \sum_{j,k} \mathbf{W}_{jk} ||\boldsymbol{\alpha}_i^j - \boldsymbol{\alpha}_i^k||_2^2 \tag{4}$$

Except for the second term, both two other terms in the objective function are convex quadratic functions of $\boldsymbol{\alpha}_i$. For the above L_1 regularized convex function, the objective is not continuously differentiable. Consequently, the most straightforward gradient-based methods are difficult to apply [12]. Various approaches have been proposed to solve this problem: generic QP solvers (e.g., CVX), interior point method [4], a modification of least angle regression (LARS) [6] and grafting [19]. In this paper, we adopt the feature-sign search algorithm introduced in [12] to solve the above L_1 regularized optimization method.

Learning Dictionary $(\mathbf{D})$ with Fixed $\boldsymbol{\alpha}$. With fixed coefficients $\boldsymbol{\alpha}$, the optimization problem for dictionary $\mathbf{D}$ is as follows

$$\min_{\mathbf{D}} \quad \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1,\ldots,n_i} ||\mathbf{X}_i^j - \mathbf{D}\boldsymbol{\alpha}_i^j||_2^2 \tag{5}$$

$$s.t. \quad \mathbf{D} \in \mathbb{R}^{d\times k} \tag{6}$$

$$\forall j = 1,\ldots,k,\ \mathbf{d}_j^T\mathbf{d}_j \leq 1 \tag{7}$$

The constraint in (7) is introduced to prevent terms in $\mathbf{D}$ from being arbitrarily large, which would result in arbitrarily small values of $\boldsymbol{\alpha}$ [12]. The above optimization problem is a least squares problem with quadratic constraints. In this work, we solve this problem using Lagrange dual.

2.3 Online Dictionary Update

As we stated in the Introduction, one contribution of our work is to automatically learn the video dictionary and perform ongoing learning as we continue to observe the sequence. Unlike previous work where a model for usual events is first learned using training data [3, 10, 1], our fully unsupervised framework can be much more practical in real-world scenarios.

Specifically, the above formulation needs initial training data to learn the dictionary. In video surveillance, it is often challenging to obtain such suitable training data. Even if we were provided with a set of training data, we postulate that the bases dictionary learned from the training data is not necessarily optimal for detecting unusual events in new query videos. We therefore propose an online dictionary learning algorithm in this section that requires no training data other than the video sequence itself. Our idea is to first learn an initial dictionary using an initial portion of the video, and update this learned dictionary using each newly observed event.

Assume the algorithm has observed t-th event in the video, the optimal dictionary is the solution of the following optimization problem

$$\min_{\mathbf{D}\in\mathcal{C}} \frac{1}{2}\sum_{i=1}^{t}\sum_{j=1,\ldots,n_i} ||\mathbf{X}_i^j - \mathbf{D}\boldsymbol{\alpha}_i^j||_2^2 \tag{8}$$

where $\mathcal{C} = \{\mathbf{D} \in \mathbb{R}^{d\times k} :\ \mathbf{d}_j^T\mathbf{d}_j \leq 1,\ \forall j = 1,\ldots,k\}$. Ideally, to solve this problem, we would need all t events $\{\mathbf{X}_1,\ldots,\mathbf{X}_t\}$. However, storing these events requires huge space and solving the optimization problem from scratch is time consuming. One possible solution is projected first order stochastic gradient descent, consisting of the following update [15]:

$$\mathbf{D}_t = \Pi_{\mathcal{C}}\left[\mathbf{D}_{t-1} - \frac{\eta}{t}\nabla_{\mathbf{D}} l(\mathbf{X}_t, \mathbf{D}_{t-1})\right] \tag{9}$$

where $l(\mathbf{X}_t, \mathbf{D}_{t-1}) = \frac{1}{2}\sum_{j=1,\ldots,n_t} ||\mathbf{X}_t^j - \mathbf{D}_{t-1}\boldsymbol{\alpha}_t^j||_2^2$, η is the learning rate, $\Pi_{\mathcal{C}}$ is the orthogonal projection onto $\mathcal{C}$. This method has shown satisfactory performance, when

a good learning rate η is selected. However, the introduction of η would further complicates our event detection method by increasing the difficulty of picking proper parameters. Therefore, in our proposed event detection framework, we follow the online dictionary update algorithm proposed in [15] as shown in Algorithm 1. Specifically, we define the following two matrices to store the corresponding variables computed in previous steps:

$$\mathbf{A}_{t-1} = [\mathbf{a}_1, \ldots, \mathbf{a}_k] = \sum_{i=1}^{t-1} \sum_{j=1}^{n_i} \boldsymbol{\alpha}_i^j \boldsymbol{\alpha}_i^{j^T} \in \mathbb{R}^{k \times k} \tag{10}$$

$$\mathbf{B}_{t-1} = [\mathbf{b}_1, \ldots, \mathbf{b}_k] = \sum_{i=1}^{t-1} \sum_{j=1}^{n_i} \mathbf{x}_i^j \boldsymbol{\alpha}_i^{j^T} \in \mathbb{R}^{d \times k} \tag{11}$$

After observing the t-th event $\mathbf{X}_t^1, \ldots, \mathbf{X}_t^{n_t}$, and computing its corresponding reconstruction vectors $\boldsymbol{\alpha}_t^1, \ldots, \boldsymbol{\alpha}_t^{n_t}$, we could update these two matrices as following:

$$\mathbf{A}_t = \mathbf{A}_{t-1} + \sum_{j=1}^{n_t} \boldsymbol{\alpha}_t^j \boldsymbol{\alpha}_t^{j^T} \tag{12}$$

$$\mathbf{B}_t = \mathbf{B}_{t-1} + \sum_{j=1}^{n_t} \mathbf{x}_t^j \boldsymbol{\alpha}_t^{j^T} \tag{13}$$

Then the optimization problem for $\mathbf{D}_t$ could be equivalently formulated as follows:

$$\mathbf{D}_t = \arg\min_{\mathbf{D} \in \mathcal{C}} \frac{1}{2} \sum_{i=1}^{t} \sum_{j=1}^{n_i} ||\mathbf{X}_i^j - \mathbf{D}\boldsymbol{\alpha}_i^j||_2^2 = \arg\min_{\mathbf{D} \in \mathcal{C}} \frac{1}{2} \mathrm{Tr}(\mathbf{D}^T \mathbf{D} \mathbf{A}_t) - \mathrm{Tr}(\mathbf{D}^T \mathbf{B}_t) \tag{14}$$

Since the objective function for $\mathbf{D}_t$ is close to that for $\mathbf{D}_{t-1}$, $\mathbf{D}_t$ could be obtained efficiently using $\mathbf{D}_{t-1}$ as warm start. In Algorithm 1, each column $\mathbf{d}_j$ in $\mathbf{D}$ is optimized separately while keeping the other ones fixed, and then projected to the constraint $\mathbf{d}_j^T \mathbf{d}_j \leq 1$. Using $\mathbf{D}_{t-1}$ as warm start, this process usually converges in a few steps.

2.4 Unusual Event Detection

As briefly mentioned in previous section, given a newly observed event $\mathbf{X}'$ and the current dictionary $\mathbf{D}^*$, the proposed algorithm learns the corresponding optimal reconstruction weight vector $\boldsymbol{\alpha}'$. $\mathbf{X}'$ is detected as an unusual event if the following criterion is satisfied

$$J(\mathbf{X}', \boldsymbol{\alpha}', \mathbf{D}^*) > \hat{\epsilon} \tag{17}$$

where $\hat{\epsilon}$ is a user defined threshold that controls the sensitivity of the algorithm to unusual events. Combining everything together, Algorithm 2 presents our unusual event detection method.

Algorithm 1 Online dictionary update

Input: $\mathbf{D}_{t-1} = [\mathbf{d}_i, \ldots, \mathbf{d}_k] \in \mathbb{R}^{d \times k}$, $\mathbf{A}_t = [\mathbf{a}_1, \ldots, \mathbf{a}_k] \in \mathbb{R}^{k \times k}$ and $\mathbf{B}_t = [\mathbf{b}_1, \ldots, \mathbf{b}_k] \in \mathbb{R}^{d \times k}$

repeat

for $j = 1$ to k **do**

Update $\mathbf{d}_j$ as following

$$\mathbf{u}_j = \frac{\mathbf{b}_j - \mathbf{D}\boldsymbol{\alpha}_j}{\mathbf{A}_{jj}} + \mathbf{d}_j \tag{15}$$

$$\mathbf{d}_j = \frac{\mathbf{u}_j}{\max(||\mathbf{u}_j||_2, 1)} \tag{16}$$

end for

until convergence

Algorithm 2 Unusual event detection using sparse coding

Input: video data, learning rate η, threshold $\hat{\epsilon}$

Learn initial dictionary using first N frames in video

repeat

Use sliding window to obtain event $\mathbf{X}_t$

Learn optimal reconstruction vectors $\boldsymbol{\alpha}_t$ for event $\mathbf{X}_t$ by solving Eq. (4) with $\mathbf{D} = \mathbf{D}_{t-1}$

if $J(\mathbf{X}_t, \boldsymbol{\alpha}_t, \mathbf{D}_{t-1}) > \hat{\epsilon}$ **then**

Fire alarm for event $\mathbf{X}_t$

end if

Update dictionary $\mathbf{D}$ with Algorithm 1

until reach the end of video

3 Related Works

Several attempts have been proposed in the literature on unsupervised unusual event detection in videos [1, 7, 22, 3, 21, 14]. Specifically, [7, 20, 9] studies the problem using tracking trajectories. However, even with the recent advances in tracking techniques, reliably tracking an object in crowded video is still a very challenging research problem. Clustering methods [22, 8] have also been applied to detect unusual events, where the detection is carried out by finding spatially isolated clusters. The fact that these methods only run in batch mode severely limits their applicability. [1] proposes a simple yet effective approach that measures typical flow directions and speeds on a grid in the video frame to detect unusual events. This algorithm is good for detecting simple events such as moving in the wrong direction. [3] proposes a database indexing algorithm, where the problem is formulated as composing the new observed video data using spatio-temporal patches extracted from previous visual examples. Regions in the query video that can be composed using large contiguous chunks of data from the example database are considered normal. Although this algorithm shows good per-

formance in discriminating complex motions, it faces scalability issues as its time and memory complexity is linear in the size of the example database. Finally, [10] utilizes a space-time Markov random field to detect unusual events, where an MRF model is built for usual events and those events that could not be described with the learned model is considered as unusual.

On the other hand, sparse coding [12] has shown promising results in finding succinct representations of stimuli. For example, applying sparse coding algorithm to natural images has been shown to be capable of learning the bases resembling the receptive fields of neurons in the visual cortex [17, 18]. Moreover, sparse coding has been shown to produce localized bases when applied to other natural stimuli such as video and speech [16, 13]. Different from conventional sparse coding, where the bases in dictionary are fixed after training, the dictionary in our dynamic sparse coding framework is updated online to adapt to changing content of the video.

4 Experiments

In this section, we show the empirical performance of the proposed unusual event detection algorithm, both qualitatively and quantitatively.

4.1 Subway Surveillance Video

The first 2 data sets are video sequences taken from surveillance camera at a subway station, with one camera monitoring the exit and the other monitoring the entrance. In both videos, there are roughly 10 people walking around in a typical frame, with a frame size of 512×384. The videos are provided by courtesy of Adam et al. [1] and we compare quantitatively the detection results of our approach against the method in [10].

4.1.1 Subway Exit

The subway exit surveillance video is 43 minutes long with 64901 frames in total. To ensure a fair qualitative comparison, we follow the same definition of unusual events used in [10] for the same data set, though it should be noted that the definition of unusual events is rather subjective. Specifically, 3 types of unusual events are defined in the subway exit video: (a) walking in the wrong direction *(WD)*; (b) loitering near the exit *(LT)* and (c) misc, including suddenly stop and look around, janitor cleaning the wall, someone gets off the train and gets on again very soon. Totally, 19 unusual events are defined as ground truth.

We use a sliding window of size 80×80 pixels along x and y axes, and 40 frames along t axis in our approach. The fist 5 minutes of the video, same as in [10], is used to build initial dictionary. Before providing the unusual event detection results, we first show the dictionary learned using our approach in Figure 4. Specifically, Figure 4 visualizes randomly selected 10 bases in the learned dictionary (the size of the learned dictionary is 100). We observe that the learned bases of the dictionary reflects our intuition about what common and usual events are in this video: people walking towards the camera (exiting the subway), walking to the left or right, train leaving station, etc.

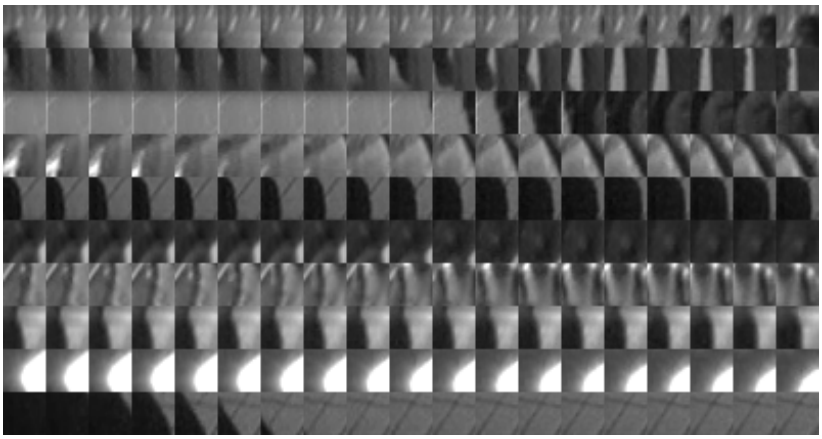


Figure 4: Dictionary learned using our approach for subway exit surveillance video. Each row in the figure corresponds to a basis in the dictionary. Typical activities in this dictionary include: walking to the left or right, walking towards the camera, train leaving station, etc.

	WD	LT	MISC	Total	FA
GT	9	3	7	19	0
ST-MRF [10]	9	3	7	19	3
Ours	9	3	7	19	2

Table 1: Comparison of unusual event detection rate and false alarm rate on subway exit surveillance data: GT stands for ground truth annotation; ST-MRF refers to the method proposed in [10].

Table 1 provides quantitative results on unusual event detection accuracy and false alarm rate. We follow the same annotation used in [10], where a frame range is defined for each unusual event. For evaluation, once the algorithm detects at least one frame in the annotated range, the detection is counted as correct. On the other hand, false alarm is also measured in the same way: at least one frame is fired outside the annotated range, then it is counted as false alarm[2]. Figure 5 shows the detection results on the subway exit data, including the correct detections, and false alarms. Our method can detect an unusual event even within a crowded scene with occlusions (e.g., Figure 5(d)). Also, we can see that our method captures the unusual event caused by fine scale irregular motion (e.g., Figure 5(k)), or abnormal event resulted by irregular temporal ordering of activities (e.g., Figure 5(j)). We also illustrate two false alarms detected by our algorithm (Figure 5(o) & (p)). Curiously, looking closer into the video, these two events are indeed "unusual": Figure 5(o) is due to the first appearance of a child, and Figure 5(p) is due to the action of a man stopping near the exit and looking back. They

[2]There are other evaluation metrics which could also be reasonable. We use this evaluation metric to be able to compare with [10].

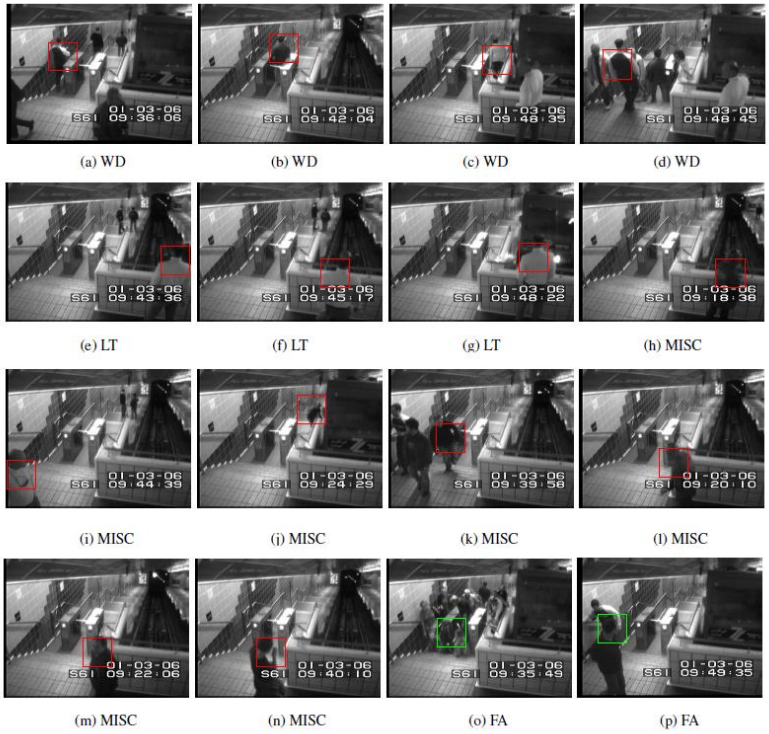


Figure 5: Unusual event detection in the subway exit surveillance video. WD: wrong direction; LT: loitering; MISC: misc; FA: false alarm. The rectangle on the figure marks the sliding window that results in the detection of unusual events. False alarms are marked using green sub-window.

are missed in ground truth annotations, hence labeled as FA in evaluation.

4.1.2 Subway Entrance

The subway entrance video is 1 hour 36 minutes long with 144249 frames in total. 66 unusual events are defined, covering 5 different types: (a) walking in the wrong direction *(WD)*; (b) no payment *(NP)*; (c) loitering *(LT)*; (d) irregular interactions between people *(II)* and (e) misc, including sudden stop, running fast.

We use the same sliding window as in subway exit video, and the fist 15 minutes for training as in [10]. Figure 6 shows the dictionary learned by our approach, where we randomly select 12 bases out of 200 in the dictionary. This dictionary shows activities such as people walking to the left or right, walking away from the camera, which are usual events in this video. Quantitative comparison results with [10] are shown in Table 2, where our approach achieves higher detection rate and fewer false alarms. Moreover, as reported in [10], the approach in [1] fails to detect abnormal activities

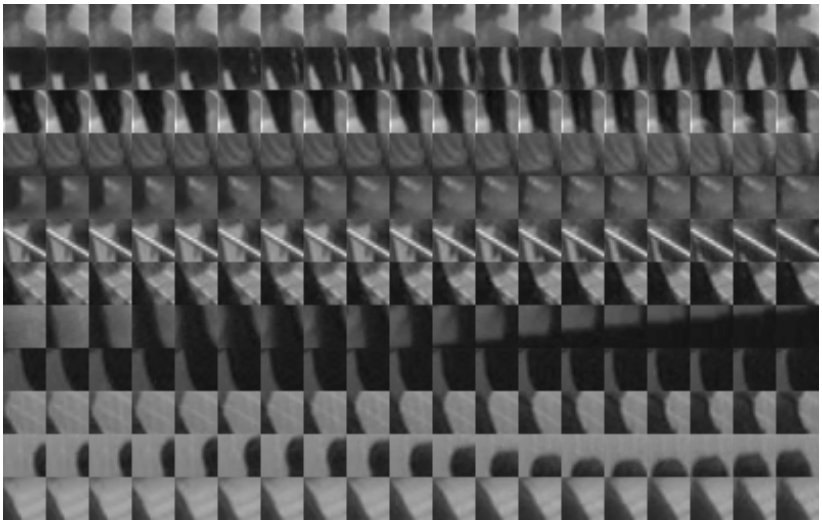


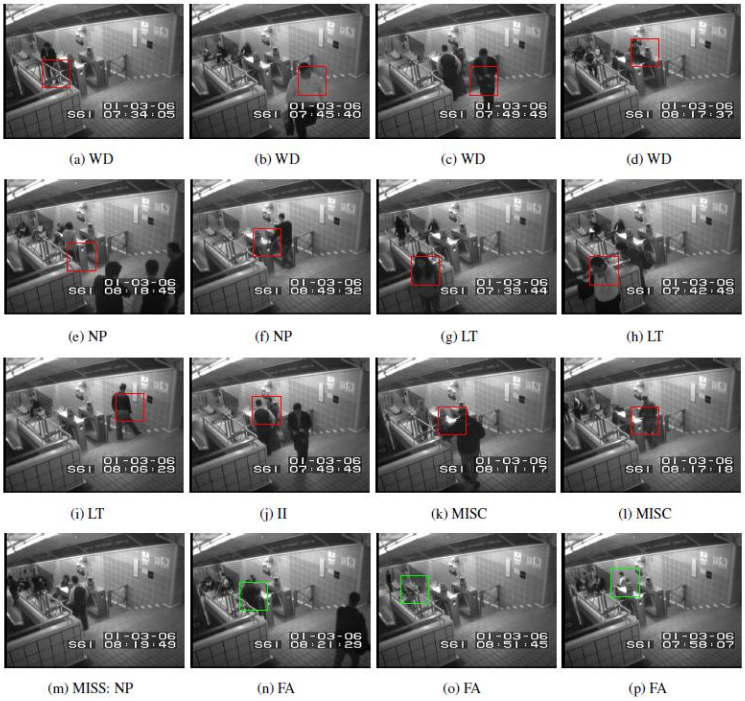
Figure 6: Dictionary learned using our approach for subway entrance surveillance data. Each row in the figure corresponds to a basis in the dictionary. Typical activities in this dictionary include: walking to the left or right, walking away from the camera, etc.

with irregular temporal orderings, such as Figure 5(j), people getting off the train and getting back quickly. Also, the method in [1] results in an order magnitude more false alarms than [10]. Moreover, the clustering-based method [22] cannot detect events happening at a fine local scale, such as Figure 7(e) & (f). Therefore, while achieving slightly better qualitative performance than [10], our method also clearly outperforms the methods in [1] and [22] by a large margin.

Figure 7 displays unusual events detected using our approach. Our method not only detects abnormalities in a fine scale (e.g., Figure 7(e) & (f)), but also unusual events caused by irregular interactions between people (e.g., Figure 7(j)). Moreover, we can see that our method could correctly detect abnormal activities where both usual and unusual events occur in the same frame (e.g., Figure 7(g)).

4.1.3 Analysis Experiment: Online Update of the Learned Dictionary

In our approach, the learned dictionary is updated after observing each new event using projected stochastic gradient descent. In this section, we compare the results of our algorithm with the method using initially learned dictionary throughout the entire video sequence. Specifically, in the subway exit surveillance data, the second method learns an initial dictionary using the first 5 minutes of video and keep this dictionary fixed in the entire detection process. Similarly, in the subway entrance video data, the second method employs the fixed dictionary learned from first 15 minutes of video. Table 3 compares the detection accuracy and false alarms of the two methods. The method using fixed dictionary generally gives more false alarms than our approach. This result underscores our contribution in developing an online learning framework to update the



(a) WD (b) WD (c) WD (d) WD

(e) NP (f) NP (g) LT (h) LT

(i) LT (j) II (k) MISC (l) MISC

(m) MISS: NP (n) FA (o) FA (p) FA

Figure 7: Unusual event detection in the subway entrance surveillance video. WD: wrong direction; NP: no payment; LT: loitering; II: irregular interactions; MISC: misc; MISS: missed unusual event; FA: false alarm.

bases dictionary. Without the online updates, the Fixed Dictionary method shows the inability for adapting to the changing contents of the video, resulting in a much greater error rate.

4.2 Unusual Event Detection in Youtube Videos

The above experiment has demonstrated our model's superiority in unusual event detection in surveillance videos, where the camera is fixed and the environment is relatively controlled. But our framework is a general approach that makes no assumptions of the cameras, the types of environment, or the contents of the video. In this section, we apply our method to a number of videos "in the wild", highlighting its application to a wide range of data. We downloaded a number of videos from YouTube. As Figure 8 shows, these videos have very different camera motion (rotation, zoom in/out, fast tracking, slow motion, etc.), contains different categories of targets (human, vehicles, animals, etc.) and covers a wide variety of activities and environmental conditions (indoor, outdoor).

For each of the 8 Youtube videos, we use approximately the first 1/5 of video data

	WD	NP	LT	II	MISC	Total	FA
GT	26	13	14	4	9	66	0
ST-MRF [10]	24	8	13	4	8	57	6
Ours	25	9	14	4	8	60	5

Table 2: Comparison of unusual event detection rate and false alarm rate on subway entrance surveillance data.

	Correct Detection	False Alarm
Fixed $\mathbf{D}$	17/54	8/12
Ours	19/60	2/5

Table 3: Comparison of unusual event detection rate and false alarm rate: online updating dictionary vs. fixed dictionary. The number before '/' is for subway exit surveillance data, while the number after '/' is for entrance surveillance data.

to learn an initial dictionary, and display detected unusual events in Figure 8. With no model assumptions of what is unusual, no need for templates, no supervision or training, our method could correctly detect abnormal activities in these real world low-quality videos.

5 Conclusions

We propose an unsupervised algorithm to automatically detect unusual events from a video sequence. A query video segment is projected onto a set of sparse coding bases learned by the algorithm, to obtain the reconstruction vectors. Normality is then computed based on these reconstruction vectors. Moreover, the sparse coding bases are updated dynamically in an online fashion, to capture possible concept drift in video contents. Experimental results on two real world surveillance videos and several Youtube videos demonstrate the effectiveness of the proposed algorithm.

Acknowledgements

E. P. Xing is supported by NSF IIS-0713379, DBI-0546594, Career Award, ONR N000140910758, DARPA NBCH1080007 and Alfred P. Sloan Foundation. L. F-F is partially supported by an NSF CAREER grant (IIS-0845230), an ONR MURI grant, and the DARPA Mind's Eye program. We thank Juan Carlos Niebles and anonymous reviewers for helpful comments.

References

[1] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *PAMI*, 30:555 – 560, 2008.

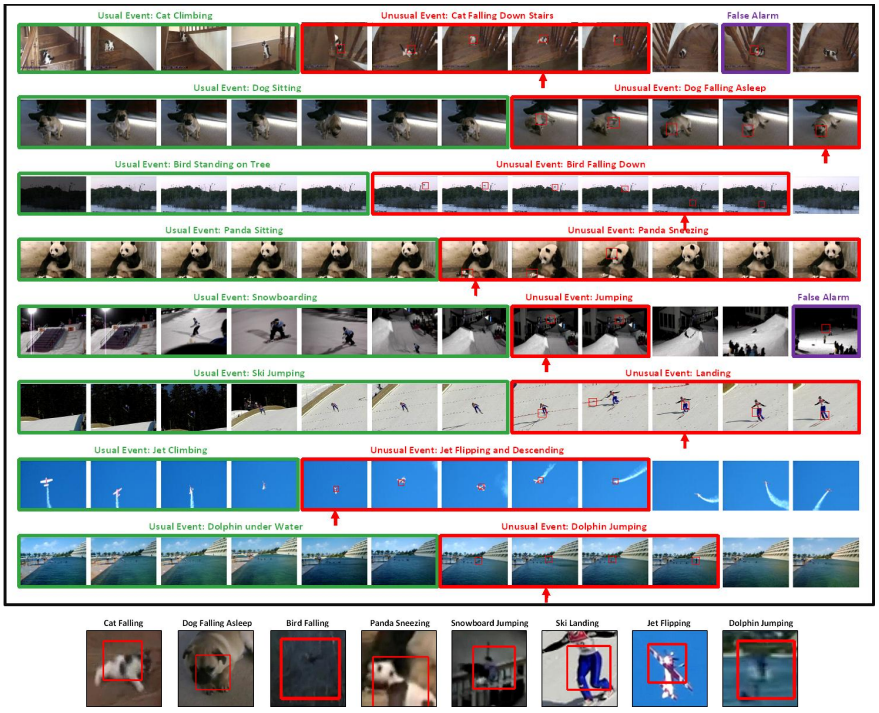


Figure 8: Unusual event detection results on 8 Youtube Videos. Frames of usual events, detected unusual events and false alarms are shown in the first 8 rows. For frames involving unusual events, red boxes on video frames represent patches that trigger the alarm. The bottom row provides a zoom-in view of those patches, taken from one frame (pointed by red arrows) per video.

[2] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005.

[3] O. Boiman and M. Irani. Detecting irregularities in images and in video. In *ICCV*, 2005.

[4] S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33 – 61, 1998.

[5] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, 2005.

[6] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2), 2004.

[7] A. B. A. Gritai and M. Shah. Learning object motion patterns for anomaly detection and improved object detection. In *CVPR*, 2008.

[8] R. Hamid, A. Johnson, S. Batta, A. Bobick, C. Isbell, and G. Coleman. Detection and explanation of anomalous activities: Representing activities as bags of event n-grams. In *CVPR*, 2005.

[9] W. Hu, X. Xiao, Z. Fu, and D. Xie. A system for learning statistical motion patterns. *PAMI*, 28:1450 – 1464, 2006.

[10] J. Kim and K. Grauman. Observe locally, infer globally: A space-time mrf for detecting abnormal activities with incremental updates. In *CVPR*, 2009.

[11] I. Laptev. On space-time interest points. *IJCV*, 64:107 – 123, 2005.

[12] H. Lee, A. Battle, R. Rajat, and A. Y. Ng. Efficient sparse coding algorithms. In *NIPS*, 2007.

[13] M. Lewicki and T. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12(2), 2000.

[14] J. Li, S. Gong, and T. Xiang. Global behaviour inference using probabilistic latent semantic analysis. In *BMVC*, 2008.

[15] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *ICML*, 2009.

[16] B. Olshausen. Sparse coding of time-varying natural images. *Vision of Vision*, 2(7):103, 2002.

[17] B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607 – 609, 1996.

[18] B. Olshausen and D. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Research*, 37:3311 – 3325, 1997.

[19] S. Perkins and J. Theiler. Online feature selection using grafting. In *ICML*, 2003.

[20] C. Stauffer and E. Grimson. Learning patterns of activity using real-time tracking. *PAMI*, 22:747 – 757, 2000.

[21] X. Wang, X. Ma, and E. Grimson. Unsupervised activity perception by hierarchical bayesian models. In *CVPR*, 2007.

[22] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *CVPR*, 2004.