Video anomaly search in crowded scenes via spatio-temporal motion context

Cong, Yang; Yuan, Junsong; Tang, Yandong

2013

Cong, Y., Yuan, J., & Tang, Y. (2013). Video Anomaly Search in Crowded Scenes via Spatio-Temporal Motion Context. IEEE Transactions on Information Forensics and Security, 8(10), 1590-1599.

https://hdl.handle.net/10356/100311

https://doi.org/10.1109/TIFS.2013.2272243

Video Anomaly Search in Crowded Scenes via Spatio-Temporal Motion Context

Yang Cong, *Member, IEEE*, Junsong Yuan, *Member, IEEE* and Yandong Tang, *Member, IEEE*

***Abstract*—Video anomaly detection plays a critical role for intelligent video surveillance. We present an abnormal video event detection system that considers both spatial and temporal contexts. To characterize the video, we first perform the spatio-temporal video segmentation and then propose a new region-based descriptor called "*Motion Context*", to describe both motion and appearance information of the spatio-temporal segment. For anomaly measurements, we formulate the abnormal event detection as a matching problem, which is more robust than statistic model based methods, especially when the training dataset is of limited size. For each testing spatio-temporal segment, we search for its best match in the training dataset, and determine how normal it is using a dynamic threshold. To speed up the search process, compact random projections are also adopted. Experiments on the benchmark dataset and comparisons with the state-of-the-art methods validate the advantages of our algorithm.**

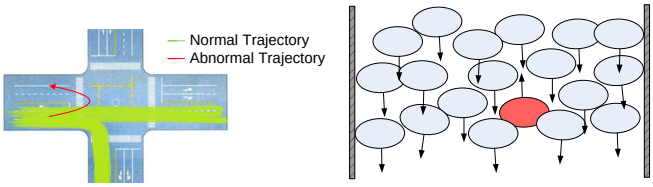
***Index Terms*—Abnormal Event Detection, Video Analysis, Event Recognition, Video Surveillance, Motion, Compact Projection** [1]

I. Introduction

NOWADAYS, a large number of surveillance cameras have been installed due to the decreasing costs of video cameras. Intelligent video surveillance [1] is of great interests in industry applications due to the increasing demand to reduce the manpower of analyzing the large-scale video data. Key technologies have been developed for intelligent surveillance, such as object tracking [2], [3], pedestrian detection [4], gait analysis [5], vehicle template recognition [6], privacy protection [7], face and iris recognition [8], video summarization [9] and crowd counting [10]. In this paper, we focus on video anomaly detection (also named as outlier detection), i.e. detecting the irregular patterns that are different from the regular video events in a given data set [11]–[20], and we intend to build an abnormal event detection system that can work in crowded scenes as well.

Despite many previous work of detecting video anomalies [11]–[20], few of them can work well in crowded scenes, due to the following challenges:

- First, a crowded scene usually contains a large number of moving persons; thus can easily distract the local anomaly detector. It is difficult, even for human beings, to effectively identify all abnormal behaviors in real time.
- Second, whether an event is normal or abnormal usually application and context dependent, thus it is difficult to model the abnormal event. An event may be considered as normal in one scenario while abnormal in another scenario. For real applications, it is desired that we can adaptively define the video anomaly rather than manually do this for each scenario.
- Third, although it is easy to obtain training videos of normal video events, it is difficult to collect sufficient samples of abnormal video events. Such an unbalanced training data brings challenges to build a robust video anomaly detector.



(a) Temporal Abnormal Event (b) Spatial Abnormal Event

Fig. 1. Examples of video anomalies in different scenarios.

We illustrate two spatio-temporal video anomalies in Fig. 1. In Fig. 1(a), as the majority of vehicles follow the green trajectories, the U-turn moving is treated as abnormal. In Fig. 1(b), each ellipse stands for a moving pedestrian, but the behavior of the red one is different from its neighborhood, thus is considered as an abnormal event. To make a video anomaly detection system easy to use, the detection added new subsection to compare the influence of different image patch size using of video anomaly should be adaptive to different scenes.

A. Problem Definition

By considering different application scenarios for abnormal event detection, we define the problem of video anomaly detection as follows. Suppose we are provided a training set $\mathbf{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, where N is the number of training samples; $\mathbf{x}_i \in \mathbb{R}^d$ is the feature vector describing a *normal* training sample (d is the feature dimension), which can be an image patch, a color histogram, a mixture of dynamic texture, or our proposed motion context, etc. Suppose we have a test sample $\mathbf{y} \in \mathbb{R}^d$, our task is to design a function to determine whether $\mathbf{y}$ is normal or abnormal. That is

$$f : \mathbf{y} \mapsto \{normal, abnormal\} \quad (1)$$

Y.Cong is with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, China, 110016 e-mail: congyang81@gmail.com

J. Yuan is with the Department of EEE, Nanyang Technological University, Singapore, 639798 e-mail: jsyuan@ntu.edu.sg

Y. Tang is with the State Key Laboratory of Robotics, Chinese Academy of Science, China, 110016 e-mail: ytang@sia.ac.cn

[1]This work was supported in part by Natural Science Foundation of China (61105013), Nanyang Assistant Professorship SUG M4080134.040 and NTU-JSPS joint project M4080882.040.

There are in general two ways to achieve Eq. (1):

i) Probability model based methods, which fit probability model, e.g. Gaussian Mixture Model (GMM) or Mixture of Probability Principle Component Analysis (MPPCA), using the training data set $\mathbf{D}$, and calculate the posterior probability to detect anomaly:

$$f = \begin{cases} normal & p(\mathbf{y}|\mathbf{D}) \geq \theta \\ abnormal & p(\mathbf{y}|\mathbf{D}) < \theta \end{cases} \quad (2)$$

To fit a good model, these methods usually need sufficient training samples (approximate $O(d^2)$), and the situation gets further deteriorated when using high dimensional features. Unfortunately, the training dataset is usually of limited size and it is not realistic to collect sufficient video anomalies.

ii) Nearest neighbor (NN) based methods, which compare the current testing sample with all the training data:

$$f = \begin{cases} normal & \forall x_i, Dist(\mathbf{y}, \mathbf{x}_i) \leq \varepsilon_i \\ abnormal & otherwise \end{cases} \quad (3)$$

where $Dist(\cdot)$ is a pairwise feature distance and ε_i is a threshold about $\mathbf{x}_i$. Compared with the probability based methods, the advantage is that they can still obtain robust results even if the training data is of limited size. Therefore, we propose to detect video anomaly by retrieving the most similar normal examples in the training dataset. If the retrieved example is similar enough to the query example, the query will be normal; otherwise a video anomaly will be detected.

To detect video anomaly, another important issue is the video event representation. Most of the state-of-the-art methods consider spatio-temporal information and extract features from the local patch. Various types of co-occurrence matrices are often chosen to describe the spatial context information. These methods are inflexible and inefficient for the event representation. We propose to dynamically group the local patches with similar characteristics together to represent the abnormal event. In summary, our contributions mainly lie in three aspects:

i) By considering the spatio-temporal characteristics of video events, we design a new feature to represent video events in crowded scenes, called motion context, which relies on the dynamic patch grouping (DPG).

ii) Based on motion context, we propose a unified approach to detect spatio-temporal abnormal events, and is adaptive to different scenes.

iii) We formulate the problem of abnormal event detection as a retrieval problem, where for each spatio-temporal video segment, we search for the best match in the training dataset and determine how normal it is. Compared with conventional probability based methods, our method can achieve robust results with limited training samples of normal events. To improve the maintain low computational cost, we apply compact projections to perform fast nearest neighbor search.

The remainder of the paper is organized as follows. Sec.II briefly surveys previous works while Sec.III summarizes our method. Then we propose our video representation in Sec.IV, followed by our framework in Sec.V. The experiment results and conclusion are presented in Sec.VI and Sec.VII, respectively.

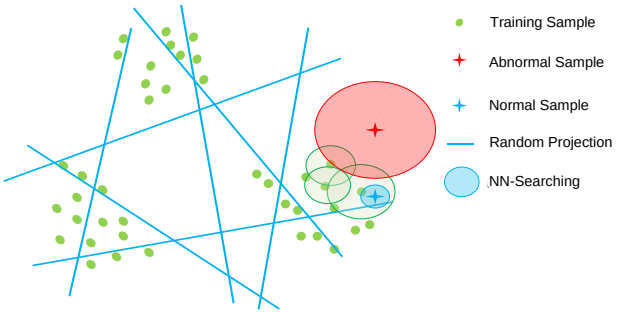


Fig. 2. The illustration of our algorithm. The definition of each symbol is shown in the top-right. The radius of the circle of each training sample corresponds to the dynamic threshold, which varies from one training sample to another; and the radius of each testing sample denotes the nearest neighbor searching radius. Please check the texts for details.

II. Related Work

A detailed review is beyond the scope of this paper, which can be referred to [21], [22]. Depending on the specific applications, abnormal event detection can be categorized into those in the crowded scenes and the un-crowded scenes. For the un-crowded scenario, binary feature based on background models are usually adopted, such as Normalization Cut clustering [11] and 3D spatio-temporal foreground mask feature fusion using Markov random field [14]. There are also some trajectory-based approaches to locate objects by tracking or frame-difference, such as [23], [24], [25], [26], [27], [28], [29] and [30], which can get satisfied results on traffic monitoring, however, may fail on density crowded scenes due to bad trajectories initialization.

For the crowded scenes, most of state-of-the-art methods consider spatio-temporal information and extract motion or gray-level SIFT-like features from local 2D image patches or local 3D video bricks, like Histogram of optical flow, 3D gradient, etc. The co-occurrence matrices are often chosen to describe the context information. For example, Adam et al. [15] use histograms to measure the probability of optical flow in local patch. Thida et al. [31] learn video manifold for abnormality detection. Thide et al. [32] also propose to detect local abnormal event using Laplacian eigenmap. Kratz et al. [33] extract spatio-temporal gradient to fit Gaussian model of each 3D video brick, and then use HMM to detect abnormal events in densely crowded subway. The saliency features are extracted and associated using a Bayesian model to detect surprising (abnormal) events in video [34]. Kim et al. [35] model local optical flow with Mixture of Probability Principle Component Analysis (MPPCA) and enforce consistency by Markov random field. In [36], a graph-based non-linear dimensionality reduction method using motion cues is applied for abnormality detection. Mahadevan et al. [16] model the normal crowd behavior by mixtures of dynamic textures. Mehran et al. [17] present a new way to formulate the abnormal crowd behavior by adopting the social force model [37]. In [38], the authors define a chaotic invariant feature to describe the

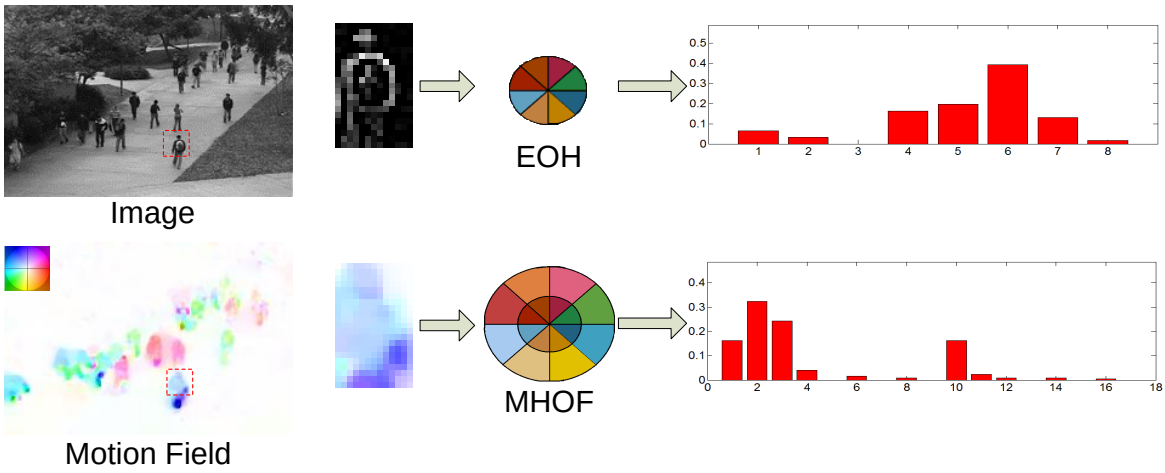


Fig. 3. Basic description unit (2D local patch or 3D local volume). Both motion and appearance information are adopted for each unit. For motion descriptor, the Multi-layer Histogram of Optical Flow (MHOF) is used, and for appearance descriptor, the Edge Orientation Histogram (EOH) is adopted. The meaning of the color of each pixel in the motion field image corresponds to the motion direction and magnitude as shown in the left-top sub-figure.

event. Boiman and Irani [18], [39] extract 3D video bricks and use dynamic programming to infer the anomaly. The sparse representation is used to overcome the problem between event representation using high-dimensional features and statistic model complexity, such as sparse reconstruction cost (SRC) [19] proposed in our previous work, and also [40], [41], which does not address the large-scale dictionary selection problem and cannot handle both Local Abnormal Event (LAE) and Global Abnormal Event (GAE) simultaneously as well.

III. Overview of Our Method

In this paper, we consider abnormal event detection as a retrieval problem, where we search the video event for its best match in the training dataset and determine how normal it is. The general idea is illustrated in Fig. 2. Either normal or abnormal event is described by motion context using a high dimensional feature point. The training samples $\mathbf{x}_i \in \mathbb{R}^d$ all belong to normal (denoted by green points). There are several clusters of green points, as training samples have several patterns; and each cluster contains limited training samples, because we cannot collect enough training data. For each testing sample $\mathbf{y}_j \in \mathbb{R}^d$ denoted by crossing point, whether $\mathbf{y}_j$ is normal or not, is determined by its nearest neighbor set. To speed up the nearest neighbor searching, the compact projection based on random projection is adopted here, as shown in Fig. 2 by blue lines. Then, a dynamic threshold is given to decide whether the current sample is normal or not, depending on the similarity, as shown by different radius in Fig. 2. Blue points with smaller searching radius belong to normal class; in the contrast, red points with larger searching radius are abnormal ones. More details will be discussed later.

IV. Event Representation

In this section, we propose a flexible video representation using dynamic patch grouping. Most of the state-of-the-art event descriptors extract motion or appearance features from local 2D patch or 3D sub-volume [33], [15], [11], [14], then adopt co-occurrence matrix to describe the pairwise spatial relationship. However, these descriptors are inflexible, because it is unrealistic to predefine a suitable patch size, and moreover the fixed spatial structure of co-occurrence matrix may lose crucial information. In [19], we make a further step by defining various types of spatio-temporal basis. Nevertheless, it also needs to predefine the spatio-temporal topology structure of the descriptor, which makes it inflexible in practical applications.

In crowded scenes, different moving objects interconnect with each other frequently. Thus it is of great importance to segment them and recover each own motion. As each event may contain many image patches, which are similar to each other in both spatial and feature space. We use dynamic patch grouping (DPG) to adaptively cluster similar patches and represent each group as motion context using the proposed motion context descriptor. Our motion context descriptor retains both spatio-temporal and co-occurrance information and is similar to "superpixel" in image parsing [42], [43], but we have two merits: 1) Because abnormal events all occur in the foreground, our method only process the foreground region by ignoring the static background. 2) Our DPG is a patch-level method, which only needs to handle a much less number of units thus is more efficient.

A. Basic Patch Descriptor

We first partition the image into a few units, 2D image patch (30×30 pixels for each). Then we calculate the motion energy of each pixel (the magnitude of motion vector), if more than half of the number of pixels in each unit greater than zero, we consider such a unit as local foreground unit and describe it by fusing both motion and appearance information, as shown in Fig. 3.

For motion feature, we adopt the Multi-scale Histogram of Optical Flow (MHOF) proposed in [19], which preserves more temporal contextual information. After estimating the motion field [44], we quantize each pixel (x, y) into the MHOF using Eq. (4):

$$h(x,y) = \begin{cases} \text{round}(\frac{\text{p}\theta(\text{x,y})}{2\pi}) \bmod \text{p} & r(x,y) < \tau \\ \text{round}(\frac{\text{p}\theta(\text{x,y})}{2\pi}) \bmod \text{p} + \text{p} & r(x,y) \geq \tau \end{cases} \quad (4)$$

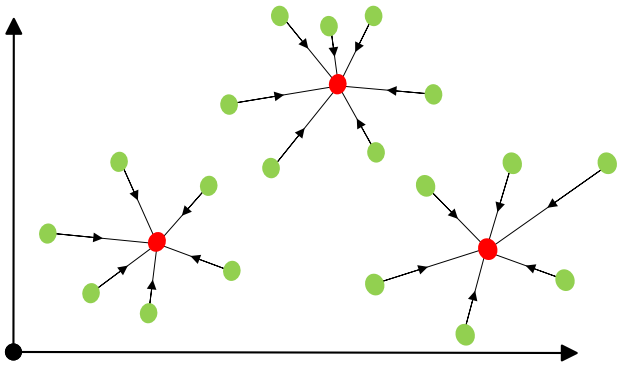


Fig. 4. Dynamic Patch Grouping (DPG). Each dot (red or green) denotes a local patch. The black arrow is the pairwise similarity. Each grouping stands for a motion context, which is grouped by most similar patches in both spatial and feature space. The red ones are the center of each motion context.

where $r(x,y)$ and $\theta(x,y)$ are the motion energy and motion direction of motion vector at (x,y), respectively. In our implementation, the MHOF has two layers of $B = 16$ bins as shown in Fig. 3: the first $\mathrm{p} = 8$ bins denote 8 directions with motion energy $r < \tau$ in the inner layer; the next $\mathrm{p} = 8$ bins correspond to $r \geq \tau$ ($\tau = 1$) in the outer layer.

For appearance feature, we use the Edge Orientation Histogram (EOH) to represent each unit by an $8-$bin feature vector. We first filter the image using Sobel masks, such as $[-1,0,1]$ and $[-1,0,1]^T$, and get the gradient image in x and y direction. Then, every pixel is quantized accordingly. Moreover, there are some foreground image patches containing too much noise or background pixels, we need to eliminate them for robustness depending on the following two criteria below:

i The foreground ratio, i.e. we consider it as the background, if

$$ratio = \frac{\sharp\{foreground\ pixel\}}{\sharp\{background\ pixel\}} < 1, \tag{5}$$

where the foreground mask is generated by the background model.

ii The Entropy of the MHOF,

$$E(H) = -\sum_i H(i)\log(H(i)), i = \{1 \ldots B\}, \tag{6}$$

where H is the feature vector MHOF of each patch, B is the feature dimension, i.e. $B = 16$ in our case, and for $E < 1$, we consider it as the intersection of different moving objects.

B. Dynamic Patch Grouping (DPG)

Motivated by Superpixel methods [42], [43] for image representation, we intend to adaptively cluster the similar image patches into one group for video event representation, called Dynamic Patch Grouping (DPG). In order to get a global optimization, we adopt the Normalized Cuts (NCut) [45] algorithm here for DPG. We first build the similarity matrix and estimate the group number accordingly. Then we use NCut to split the region and cluster the most similar patches into groups to generate the motion context. We call this procedure as DPG, as shown in Fig. 4.

Initially, we consider the 8-connected foreground 2D patches as a whole set, and construct the graph $\mathrm{G} = (\mathrm{V}, \mathrm{E})$ by taking each patch as a node, where V is the nodes consisting of n patches and E is the edge set. The symmetric similarity matrix S is defined with the edge weight as w_{ij}, which corresponds to pairwise similarity as in Eq. (7):

$$\begin{aligned} w_{ij} &= S_{Img} * S_{Dist} \\ &= \exp\left(-\frac{\|F(i)-F(j)\|^2}{\sigma_F^2}\right)\times \\ &\begin{cases} \exp\left(-\frac{\|x(i)-x(j)\|^2}{\sigma_x^2}\right) & \|x(i)-x(j)\|^2 < \tau_c \\ 0 & otherwise \end{cases} \end{aligned} \tag{7}$$

where $F(\cdot)$ is the feature vector of each unit, which is combined by MHOF and EOH; $x(\cdot)$ indicates the pixel position respectively; and σ_F and σ_x are scale parameters. We predefine the distance threshold τ_c to specify the neighbor of each node.

Actually, the DPG can be considered as a labeling procedure, in which one label $c \in \{1, \ldots, C\}$ is assigned to each node i. Let $\overrightarrow{y_c} = \{y_{ic}\}_{n\times 1}$ be a partitioning vector with $y_{ic} = 1$ if i belongs to the $k-$th segment and $y_{ic} = 0$ otherwise.

$$NCut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}, \tag{8}$$

where $cut(A,B) = \sum_{i\in A, j\in B} s(i,j)$ is the cut value and $assoc(A,V) = \sum_{i\in A, j\in V} s(i,j)$ is the total connection from the vertex set A to all vertices in G. To minimize the NCut in Eq. (8), the issue can be transformed into a standard eigenvalue problem:

$$D^{-\frac{1}{2}}(D-S)D^{-\frac{1}{2}}z = \lambda z, \tag{9}$$

where D is a diagonal matrix with $\sum_j s(i,j)$ on its diagonal and zero otherwise. The eigenvector corresponding to the second smallest eigenvalue can be used to partition V into A and B. In the case of multiple classes partitioning, the bipartition can be utilized recursively or just apply the eigenvectors corresponding to the $K+1$ smallest eigenvalues. The procedure of DPG is as shown in Alg. 1.

As mentioned by Wang et al. [46], the determination of the number of clusters is a challenging problem for clustering algorithms. In our case, we design an optimization processing to estimate the number of classes for Normalized Cuts (NCut) using the sum Energy of each cluster E_{N_C}:

$$E_{N_C} = \sum_{i=1}^{N_C}\sum_{j\in C_i} \|F_j^{(i)} - \overline{F}^{(i)}\|_2^2, \tag{10}$$

where $N_C \in [1 \ldots N_{max}]$ is the range of NCut clusters, C_i is the set of points belonging to cluster i and $\overline{F}^{(i)}$ is the mean value of cluster C_i. Therefore, our intention is to minimize the within-cluster sum of squares of Eq. (10). We denote the first derivative of E_{N_C} as ∇E_{N_C}. While the number N_C increases starting from one, the value of energy E_{N_C} will decrease. The optimal value of N_C is the first ∇E_{N_C} greater than zero, that is the energy E_{N_C} start to increase. In contrast with other pixel

Algorithm 1 Dynamic patch grouping

Input: Similarity matrix $S \in \mathbb{R}^{n\times n}$, number of clusters to be constructed c.

Output: Clusters $A_1, \ldots, A_c$

1: Construct a similarity graph by Eq. (7)
2: Compute the unnormalized Laplacian L
3: Compute the first k generalized eigenvectors $u_1, \ldots, u_c$ as columns
4: Let $U \in \mathbb{R}^{n\times c}$ be the matrix containing the vectors $u_1, \ldots, u_c$ as columns
5: **for** each $i = 1 \ldots n$ **do**
6: Let $y_i \in \mathbb{R}^c$ be the vector corresponding to the $i-$th row of U.
7: Cluster the points $(y_i)_{i=1,\ldots,n}$ in $\mathbb{R}^c$ with the $k-$means algorithm into clusters $A_1, \ldots, A_c$.
8: **end for**

based clustering methods, our motion context is based on patch level. Since the number of patch is smaller, the computation burden of NCut operation is also lower.

C. Motion Context

Similar patches of the same event have been grouped by DPG and we represent them as motion context here by fusing the spatio-temporal information. Suppose each motion context is constructed by N_m local patches, and the motion and appearance information of each patch is described by $16-$bin MHOF and $8-$bin EOH, respectively. Therefore, each motion context can be represented below,

- For motion feature, we consider both motion energy and motion direction information. For motion energy, max, min, mean and standard deviation of motion energy of N_m patches are used. For motion direction, the average of N_m $16-$bin MHOF is adopted.
- For appearance feature, the average of EOH from N_m patches with $8-$bin is adopted.

Then we normalize them respectively, and combine them into a high-dimensional feature vector as our event descriptor, where the dimension d is equal to 28 in our case.

V. Spatio-temporal Abnormal Event Detection

In this section, we formulate the anomaly detection as a retrieval problem. As mentioned before, the probability model based method may not provide a good matching especially with limited training examples in high-dimensional feature space. Thus, we detect abnormal event by measuring the similarity, i.e. finding the nearest neighbor sample related to each testing sample. Image similarity search is a fundamental problem in computer vision. Define the training sample or database as $\mathbf{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathbb{R}^d$. Given a query sample $\mathbf{y} \in \mathbb{R}^d$, the similarity $d(\cdot)$ may vary depending on specific application or dataset. We define a ℓ_p-norm distance d here,

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_p = (\sum_{j=1}^{d} |\mathbf{x}(j) - \mathbf{y}(j)|^p)^{1/p}, \qquad (12)$$

Algorithm 2 Abnormal Event Retrieval

Input: Training Data Samples $\mathbf{D} = \{\mathbf{x}_1, \ldots \mathbf{x}_N \in \mathbb{R}^d\}$
Testing Data Sample $\mathbf{y} \in \mathbb{R}^d$

Output: $\mathbf{x}^*$, the label of $\mathbf{y}$

1: Generate $R \in \mathbb{R}^{k\times d}$ with independent normal distribution $R_{ij} \sim N(0, 1)$
2: **for** each $i = 1 \ldots N$ **do**
3: $\mathbf{x}_i^k = \sigma(R\mathbf{x}_i)$,
where $\sigma(x) = \begin{cases} 1 & x > 0 \\ 0 & x \le 0 \end{cases}$
4: **end for**
5: $\mathbf{x}^* = CompactProjSearch(\mathbf{D}, \mathbf{x}^k, \mathbf{y}, R)$
6: Detect anomaly

$$\mathbf{y} = \begin{cases} normal & d(\mathbf{x}^\star, \mathbf{y}) \le 1.2 \times \tau_{x^\star} \\ abnormal & d(\mathbf{x}^\star, \mathbf{y}) > 1.2 \times \tau_{x^\star} \end{cases} \qquad (11)$$

Function $\mathbf{x}^* = CompactProjSearch(\mathbf{D}, \mathbf{x}^k, \mathbf{y}, R)$

1: Generate the code $\mathbf{y}^k = \sigma(R\mathbf{y})$.
2: **for** each $i = 1 \ldots N$ **do**
3: Compute the Hamming distance $\|\mathbf{y}^k - \mathbf{x}_i^k\|_0$
4: **end for**
5: Select the points $\tilde{\mathbf{D}} \subset \mathbf{D}$ with whose code have $\mathcal{T} = \mathrm{O}(n^l)(0 < l < 1)$ smallest Hamming distance from $\mathbf{y}^k$.
6: $\mathbf{x}^* = \arg\min_{\mathbf{x}\in\tilde{\mathbf{D}}} d(\mathbf{x}, \mathbf{y})$

where $p = 1$ is the ℓ_1-norm, and $p = 2$ is ℓ_2-norm (we use $p = 2$ in our case). The critical task is to efficiently return a vector $\mathbf{x}^* \in \mathbf{D}$, which is similar to $\mathbf{y}$, i.e. the distance $d(\mathbf{y}, \mathbf{x}^*)$ is as small as possible. The most straightforward method is to simply scan through the database and return the point $\mathbf{x}^\star \in \mathbf{D}$ that minimizes $d(\mathbf{x}, \mathbf{y})$, yielding an excellent neighbor: $d(\mathbf{y}, \mathbf{x}^\star) = \min_{\mathbf{x}\in\mathbf{D}} d(\mathbf{x}, \mathbf{y})$. However, the algorithm complexity of such a method is about $O(dN)$. If in practical searching d and N are large, such an exhaustive searching for finding nearest neighbors in the high-dimensional feature space is too time consuming and unacceptable for both abnormal event detection and web-scale image retrievals.

Efficient similarity search across large image training data depends critically on the availability of compact image representations and good data structures for indexing them. The random projection (by Johnson-Lindenstrauss lemma [47]) presents a way for efficient similarity searching. Given a random projection function, $f : \mathbb{R}^d \to \mathbb{R}^k, k < d$ into a $k = O(\log(N))$ dimensional space nearly preserves the pairwise distances between the points with high probability:

$$(1-\varepsilon)\|\mathbf{x} - \mathbf{y}\|_2^2 \le \|f(\mathbf{x}) - f(\mathbf{y})\|_2^2 \le (1+\varepsilon)\|\mathbf{x} - \mathbf{y}\|_2^2 \quad (13)$$

for all pairs $\mathbf{x}, \mathbf{y} \in \mathbf{D}$. This phenomenon is fairly flexible with respect to the distribution of f. Therefore, in this paper, we adopt compact projection [48] for motion context similarity search. Compact projection is an approximate nearest neighbor algorithms with excellent performance guarantees, in some cases nearly optimal. In contrast to locality sensitive hashing (LSH) [49], which exploits the distance preserving properties of random projections, compact projection generates

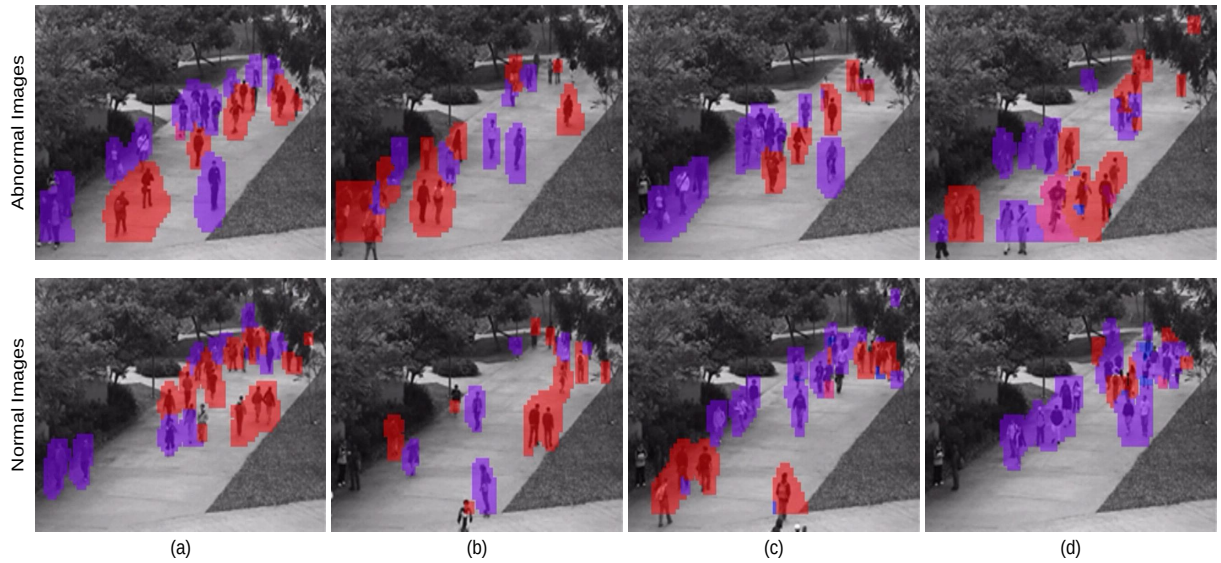


Fig. 5. The example results of Dynamic Patch Grouping (DPG) for motion context extraction. Different colors denote different moving direction of each motion context. The images of top row include both normal and abnormal events, while the images of bottom row only include normal ones.

compact binary representations of image data based on random projections.

There are two tuning parameters, k and l, in which k controls the number of random projections, and l determines the feedback ratio of rough searching by binary compact projection. The greater the value of these two parameters are, the higher the accuracy of the searching results is, while the lower the efficiency of searching. Here, we have $k > \log N$, and set $k = \frac{N}{2}$ and $l = 0.1$.

For each training sample $\mathbf{x}_i, i \in \{1, \cdots, N\}$, we find its nearest neighbor $\mathbf{x}_i^\star$ by Alg.2. Therefore, we can set the threshold τ_{x_i} of each training sample according to $d(\mathbf{x}_i, \mathbf{x}_i^\star)$. The value of $\tau_{x^\star} = d(\mathbf{x}_i, \mathbf{x}_i^\star)$ is related to the radius of green circle of each training sample in Fig. 2. Therefore, the threshold τ_{x_i} is different for varying $\mathbf{x}_i$.

For testing, we find the nearest neighbor $\mathbf{x}^\star$ of query sample $\mathbf{y}$ and compute the similarity $d(\mathbf{x}^\star, \mathbf{y})$, whether $\mathbf{y}$ is normal or not depends on both the similarity $d(\mathbf{x}^\star, \mathbf{y})$ and the threshold $\tau_{x^\star}$ as shown in Eq. (11). Please refer to Alg.2 for more details. When there are some noises in the detection, the rough detection results have many false alarms, which will affect the accuracy. As the abnormal event cannot occur stand-alone in only one frame or one patch, there exists Markov properties. Therefore, we use a simplified version of spatial-temporal Markov random field to eliminate noise and maintain speed.

VI. Experiments and Comparisons

A. Dataset

We use the UCSD dataset [16], [50] in our experiments, where the crowd density varies from sparse to very crowded. The training set are all normal events and contain only pedestrians. The abnormal events in testing set are due to either 1) the circulation of non pedestrian entities in the walkways, or 2) anomalous pedestrian motion patterns. Commonly occurring anomalies include bikes, skaters, small cars, and people walking across a walkway or in the grass that surrounds it.

For Ped1, the training set includes 34 normal video clips and the testing set contains 36 video clips in which some of the frames have one or more anomalies present (a subset of 10 clips in testing set are provided with pixel-level binary masks to identify the regions containing abnormal events). For each clip, there are about 200 frames with the resolution 158×238, The total number of anomalies frames (≈ 3400) is somewhat smaller than that of normal frames (≈ 5000).

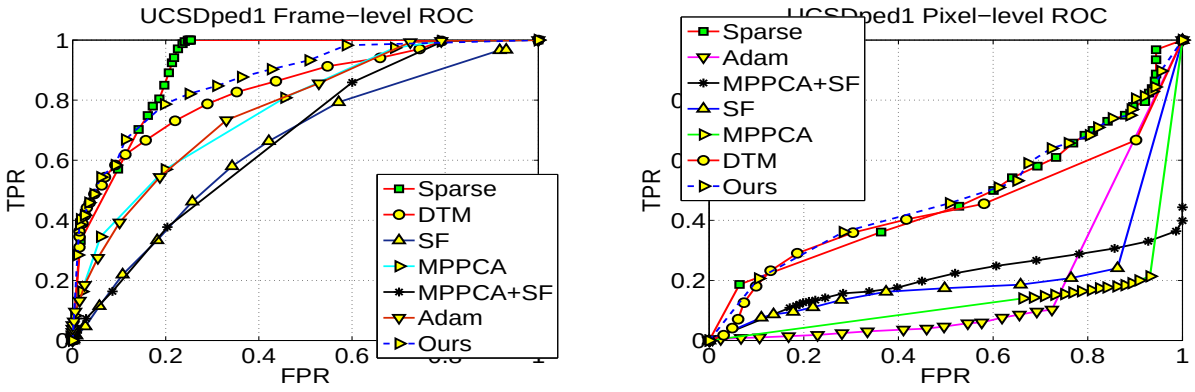
For Ped2, the training set includes 16 normal video clips and the testing set contains 14 video clips with the image resolution 320×240. The total number of anomalies frames (≈ 2384) is also smaller than that of normal frames (≈ 2566).

B. Measurements

To test the effectiveness of our proposed algorithm, two different levels of measurements are applied for evaluation, i.e. Pixel-level and Frame-level.

- Frame-level: If a frame contains at least one abnormal pixel, it is considered as a detection. These detections are compared to the frame-level ground truth annotation of each frame. Note that this evaluation does not verify whether the detection coincides with the actual location of the anomaly. It is therefore possible for some true positive detections to be "lucky" co-occurrences of erroneous detections and abnormal events.
- Pixel-level: To test the localization accuracy, detections are compared to pixel-level ground truth masks, on a subset of ten clips. The procedure is similar to that described above. If at least 40% of the truly anomalous pixels are detected, the frame is considered detected correctly, and counted as a false positive otherwise.

The Receiver Operating Characteristic (ROC) curve is used to measure the accuracy.



(a) Frame-level ROC

(b) Pixel-level ROC

Fig. 6. The evaluation results of UCSD Ped1 dataset. (a) Frame-level ROC for Ped1 Dataset, (b) Pixel-level ROC for Ped1 Dataset.

	EER	RD	AUC
MPPCA [16]	40%	18%	20.5%
SF-MPPCA [16]	32%	18%	21.3%
MDT [16]	25%	45%	44.1%
Adam [15]	38%	24%	13.3%
Sparse [19]	19%	46%	46.1%
Ours	**23**%	**47**%	**47.1**%

TABLE I
THE EVALUATION RESULTS OF UCSD PED1 DATASET. QUANTITATIVE COMPARISON OF OUR METHOD WITH [16], [15] AND [19]: EER IS EQUAL ERROR RATE, RD IS RATE OF DETECTION, AND AUC IS THE AREA UNDER ROC.

	EER	AUC
Adam [15]	42%	63.4%
SF [16]	42%	62.3%
SF-MPPCA [16]	36%	71.0%
MPPCA [16]	30%	77.4%
MDT [16]	25%	84.8%
Sparse [19]	25%	86.1%
Ours	**24.8**%	**86.8**%

TABLE II
THE STATISTICAL RESULT OF UCSD PED2 DATASET.

C. Performance

The performance of motion context extraction is crucial for our algorithm. The key challenge is how to separate the adjacent objects or events in the foreground blobs. As shown in Fig. 5, we list the result of motion context extraction for both normal and abnormal events. Different motion contexts have been extracted and labeled using different colors, which correspond to the main moving direction. We can see that the abnormal objects have been separated and labeled, which is useful for the further processing. Moreover we only extract motion context from the foreground region by using a patch-level method (DPG), thus the estimated number of cluster E_{N_C} in Eq. (10) is not too much (usually $E_{N_C} = 2$ from each adjacent foreground region).

In Fig. 6, we compare our method with the state-of-the-art methods, such as MDT [16], Social Force model [17], MPPCA [35], Adam's work [15] and our previous work [19] by using sparse reconstruction cost (SRC) for abnormal event detection. We use pixel-level and frame-level measurements defined above for quantitative comparison. It is easy to find that for frame-level measurement, our ROC curve is better than others except a bit lower than [19], and for pixel-level measurement, our ROC curve outperforms all of others.

We present the results in Tab. I with different evaluation criteria:

- Equal Error Rate (EER), which reports the percentage of misclassified frames when the false positive rate is equal to the miss rate;
- Rate of Detection (RD), which is the detection rate at equal error point;
- Area Under Curve (AUC), which measures the area under the ROC curve.

For EER, ours is 23%, which is slightly worse than our previous work [19] 19%, but outperforms all of others. For RD, ours is 47%, which outperforms the state-of-the-art methods, including [19] with RD as 46%. For AUC, ours is 47.1%, which also performs the best and the second best one is [19] with AUC as 46.1%.

Some image results are shown in Fig. 7 (the abnormal events are labeled by red masks), in which the top row is generated by MDT method [16], the second and third rows are given by SF-MPPCA method [16] and SRC method [19], respectively, and the bottom is by our algorithm. For SF-MPPCA method, they completely miss the skater in (b), the person running in (c) and the bike in (d). For MDA method, although they detect nearly all of the abnormal events, the foreground mask is too large, which is not accurate. For ours, we can detect the abnormal objects robustly with more accurate boundary, such as bikers, skaters, small cars, etc. Obviously, ours outperforms the other methods.

In Fig. 8, we show the results on UCSD Ped2 dataset. As Ped2 dataset does not provide pixel-level groundtruth, we only use Frame-level ROC for comparison, and ours outperforms the state-of-the-art methods as well. The statistical result is also shown in Tab. II. For the Equal Error Rate (EER), ours is 24.8%, which is better than MDT [16]; and for the Area Under Curve (AUC), ours is 86.8%, which also outperforms the state-of-the arts [16] [15] .

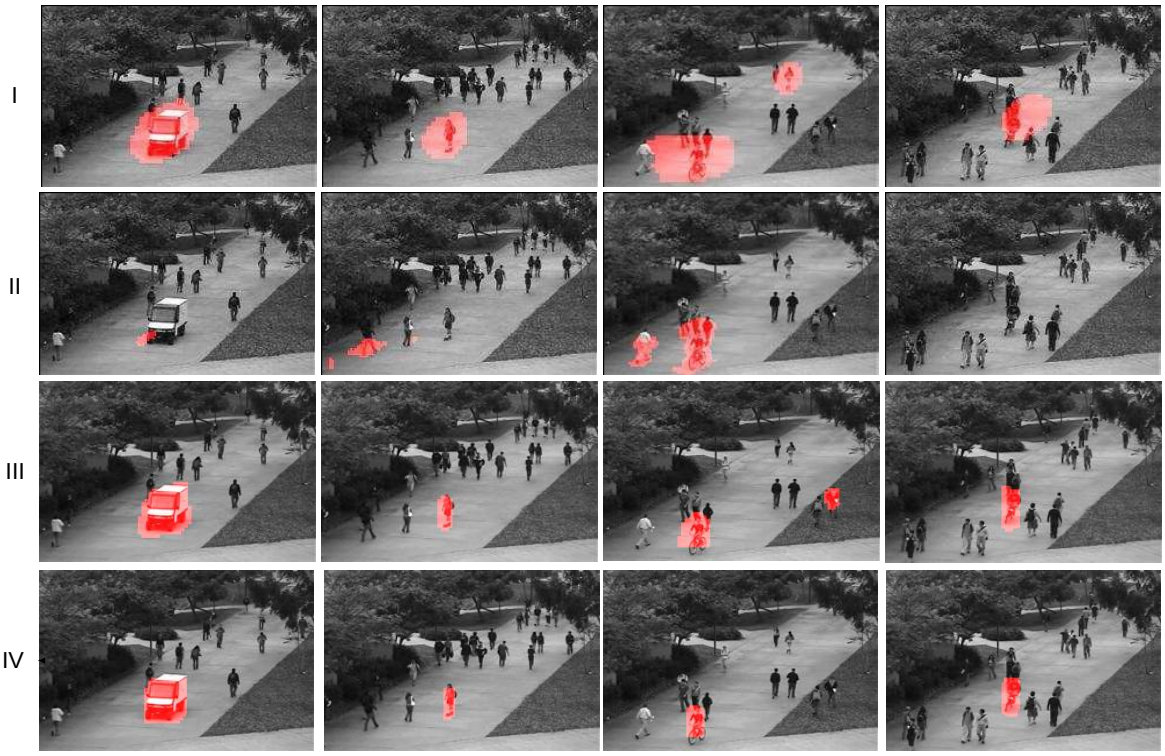


Fig. 7. Examples of abnormal detections using (I) the MDT method [16]; (II) the SF-MPPCA method [16], which completely misses the skater in (b), the person running in (c) and the bike in (d); (III) the SRC method [19]; and (IV) our results.

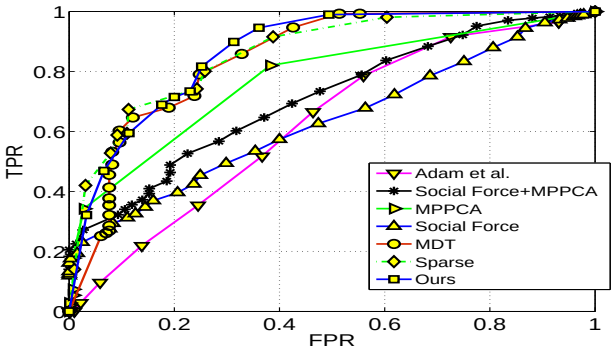


Fig. 8. Frame-level ROC for UCSD Ped2 Dataset.

	15×15	30×30	45×45
AUC	86.2%	86.8%	83.7%

TABLE III
COMPARISON THE AUC OF DIFFERENT IMAGE PATCH SIZES FOR UCSD PED2 DATASET.

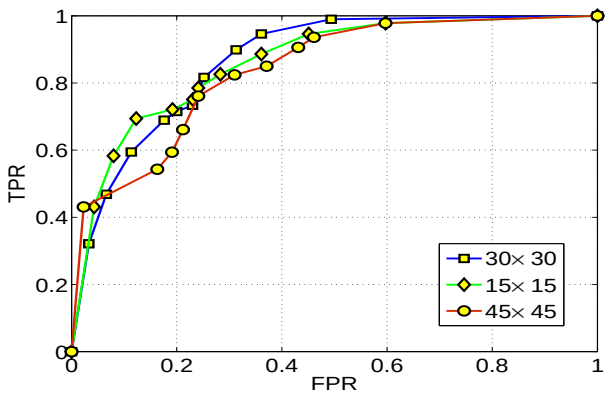


Fig. 9. Comparison the ROC of different image patch sizes (15×15, 30×30, 45×45) for UCSD Ped2 Dataset.

D. Comparisons

1) The Influence of Image Patch Size: In this sub-section, we will evaluate the influence of different image patch sizes for UCSD Ped2 dataset and we choice the patch size as 15×15, 30×30 and 45×45. Then, the statistical results of AUC is shown in Tab. III and the ROC curves are shown in Fig. 9. We find that the AUC of 15×15 is similar to that of 30×30, this is because our DPG method can adaptively group related image patches to generate the motion context automatically; and the result of the larger size 45×45 is much more worse as it is too large and contains more background pixels and noise, which makes the result deteriorated.

2) Multi-layer MHOF vs. Single-layer HOF: In this subsection, we compare our multi-layer MHOF with the traditional HOF, i.e. single layer HOF, where for MHOF we set the motion magnitude threshold as $\tau = 1$ and the scale as $s = 2$. The results of AUC is shown in Tab. IV and the ROC curves are shown in Fig. 10. We find that the AUC of multi-scale, i.e. the MHOF, is better than single-layer, because the MHOF is more precise to represent the motion, e.g. the fast moving objects.

3) Searching Efficiency: The efficiency for similarity searching is crucial for our proposed video anomaly detection method. In this section, we compare the efficiency of Compact Projection (CP) based searching with the traditional K-Nearest neighbor (KNN) searching. In the training procedure, we

	Multi-layer MHOF	Single-layer HOF
AUC	86.8%	78.73%

TABLE IV
COMPARISON THE AUC OF MULTI-LAYER WITH SINGLE-LAYER FOR UCSD PED2 DATASET.

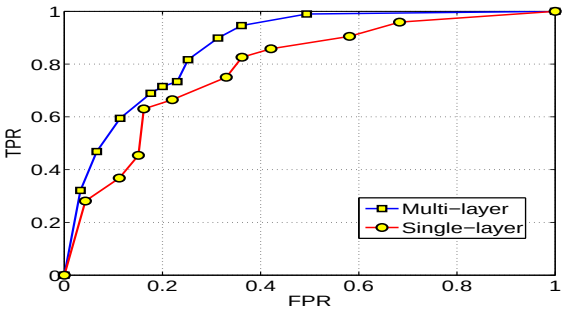


Fig. 10. Comparison the ROC of Multi-layer MHOF (S=2) with Single-layer MHOF for UCSD Ped2 Dataset.

collect the query samples from the normal training dataset, and for the testing procedure, we extract testing sample and find the most similar query sample for anomaly detection. In our case, we collect about 10k and 4k query samples from the UCSD Ped1 and Ped2 dataset, respectively. As shown in Fig. 11, we compare the time cost for one thousand times searches. Clearly, the CP based searching is more efficient than exhaustive K-NN search. In practice, the amount of video dataset will be much larger thereby generating more nearest neighbors. As point out in [48], when the size of query dataset increases, CP is more efficient and will occupy less memory than other methods. With the help of CP to speed up the NN search, our method is efficient for video anomaly searching in practice. All the experiments are done on a Pentium 4 3.0GHz machine with 4GB memory.

4) Time Consumption: We compare the time consumption here in this subsection. For the "sparse" method [19], it takes 3.8second/frame for UCSD dataset on the platform with 2GB RAM and 2.6GHz CPU. For MDT [16], the testing time is about 25second/frame on a standard platform with 3GHz CPU and 2GB RAM. For Adam [15], the authors claim that their propose method can run in real-time. For our method, it takes about 1.2second/frame on a platform with 4GB RAM and 3GHz CPU. So for video anomaly searching, our proposed method is more efficient than most of the state-of-the-art methods except Adam [15], however ours outperforms other methods as shown above.

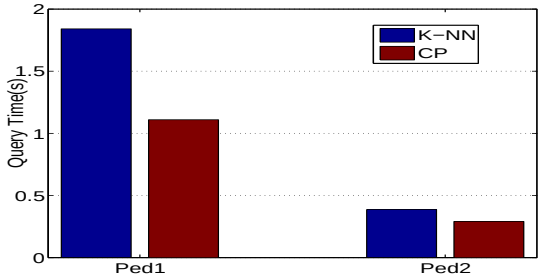


Fig. 11. Speed comparison with Compact Projection (CP) and K-Nearest neighbor (KNN).

VII. CONCLUSION

In this paper, we present an algorithm for abnormal event detection in spatio-temporal video space by considering video anomaly detection as a retrieval problem. Two key technologies are developed here for both event representation and anomaly measurement. Motivated by "superpixel" methods [42], [43], we design a patch-level event descriptor, named as dynamic patch grouping (DPG), to represent each event as motion context by associating both motion and appearance cues, which is more effective and flexible than existing methods, such as [33], [15] or [19]. For anomaly measurement, the abnormal event is detected based on the nearest neighborhood searching procedure via dynamic threshold, which overcomes the contradiction between insufficient training data and high-dimensional feature vector for fitting most state-of-the-art probability models. Moreover, the compact projection is applied here to speed up the searching procedure. The experiments on the benchmark datasets show favorable results when compared with the state-of-the-art methods.

REFERENCES

[1] V. Singh and M. Kankanhalli, "Adversary aware surveillance systems," *Information Forensics and Security, IEEE Transactions on*, vol. 4, no. 3, pp. 552–563, 2009.
[2] S. Avidan, "Ensemble tracking," *IEEE transactions on pattern analysis and machine intelligence*, pp. 261–271, 2007.
[3] Z. Khan and I. Gu, "Joint feature correspondences and appearance similarity for robust visual object tracking," *Information Forensics and Security, IEEE Transactions on*, vol. 5, no. 3, pp. 591–606, 2010.
[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005, pp. 886–893.
[5] L. Wang, "Abnormal walking gait analysis using silhouette-masked flow histograms," in *ICPR*, vol. 3, 2006, pp. 473–476.
[6] S. Wang and H. Lee, "A cascade framework for a real-time statistical plate recognition system," *Information Forensics and Security, IEEE Transactions on*, vol. 2, no. 2, pp. 267–282, 2007.
[7] X. Yu, K. Chinomi, T. Koshimizu, N. Nitta, Y. Ito, and N. Babaguchi, "Privacy protecting visual processing for secure video surveillance," in *ICIP*, 2008, pp. 1672–1675.
[8] U. Park and A. Jain, "Face matching and retrieval using soft biometrics," *Information Forensics and Security, IEEE Transactions on*, vol. 5, no. 3, pp. 406–415, 2010.
[9] Y. Cong, J. Yuan, and J. Luo, "Towards scalable summarization of consumer videos via sparse dictionary selection," *Multimedia, IEEE Transactions on*, vol. 14, no. 1, pp. 66–75, 2012.
[10] Y. Cong, H. Gong, S. Zhu, and Y. Tang, "Flow mosaicking: Real-time pedestrian counting without scene-specific learning," in *CVPR*, 2009, pp. 1093–1100.
[11] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," in *CVPR*, 2004.
[12] A. M. Cheriyadat and R. J. Radke, "Detecting Dominant Motions in Dense Crowds," *Selected Topics In Signal Processing, IEEE Journal Of*, vol. 2, no. 4, pp. 568–581, 2008.
[13] C. Loy, T. Xiang, and S. Gong, "Detecting and discriminating behavioural anomalies," *Pattern Recognition*, vol. 44, no. 1, pp. 117–132, 2011.
[14] Y. Benezeth, P. Jodoin, V. Saligrama, and C. Rosenberger, "Abnormal events detection based on spatio-temporal co-occurences," in *CVPR*, 2009.
[15] E. S. I. R. D. Adam, A.; Rivlin, "Robust real-time unusual event detection using multiple fixed-location monitors," *TPAMI*, vol. 30(3)Volume 30, pp. 555 – 560, 2008.
[16] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *CVPR*, 2010.
[17] M. S. Ramin Mehran, Alexis Oyama, "Abnormal crowd behavior detection using social force model," in *CVPR*, 2009.
[18] O. Boiman and M. Irani, "Detecting irregularities in images and in video," in *ICCV*, 2005.

[19] Y. Cong, J. Yuan, and J. Liu, "Sparse Reconstruction Cost for Abnormal Event Detection," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3449–3456.

[20] V. Saligrama and Z. Chen, "Video anomaly detection based on local statistical aggregates," in *CVPR*. IEEE, 2012, pp. 2112–2119.

[21] O. Popoola and K. Wang, "Video-based abnormal human behavior recognition—a review," *TSMC-C*, 2012.

[22] V. Saligrama, J. Konrad, and P. Jodoin, "Video anomaly identification," *Signal Processing Magazine, IEEE*, vol. 27, no. 5, pp. 18–33, 2010.

[23] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank, "A system for learning statistical motion patterns," *TPAMI*, vol. 28, no. 9, pp. 1450–1464, 2006.

[24] X. Wang, X. Ma, and W. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 31, no. 3, pp. 539–555, 2009.

[25] M. S. Saad Ali, "Floor fields for tracking in high density crowd scenes," in *ECCV*, 2008.

[26] T. Xiang and S. Gong, "Video behavior profiling for anomaly detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPMAI)*, vol. 30, no. 5, pp. 893–908, 2008.

[27] B. Morris and M. Trivedi, "Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 11, pp. 2287–2301, 2011.

[28] N. Vaswani, A. K. Roy-Chowdhury, and R. C. Khan, ""shape activity": A continuous-state hmm for moving/deforming shapes with application to abnormal activity detection," *Image Processing, IEEE Transactions on*, vol. 14, no. 10, pp. 1603–1616, 2005.

[29] I. Ivanov, F. Dufaux, T. Ha, and T. Ebrahimi, "Towards generic detection of unusual events in video surveillance," in *AVSS*, 2009, pp. 61–66.

[30] F. Jiang, J. Yuan, S. Tsaftaris, and A. Katsaggelos, "Anomalous video event detection using spatiotemporal context," *Computer Vision and Image Understanding*, pp. 323–333, 2011.

[31] M. Thida, H.-L. Eng, M. Dorothy, and P. Remagnino, "Learning video manifold for segmenting crowd events and abnormality detection," in *ACCV*, 2010, pp. 439–449.

[32] M. Thida, H.-L. Eng, and P. Remagnino, "Laplacian eigenmap with temporal constraints for local abnormality detection in crowded scenes," *Systems, Man and Cybernetics, IEEE Transactions on*, 2013.

[33] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *CVPR*, 2009.

[34] L. Itti and P. Baldi, "A principled approach to detecting surprising events in video," in *CVPR*, 2005.

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

[36] I. Tziakos, A. Cavallaro, and L. Xu, "Event monitoring via local motion abnormality detection in non-linear subspace," *Neurocomputing*, 2010.

[37] A. J. W.Yu, "Modeling crowd turbulence by many-particle simulation," *Phycical Review E*, vol. 76(4), p. 046105, 2007.

[38] S. Wu, B. Moore, and M. Shah, "Chaotic invariants of Lagrangian particle trajectories for anomaly detection in crowded scenes," in *CVPR*, 2010.

[39] O. Boiman and M. Irani, "Detecting irregularities in images and in video," *IJCV*, vol. 74, no. 1, pp. 17–31, 2007.

[40] B. Zhao, L. Fei-Fei, and E. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *CVPR*, 2011.

[41] Y. Cong, J. Yuan, and J. Liu, "Abnormal event detection in crowded scenes using sparse representation," *Pattern Recognition*, vol. 46, pp. 1851–1864, 2013.

[42] X. Ren and J. Malik, "Learning a classification model for segmentation," in *ICCV*, 2003.

[43] A. Moore, S. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel lattices," in *CVPR*, 2008.

[44] C. Liu, W. Freeman, E. Adelson, and Y. Weiss, "Human-assisted motion annotation," in *CVPR*, 2008.

[45] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[46] L. Wang, C. Leckie, K. Ramamohanarao, and J. Bezdek, "Automatically determining the number of clusters in unlabeled data sets," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 21, no. 3, pp. 335–350, 2009.

[47] W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," *Contemporary mathematics*, vol. 26, no. 189-206, pp. 1–1, 1984.

[48] K. Min, L. Yang, J. Wright, L. Wu, X. Hua, and Y. Ma, "Compact projection: Simple and efficient near neighbor search with practical memory requirements." CVPR, 2010.

[49] M. Datar, N. Immorlica, P. Indyk, and V. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proceedings of the twentieth annual symposium on Computational geometry*. ACM, 2004, pp. 253–262.

[50] "http://www.svcl.ucsd.edu/projects/anomaly."

Yang Cong (S'09-M'11) received the B.Sc. degree from Northeast University in 2004, and the Ph.D. degree from State Key Laboratory of Robotics, Chinese Academy of Sciences in 2009. He is a Research Fellow of National University of Singapore (NUS) and Nanyang Technological University (NTU) from 2009 to 2011, respectively. Now, he is an Associate Researcher of Chinese Academy of Science. His current research interests include compute vision, pattern recognition, multimedia and robot navigation. He is a member of IEEE.

Junsong Yuan (M'08) is a Nanyang Assistant Professor at School of EEE, Nanyang Technological University (NTU), Singapore. He is also the Program Director of Video Analytics at Infocomm Center of Excellence (INFINITUS), EEE, NTU. He received Ph.D. from Northwestern University and M.Eng. from National University of Singapore, in 2009 and 2005, respectively. Before that, he was selected to the Special Program for the Gifted Young at Huazhong University of Science and Technology, and received his B.Eng. in 2002.

Dr. Yuan's research interests include computer vision, video analytics, large-scale visual search and mining, human computer interaction, biomedical image analysis etc. He has published over 90 technical papers and received over 4.2 million Singapore dollar research grants as PI or Joint-PI. He received Nanyang Assistant Professorship and Tan Chin Tuan Exchange Fellowship from Nanyang Technological University, Outstanding EECS Ph.D. Thesis award from Northwestern University, Best Doctoral Spotlight Award from IEEE Conf. Computer Vision and Pattern Recognition Conference (CVPR'09), and National Outstanding Student Award from Ministry of Education, P.R.China. He is Area Chair of Winter Applications of Computer Vision (WACV 2014), and Organization Co-Chair of Asian Conf. on Computer Vision (ACCV 2014). He gives tutorials at a few conferences including ICIP'13, FG'13, ICME'12, SIGGRAPH VRCAI'12, and PCM'12, and co-chairs workshops at CVPR'12, CVPR'13, and ICCV'13. He has filed three US patents and two provisional US patents.

Yandong Tang received his B.S. and M.S. degrees in mathematics from Shandong University, P. R. China, in 1984 and 1987, respectively. In 2002 he received the doctor's degree in applied mathematics from the University of Bremen, Germany. Currently he is a professor in Shenyang Institute of Automation, Chinese Academy of Sciences. His research interests include robot vision, image processing and pattern recognition.