Video Anomaly Detection and Localization Using Hierarchical Feature Representation and Gaussian Process Regression

Kai-Wen Cheng and Yie-Tarng Chen and Wen-Hsien Fang
Department of Electronic and Computer Engineering
National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C.
Email:{D10102101, ytchen, whf}@mail.ntust.edu.tw

Abstract

This paper presents a hierarchical framework for detecting local and global anomalies via hierarchical feature representation and Gaussian process regression. While local anomaly is typically detected as a 3D pattern matching problem, we are more interested in global anomaly that involves multiple normal events interacting in an unusual manner such as car accident. To simultaneously detect local and global anomalies, we formulate the extraction of normal interactions from training video as the problem of efficiently finding the frequent geometric relations of the nearby sparse spatio-temporal interest points. A codebook of interaction templates is then constructed and modeled using Gaussian process regression. A novel inference method for computing the likelihood of an observed interaction is also proposed. As such, our model is robust to slight topological deformations and can handle the noise and data unbalance problems in the training data. Simulations show that our system outperforms the main state-of-the-art methods on this topic and achieves at least 80% detection rates based on three challenging datasets.

1. Introduction

Visual analysis of suspicious events is a topic of great importance in video surveillance. A critical issue in anomaly analysis is to effectively represent an event to allow for a robust discrimination. Trajectory representation [11] is ubiquitous but unreliable in crowded scenes. Alternatively, local statistics of low-level observations are utilized in [1, 19]. These methods typically begin with extracting local spatio-temporal descriptors densely or in a sparse manner via interest point detection. To handle the inter- and intra-classes variations in normal events, mixture of models [14] or the bag-of-words techniques [15, 23] are performed. However, the geometric relations between local patterns have not been considered.

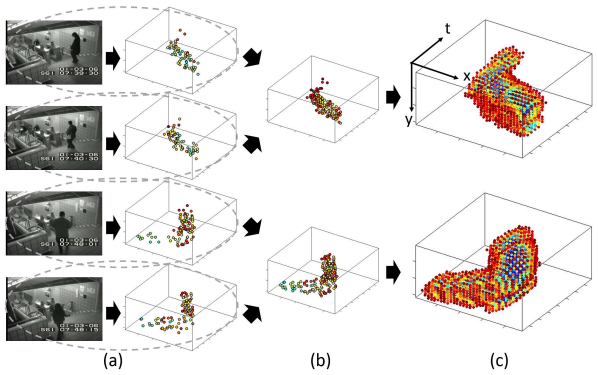


Figure 1: **Complex interaction modeling**: (a) Input videos are represented by a sparse set of interest points. (b) Incidents with similar spatio-temporal relationships of interest points are merged altogether to form deformable interaction templates. (c) Gaussian process regression is used to model each templates. The likelihood of being part of a specific interaction is indicated from low (red) to high (blue). Unlikely locations are invisible for better visualization.

Similar to [6], video event anomalies can be classified as local and global anomalies. A local anomaly is defined as an event that is different from its spatio-temporal neighboring events; whereas, a global anomaly is defined as multiple events that globally interact in an unusual manner, even if any individual local event can be normal. Most research on anomaly detection like [1, 14, 19] have focused more on detecting local anomalies such as objects with strange appearance or speed, but less on global anomaly. Global anomalies are common phenomenon in many scenarios like traffic surveillance. The methods in [3, 18] were devised to model the spatio-temporal relationships of dense features with heavy load in space and time, and did not work that well for modeling sparse features.

As video events can be discriminated from their geometric relations of spatio-temporal interest points (STIPs) in Fig. 1, this paper proposes a unified framework, shown in

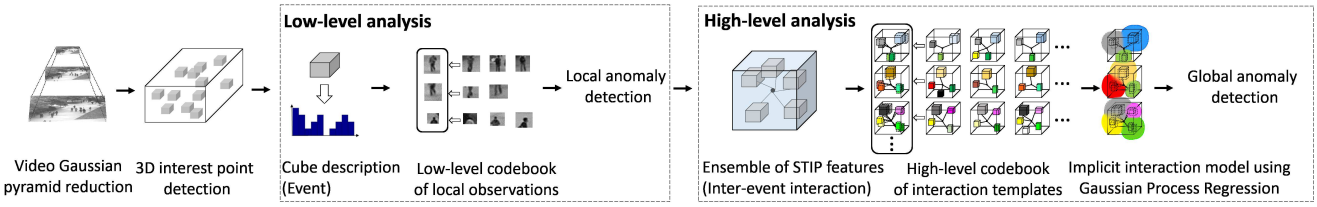


Figure 2: **Overview**: Local features are extracted around interest points in different scales and quantized into a low-level visual vocabulary. The local anomaly detection is to measure the k-NN distance of a test cuboid against the visual vocabulary. For the high-level analysis, ensembles of the nearby STIP features are extracted by dense sampling and are clustered to construct a high-level codebook of interaction templates. Gaussian process regression provides a probabilistic framework to model the geometric relations of the STIP features and detect global anomaly.

Fig. 2, to detect both local and global anomalies using a *sparse* set of STIPs. We identify local anomalies as those STIP features with low-likelihood visual patterns. To deal with inter-event interactions, we further collect an ensemble of the nearby STIP features and consider that an observed ensemble is regular if its semantic (appearance) and structural (position) relations of the nearby STIP features occur frequently. Global anomalies are identified as interactions that have either dissimilar semantics or misaligned structures with respect to the probabilistic normal models.

More specifically, the proposed approach has two main stages to deal with global anomaly. As recognizing global anomaly requires a set of normal interaction templates, we first pose the extraction of normal interactions from training videos as the problem of finding the frequent geometric relations of the nearby interest points. As shown in Fig. 1 b, the proposed extraction method builds a high-level codebook of interaction templates, each of which has an ensemble of STIPs arranged in a non-rigid deformable configuration. Moreover, it can efficiently deal with large training data by utilizing an optimal computation of high-dimensional integral images [22]. We next model the geometric relations of STIP features and propose a novel inference method using Gaussian process regression (GPR). GPR is more suitable on the topic of anomaly detection since it is fully non-parametric and robust to the noisy data and it also supports missing input values like sparse STIPs.

Compared to previous works to be discussed in Sec. 2, our method possesses several advantages: 1) it provides a novel hierarchical event representation to simultaneously deal with local and global anomalies; 2) it employs an efficient clustering method to extract deformable templates of inter-event interactions from training videos; 3) it constructs a GPR model based on a sparse set of STIPs, which is not only adaptive based on the available data, it can also learn interactions in a large context while individually locate abnormal events instead of taking an entire interaction as an atomic unit. Note that since our model is built upon STIPs rather than densely-sampled patches [18, 3], the space and time complexity of the event modeling can be greatly reduced (*e.g.* 150 STIPs v.s. 35301 dense patches in a $41\times41\times21$ volume). Experiments on three public datasets are conducted and the comparisons with the main state-of-the-art methods verify the superiority of our method.

The rest of paper is organized as follows. Sec. 3 introduces the hierarchical event and interaction representation. The high-level codebook construction is elaborated in Sec. 4. Sec. 5 details the GPR learning and inferring, and joint anomaly detection. Experiments are conducted in Sec. 6, and Sec. 7 concludes our work.

2. Related Work

A considerable amount of literature has been published on visual anomaly analysis. A detailed survey [16] on this topic shows the increasing publications in the last decades. Behavior representation, understanding and anomaly inference are the major issues. Oftentimes, normal event understanding is posed as a 3D pattern-learning problem. Suspicious events are treated as low-likelihood patterns with respect to either offline templates of normal events [19] or adaptive models learned from sequential data [1]. To detect anomalies from unconstrained scenes, Mahadevan *et al.* [14] proposed a mixtures of dynamic textures (MDT) model [4] to detect temporal and spatial abnormalities. These approaches (*e.g.* [19]) flag abnormal events based on independent location-specific statistical models and have not considered the relationships between local observations. Benezeth *et al.* [2] used a Markov Random Fields (MRF) model parameterized by a co-occurrence matrix to allow for spatial consistency detection. Real-time constrains are another pursuit in [1, 13]. One-class support vector machine [20] was used in [25, 5] to detect unusual behavior.

As for modeling group interactions, Cui *et al.* [7] proposed an interaction energy potential to model the interpersonal relationship. Social force model was extended from physics to analyze crowd dynamics [15]. These models strongly adhered to motion information and had their limitation in specific scenarios. Roshtkhari and Levine [18] encoded the spatio-temporal compositon (STC) of densely-

sampled 3D patches with a probability density function (pdf). The high-dimensional pdf had to be approximated but suffered from the curse of dimensionality. Boiman and Irani [3] proposed an inference by composition (IBC) algorithm to compute the joint probability between a database and a query ensemble. However, the underlying graph expands substantially in accordance with the database size leading to inefficient message passing. Also, [3, 18] modeled the spatio-temporal relations of densely-sampled 3D patches which are extracted with high computational demand.

Gaussian process regression has been applied to trajectory analysis [11] and human motion modeling [24]. For the multi-object activity modeling, Loy *et al*. [12] formulated the non-linear relationships between decomposed image regions as a regression problem. As the normalness of a specific region at time *t* is predicted based on its complements from $(t-1)$, spatial configurations between objects can be well characterized. However, Markov assumption cannot handle complex causality.

3. Hierarchical Feature Representation

We first propose a hierarchical structure for event and interaction representations. In contrast to [18], the geometric relations are characterized upon the nearby STIP features (events) rather than the dense-sampled local observations to facilitate more efficient processing.

3.1. Low Level: Multi-Scale Event Representation

Since any event cannot happen without dynamic, an STIP feature is used to represent an event. We use the STIP detector proposed by Dollar *et al*. [10]. It utilizes two separate filters in the spatial and temporal directions: 2D Gaussian filter in space and 1D Gabor filter in time. To handle events with different scales due to the camera perspective distortion, a two-level Gaussian video pyramid is built from input video. Depending on the scenario, we empirically chose an appropriate descriptor from the interest point response (IPR) [10], 3DSIFT [21] and the 3D extensions of HOG [8] and HOF [9].

We next attempt to build a normal model to handle inter- and intra-class variations. This is done by quantizing normal events into a visual vocabulary $\mathcal{C}$ using the k-means algorithm based on the Euclidean metric. The pattern similarity of each interest point is based on the k-nearest neighbors (*k*-NN) distance with respect to the visual vocabulary $\mathcal{C}$ given by

$$y_i^l = \frac{1}{k} \sum_{\mathbf{c}_j \in \mathcal{C}_i} ||\mathbf{d}_i - \mathbf{c}_j||_2 \tag{1}$$

where $\mathcal{C}_i \subseteq \mathcal{C}$ is the subset of the top-k nearest codewords for the interest point STIP_i and $\mathbf{d}_i$ is its feature vector. The *k*-NN-based detector is simple but predictable. Abnormal events with strange appearances and unusual motions can then, efficiently and effectively, be detected by a user-specified threshold.

3.2. High Level: Ensemble of STIP features

To acquire the possible interactions in videos, we densely slide a 3D window over the video space with a 10-pixel sampling step to obtain the ensembles of the nearby STIP features given by

$$E_k = \{(\mathbf{v}_i, y_i^l, \mathcal{C}_i) | \forall \text{STIP}_i \in R_k\} \tag{2}$$

where R_k denotes the spatio-temporal neighborhood around the center. For each interest point $\text{STIP}_i \in R_k$, its relative location $\mathbf{v}_i \in \mathbb{R}^3$, its *k*-NN distance y_i^l, and the subset of the matched codewords $\mathcal{C}_i \subseteq \mathcal{C}$ are stored.

There are ensembles containing only few STIPs or nothing. Since we emphasize the interaction between multiple events, we enforce a quality control on ensembles to filter out such ensembles and accelerate the processing in the next stage. The quality function of an ensemble is defined as the area ratio of cuboid volumes $\mathcal{V}(\text{STIP}_i)$ to the ensemble volume $\mathcal{V}(E_k)$:

$$q(E_k) = \frac{\bigcup_{\forall \text{STIP}_i \in R_k} \mathcal{V}(\text{STIP}_i)}{\mathcal{V}(E_k)} \tag{3}$$

To efficiently calculate the union volume of cuboids, we adopt the computation of high-dimensional image integral technique in [22]. Suppose there is a volumetric mask which flags coverages of all cuboids found in the input video. Its 3D integral image is denoted by I_C. Eight corner locations of the ensemble E_k is denoted by $\{x^p | p \in \{0,1\}^3\}$. The quality function in Eq. 3 can then be computed by

$$q(E_k) = \frac{\sum_{p \in \{0,1\}^3} (-1)^{3-||p||_1} I_C(x^p)}{\mathcal{V}(E_k)} \tag{4}$$

Each successive computation of quality function reduces to $O(1)$ at the cost of the first acquirement of I_C. Note that we consider local and global anomalies individually. That is, we exclude the anomalous interest points detected by the local anomaly detector after this step as we emphasize the interaction analysis of normal events.

4. High-level Codebook Construction

To find the frequent geometric relations of the nearby STIP features from training videos, we cluster these qualified ensembles to acquire a high-level codebook of implicit interaction templates. Specifically, given a set of qualified ensembles, we aim to assign the ensembles into k sets $S = \{S_1, ..., S_k\}$ so as to minimize the within-cluster distance function given by

$$J = \min_{S,k} \sum_{i=1}^{k} \sum_{E_j \in S_i} sim(E_j, \mathcal{E}_i) \tag{5}$$

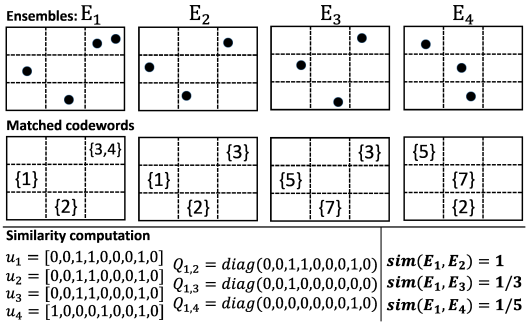


Figure 3: **A spatial example of measuring similarity**: We partition the ensemble space into 3-by-3 regions. Four different spatial relations of STIPs (black dots) and their matched codewords are shown. Ensembles $E1$ and $E2$ share the similar semantic and structural relationships while $E1$ and $E3$ only have similar structural relationships and $E1$ and $E4$ are quite different.

where $\mathcal{E}_i$ is the representative ensemble in S_i.

Note that the ensemble topology here is represented by STIPs, which is contrary to the vector form in [18]. The major advantage is that the resulting centroids can be deformable by agglomerating the ensembles of the same cluster rather than calculating the within-cluster mean vector.

4.1. Semantic and Structural Similarities

A similarity measurement of two ensembles is required for clustering. We employ a two-phase strategy for computational efficiency. It begins with partitioning an ensemble space into n_r 3D subregions. We then compute the topology similarity based on a newly defined co-occurrence statistics:

$$sim(E_k, E_l) = \frac{\mathbf{u}_k^T \mathbf{Q}_{k,l} \mathbf{u}_l}{||\mathbf{u}_k + \mathbf{u}_l||_1 - \mathbf{u}_k^T \mathbf{u}_l} \tag{6}$$

where the location occurrence $\mathbf{u}_k$ for an ensemble E_k is an $n_r \times 1$ binary vector in which every entry indicates whether any STIP exists in the corresponding subregion; the label co-occurrence matrix $\mathbf{Q}_{k,l}$ is an $n_r \times n_r$ binary diagonal matrix in which the ith diagonal entry indicates whether any pair of the matched codewords from ensembles E_k and E_l coincides in the ith subregion. Fig. 3 demonstrates the similarity computation in the spatial domain.

4.2. Bottom-Up Greedy Clustering

As the ensemble quantity grows substantially in proportion to the size of training data, it is advantageous to adopt a bottom-up procedure for large datasets to reduce the time and memory requirements. Algorithm 1 shows a greedy approach which sequentially updates an ever-growing codebook $\mathcal{E}$ once a qualified ensemble E_k is available. Based on the dataset, we set the similarity threshold $T_s \in [0.4, 0.6]$ so

Algorithm 1 Clustering ensembles of STIP features

Input: E_k (a qualified ensemble)
Output: $\mathcal{E} = \{\mathcal{E}_i\}$ (a codebook of interaction templates)

```
s = max_i sim(E_k, E_i)                          ▷ using Eq. 6
i* = arg max_i sim(E_k, E_i)
if s > T_s then
    if q(E_{i*}) ≤ T_q then                      ▷ using Eq. 4
        E_{i*} = E_{i*} ∪ E_k
    end if
else
    add new template E_k to E
end if
```

that every templates have uniform amount of members. The agglomeration procedure collects STIPs from the matched ensembles to form a more informative one.

We also prune noise for each template by discarding subregions with lower support (i.e., number of interest points). In addition, we enforce quality control by using Eq. 4 in order to avoid templates with unbalanced amount of data. The low-support suppression and quality control mechanisms are straightforward but effective. Fig. 8 shows that templates are more compact and distinguishable through these mechanisms. After we apply the quality control, we can find that the first template drastically discriminates the first test ensemble with the others. This is because the number of STIPs in each template is balanced. Moreover, the misclassification rate of abnormal ensembles can be significantly reduced by using the low-support suppression.

5. GPR-based Global Anomaly Detection

We next formulate each template in $\mathcal{E}$ as a *k*-NN regression problem and construct a model using GPR for learning and inferring, as shown in Fig. 4. The details are delineated in the following subsections.

5.1. GPR Model Learning

For a specific template, let $\mathbf{V} = \{\mathbf{v}_i \in \mathbb{R}^3 | i = 1, ..., n\}$ be a sequence of relative positions of STIPs. Let *k*-NN distances $\mathbf{y} = \{y_i^l \in \mathbb{R} | 1, ..., n\}$ serve as the target values. The goal of GPR is to learn the mapping from inputs $\mathbf{V}$ to the continuous observable targets $\mathbf{y}$. Assume the target vector $\mathbf{y}$ follows a zero-mean Gaussian prior. According to [17], the predictive distribution on $\mathbf{f}_* = \{f(\mathbf{v}_*^{(i)}) \in \mathbb{R} | i = 1, ..., n_*\}$ at test locations $\mathbf{V}_* = \{\mathbf{v}_*^{(i)} \in \mathbb{R}^3 | i = 1, ..., n_*\}$ is a multivariate Gaussian distribution given by

$$\mathbf{f}_* | \mathbf{V}, \mathbf{y}, \mathbf{V}_* \sim \mathcal{N}(\bar{\mathbf{f}}_*, \mathbb{V}(\mathbf{f}_*)) \tag{7}$$

where $\bar{\mathbf{f}}_* = \mathbf{K}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I}_n)^{-1} \mathbf{y}$ and $\mathbb{V}(\mathbf{f}_*) = \mathbf{K}_{**} - \mathbf{K}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I}_n)^{-1} \mathbf{K}_*$, in which $\mathbf{I}_n$ is an $n \times n$ identity matrix, and $K(\mathbf{V}, \mathbf{V})$, $K(\mathbf{V}, \mathbf{V}_*)$, and $K(\mathbf{V}_*, \mathbf{V}_*)$, denoted by $\mathbf{K}$, $\mathbf{K}_*$,

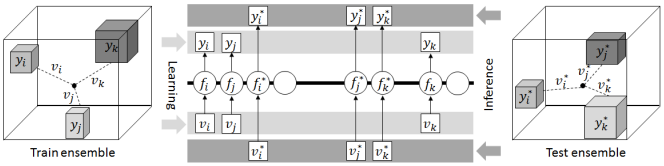


Figure 4: **Implicit interaction model learning and inferring**: In the middle, squares represent observed variables and circles represent model prediction. The thick horizontal bar represents a set of fully connected nodes. For each normal template, its topology are formulated as a *k*-NN regression problem for Gaussian process regression. Global anomaly detection is to measure the semantic and structural similarities of a test ensemble w.r.t. GPR models.

and $\mathbf{K}_{**}$, respectively, are the covariance matrices evaluated based on a predefined kernel function.

We mainly use the radial basis function (RBF) kernel, $k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp(-0.5||\mathbf{x} - \mathbf{x}'||_2^2/l^2)$, to relate predictions at nearby locations with each other. To handle noisy observations, additive identical, independent distributed Gaussian noise with variance σ_n^2 is imposed on each of observations in the training set. Therefore, the hyper-parameters in the RBF kernel include the length-scale l and the signal variance σ_f^2, and the noise variance σ_n^2 which can be estimated by minimizing the negative log marginal likelihood with respect to the hyper-parameters using the conjugate gradient optimizer. After the learning process, the GPR model for each template records the training data and the learned hyper-parameters, *i.e.*, $D = \{\mathbf{V}, \mathbf{y}, l, \sigma_f, \sigma_n\}$).

While GPR can deal with missing or noisy data, it is, however, ill-conditioned when training data involve STIPs that are low-support to an event (e,g., STIPs from dynamic background). As GPR uses the entire training set including the low-support STIPs in the learning process, these STIPs inevitably impact the performance of GPR prediction. Besides, comparing a test sample to each of the GPR models, with very unbalanced numbers of training data, may be unfair, since it will favor the one with more training data.

To tackle the above problems, the clustering procedure in Sec. 4 has taken these into consideration. Moreover, observed ensembles with similar structure are agglomerated altogether so that STIPs within the merged ensemble are arranged in a deformable configuration.

5.2. GPR Model Inference

In this section, we describe how to infer the likelihood of a test sample with respect to a GPR model using a probabilistic framework. Given an observed sample $E_* = (\mathbf{V}_*, \mathbf{y}_*)$, the likelihood to a specific GPR model D_i

Algorithm 2 Marginal Inference Algorithm

Input: $D_i = \{\mathbf{V}, \mathbf{y}, l, \sigma_f, \sigma_n\}$ (GPR model), $E_* = (\mathbf{V}_*, \mathbf{y}_*)$ (test ensemble), k (kernel function)
Output: $-\log p(\mathbf{y}_*|\mathbf{V}_*, D_i)$
GPR-PREDICTION$(\mathbf{V}, \mathbf{y}, k, \sigma_f^2, \sigma_n^2, \mathbf{V}_*)$
$\mathbf{L}_* := \text{cholesky}(\mathbb{V}(\mathbf{f}_*) + \sigma_n^2 \mathbf{I}_{n_*})$ ▷ Cholesky decompose
$\mathbf{v} := \mathbf{L}_* \backslash \mathbf{y}_*$
$\mathbf{u} := \mathbf{L}_* \backslash \bar{\mathbf{f}}_*$
return $\frac{1}{2}\mathbf{u}^T\mathbf{u} + \frac{1}{2}\mathbf{v}^T\mathbf{v} - \mathbf{v}^T\mathbf{u} + \sum_i \log[\mathbf{L}_*]_{ii} + \frac{n}{2}\log 2\pi$

function GPR-PREDICTION$(\mathbf{V}, \mathbf{y}, k, \sigma_f^2, \sigma_n^2, \mathbf{V}_*)$
$\mathbf{L} := \text{cholesky}(\mathbf{K} + \sigma_n^2 \mathbf{I}_n)$
$\boldsymbol{\alpha} := \mathbf{L}^T \backslash (\mathbf{L} \backslash \mathbf{y})$
$\bar{\mathbf{f}}_* := \mathbf{K}_*^T \boldsymbol{\alpha}$
$\mathbf{W} := \mathbf{L} \backslash \mathbf{K}_*$
$\mathbb{V}(\mathbf{f}_*) := \mathbf{K}_{**} - \mathbf{W}^T\mathbf{W}$
return $\bar{\mathbf{f}}_*, \mathbb{V}(\mathbf{f}_*)$
end function

is defined by the marginal probability:

$$p(\mathbf{y}_*|\mathbf{V}_*, D_i) = \int p(\mathbf{f}_*|\mathbf{V}_*, D_i) p(\mathbf{y}_*|\mathbf{f}_*) d\mathbf{f}_* \tag{8}$$

where the possibility $p(\mathbf{f}_*|\mathbf{V}_*, D_i)$ accounts for the positional distribution and $p(\mathbf{y}_*|\mathbf{f}_*)$ captures the appearance similarity. The inference can jointly consider how likely the semantic and structural relationships in the test ensemble belong to the GPR model.

In Eq. 8, we use GPR to model the first term which yields a multivariate Gaussian distribution. As for the similarity term $p(\mathbf{y}_*|\mathbf{f}_*)$, there are many choices. Kim *et al*. [11] ignored the semantic similarity term as they focused on the motion trajectory. Alternatively, Rasmussen and Williams [17] used a zero-mean Gaussian assumption and showed that the integral boils down to a multivariate Gaussian distribution. It is, however, an inappropriate suggestion in our case because no prior information learned from a GPR model is used to model $p(\mathbf{y}_*|\mathbf{f}_*)$. Therefore, we augment the Gaussian assumption by incorporating the prediction results given by

$$\mathbf{y}_*|\mathbf{f}_* \sim \mathcal{N}(\mathbf{f}_*, \sigma_n^2 \mathbf{I}_{n_*}) \tag{9}$$

If we assume that the pattern residuals $\boldsymbol{\epsilon} = \mathbf{y}_* - \mathbf{f}_*$ follow an independent, identical Gaussian distribution with variation σ_n^2, Eq. 8 becomes an integral of Gaussian product. Substituting this into Eq. 8 results in

$$\begin{aligned} p(\mathbf{y}_*|\mathbf{V}_*, D_i) = & \frac{1}{(2\pi)^n \sqrt{|\mathbb{V}(\mathbf{f}_*)||\sigma_n^2 \mathbf{I}_{n_*}|}} \\ & \cdot \int \exp\Big[-\frac{1}{2}(\mathbf{f}_* - \bar{\mathbf{f}}_*)^T \mathbb{V}(\mathbf{f}_*)^{-1}(\mathbf{f}_* - \bar{\mathbf{f}}_*) \\ & - \frac{1}{2}(\mathbf{y}_* - \mathbf{f}_*)^T (\frac{1}{\sigma_n^2}\mathbf{I}_{n_*})(\mathbf{y}_* - \mathbf{f}_*)\Big] d\mathbf{f}_* \end{aligned} \tag{10}$$

By making use of the general case of Sylvester's determinant theorem and Woodbury inversion lemma, the log likelihood of Eq. 10 can be simplified as

$$\log p(\mathbf{y}_*|\mathbf{V}_*, D_i) = -\frac{1}{2}\bar{\mathbf{f}}_*^T \mathbf{\Sigma}_*^{-1}\bar{\mathbf{f}}_* - \frac{1}{2}\mathbf{y}_*^T\mathbf{\Sigma}_*^{-1}\mathbf{y}_* + \mathbf{y}_*^T\mathbf{\Sigma}_*^{-1}\bar{\mathbf{f}}_* - \frac{n}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{\Sigma}_*| \tag{11}$$

where $\mathbf{\Sigma}_* := \mathbb{V}(\mathbf{f}_*) + \sigma_n^2\mathbf{I}_{n_*}$. A practical implementation of Eq. 11 is shown in Algorithm 2. We replace the matrix inversion with the Cholesky decomposition for faster and numerically stable computation. In case of failure in the Cholesky decomposition, we relax the input dependences by discarding the off-diagonal entries in $\mathbf{\Sigma}_*$.

The computations in Algorithm 2 are mainly contributed by matrix multiplication. Since the L and $\boldsymbol{\alpha}$ can be precomputed in the training period, the overall running time takes approximately $O(n^2 n_*)$ provided that $n >> n_*$.

5.3. Global Anomaly Detection

We next calculate the likelihood of a test ensemble with respect to the GPR models. The global negative log likelihood (GNLL) of a test ensemble against the kth model is defined as the average on the point-wise negative log likelihoods given by

$$\mathcal{G}_k(E_*) = -\frac{1}{n_*}\sum_{i=1}^{n_*}\log p(y_*^{(i)}|v_*^{(i)}, D_k) \tag{12}$$

The nearest neighbor strategy is then invoked to choose the best-matched GPR model:

$$k^* = \arg\min_k \mathcal{G}_k(E_*). \tag{13}$$

To precisely locate abnormal events, each STIP within the test ensemble is assigned with its local negative log likelihood (LNLL) w.r.t. the best-matched GPR model:

$$y_i^h = -\log p(y_*^{(i)}|v_*^{(i)}, D_{k^*}), \forall \text{STIP}_i \in R_*. \tag{14}$$

For point-wise likelihood evaluation, most of the matrix manipulations in Algorithm 2 reduce from polynomial to linear time. Though the computation order remains unchanged, a salient speedup is perceived in practice. The overall computational time reduces to $O(n_* \sum_{k=1}^{N} n_k^2)$ where n_k is the number of STIPs in the kth GPR model. Fig. 5 computes the likelihoods of three test cases on the *Subway* dataset where a large-scale ensemble is adopted to monitor short-term clips of videos.

To combine the results from local and global anomaly detectors, the weighted sum is applied:

$$\hat{y}_i = \alpha\hat{y}_i^l + (1-\alpha)\hat{y}_i^h \tag{15}$$

where $\alpha \in [0, 1]$ is the preference factor and $\hat{y}_i^l$ and $\hat{y}_i^h$ are the standard scores of y_i^l and y_i^h as defined in Eq. 1 and Eq. 14, respectively.

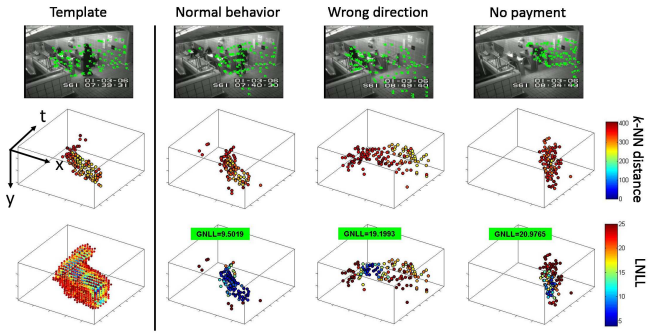


Figure 5: **Visual example of global anomaly detection**: A learned GPR model is shown in the first column while test behaviors are shown in the remaining columns. We intentionally use the rotation-invariant 3DSIFT descriptor such that these behaviors cannot be distinguished solely using their patterns (k-NN distances) (the second row) unless the positional information (the third row) is considered.

6. Experimental Results

We apply the proposed method to three public real-world datasets: the UCSDped1 [14], Subway [1], QMUL Junction [12] datasets, as shown in Fig. 6. Tab. 1 describes the datasets and the percentages of videos we used for training and testing (GT indicates whether an official ground truth is provided (Y) or not (N)). The challenge here is to understand dominant behaviors and identify irregular events and interactions in crowded scenarios which suffer from partial occlusion and scale variation.

We mainly use the pixel-level and the frame-level protocols [14] to evaluate our work. In the frame-level criterion, an observed frame is considered true positive if both of the frame and its ground truth detect anomalies regardless of the location. In the pixel-level criterion, a true positive is hit when a frame coincides with its ground truth in which at least 40% of co-located pixels are identified. ROC curves are plotted by imposing multiple thresholds on detection results. We quantify the performances in terms of the *equal error rate* (EER) [14] and *area under curve* (AUC) [13].

To speed up the analysis, we down-sample videos to a 238×158 resolution at a frame rate of 5 fps. We choose a cuboid size of $9 \times 9 \times 9$ for the UCSDped1 dataset, and $13 \times 13 \times 13$ for other datasets as close to objects as possible. Depending on the scenarios, we adopt an ensemble of size $41 \times 41 \times 21$ which is affordable for speed or appearance anomalies in the UCSDPed1 dataset. For other datasets, we apply large-scale ensembles covering a whole frame with a duration of about 10 seconds to monitor video segments with a hope to understand global behaviors. We conduct experiments using different features and empirically use 3DSIFT, HOF, HOG on the UCSDped1, Subway,

Table 1: Dataset Description

Dataset	Scenario	GT	Length	Train	Test
UCSDped1 [14]	walkway	Y	14000 frames	41%	59%
Subway [1]	subway	Y	96 minutes	53%	47%
QMUL Junction [12]	intersection	N	60 minutes	34%	66%

Figure 6: **Dataset snapshots and detection results**: Abnormal events of UCSDped1, Subway, and QMUL Junction are arranged in the first, second, and third rows, respectively. Local anomalies are indicated with orange boundaries and unusual interactions with red boundaries. The detected anomalies of our method are marked with red regions.

and QMUL Junction datasets, respectively.

6.1. Effect of Data Pruning and Balance

In this subsection, we assess the proposed filtering scheme described in Sec. 4.2 based on the UCSDped1 dataset. We can note from Fig. 8 that by averaging the off-diagonal entries in the confusion matrices as a measure of noise, the proposed GPR method without using the data balance and pruning schemes (the leftmost matrix in Fig. 8) can have a noise level of 16.74%. By putting the mentioned schemes all together (the rightmost matrix in Fig. 8), the noise level can then be suppressed from 16.74% to 4.32%.

6.2. Comparison With State-of-the-Art Methods

In this subsection, we compare our method with some previous works including the MDT[14], the OptiFlow Stat [1], the Local kNN[19], the Sparse Recon [6], the STC [18], and the IBC [3] methods. For clarity, we use the prefixes *Dense* or *Sparse* in STC [18] and IBC [3] to emphasize that their models are used to characterize the relationships of densely-sampled or the STIP features provided by our local anomaly detector, respectively.

The simulations based on the three data sets are shown in Fig. 7 and summarized in Tab. 2, and will be elaborated more in the following subsections. Also, to evaluate the impact of the GPR model modeling, the proposed method with and without the GPR mdeling, referred to as the GPR and the Sparse Cuboids methods respectively, are both provided for comparison.

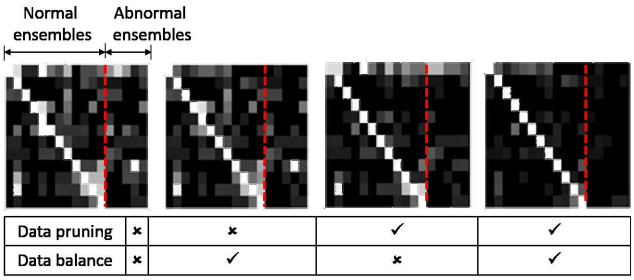


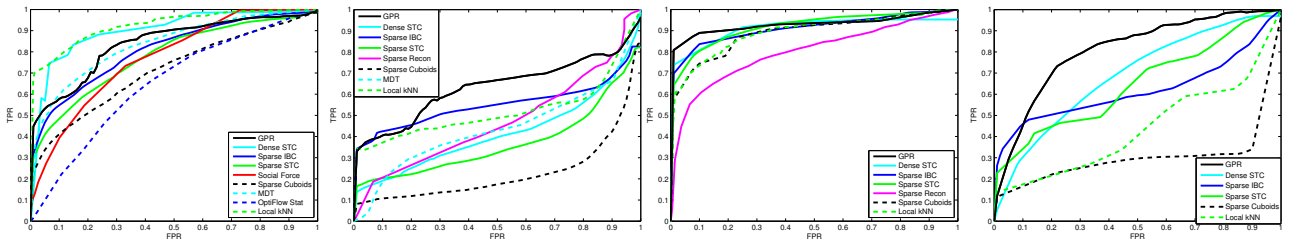
Figure 8: **Effect of data pruning and balance**: In each confusion matrix, the rows indicate 12 learned templates, and the first 12 columns are the normal interactions selected from members of each templates in the corresponding rows, and the last 5 columns are the abnormal interactions from the UCSDped1 dataset. For better visualization, we normalize the topology similarities of each template from low (black) to high (white). The bottom table provides an index for each confusion matrix.

6.2.1 The UCSDped1 dataset

We first evaluate the proposed method based on the UCSDped1 dataset. From the ROC curves shown in Fig. 7a, we can note that the proposed Sparse Cuboids scheme relies on the multi-scale STIP detection so that local anomalies (*e.g.* biker, car, and skater) with different scales (*e.g.* small-scale wheelchair) can well be detected with 72.2% AUC. Together with the GPR model, our proposed method can outperform the other methods by an average of 6.8% AUC. Since we consider the nearby STIPs, the local anomalies ignored by Sparse Cuboids are likely to be identified. Compared to methods using dense features like Dense STC and Local kNN, our method based on sparse features achieves competing results, but with much lower processing time. Moreover, our GPR method can precisely locate abnormal events with 63.3% AUC, as shown in Fig. 7b. The STC method degrades in the localization rate since they treat an ensemble of densely-sampled patches or STIPs as an atomic unit and cannot identify whether each of local observations is abnormal or not. Surprisingly, unnoticed events like *two man talk* and *suddenly turn left* are detected as well, which have unusual interactions but may not be considered as abnormal.

6.2.2 The Subway Dataset

Next, we compare the proposed method with the other aforementioned methods based on the Subway dataset.



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

Figure 7: **ROC curves of different methods**: Methods with solid curves focus on modeling the relationships between neighbors. Results from the UCSDped1, Subway, and QMUL Junction datasets based on the frame-level criterion are shown in (a), (c), (d), respectively. Results from the UCSDped1 dataset based on the pixel-level criterion is shown in (b).

Table 2: Comparisons with other methods

Methods	GPR	Sparse Cuboids	Sparse IBC [3]	Sparse STC [18]	Dense STC [18]	Local kNN [19]
UCSDped1	23.7/83.8[a]	34.2/72.2	28.2/80.8	30.2/77.9	16.0/89.9	**16.0/92.7**
Subway	**10.9/92.7**	20.0/88.9	15.0/91.0	15.7/91.7	15.3/91.1	18.4/89.1
QMUL Junction	**24.6/80.9**	69.2/27.3	42.7/61.8	42.7/64.5	36.4/68.7	54.3/43.6
UCSDped1	**37.3/63.3**[b]	75.0/22.5	45.8/54.9	62.6/36.8	57.7/41.7	51.0/52.0

[a]Each entry indicates EER(%)/AUC(%) on the frame-level criterion
[b]Each entry indicates EER(%)/AUC(%) on the pixel-level criterion

From the ROC curves shown in Fig. 7c, we can find that our GPR method achieves the highest detection rate with 92.7% AUC while our Sparse Cuboids method can sufficiently detect the *no payment* and *wrong direction* events. The improvement testifies the robustness of the GPR model to noise by considering the nearby noise-prone optical flows.

6.2.3 The QMUL Junction Dataset

Finally, we compare the proposed method with the other two closely related methods: the STC [18] and the IBC [3] methods based on the QMUL Junction dataset. From the ROC curves shown in Fig. 7d, we can observe that our method outperforms the IBC [3] and STC [18] approaches by at least 12% AUC. This is due to the fact that the parametric model in the Sparse IBC approach requires to initialize the covariance matrices while the Sparse STC method may encounter the curse of dimensionality when approximating an ensemble topology. Our Sparse Cuboids method cannot provide satisfactory results as accidents like jaywalking and traffic interruption usually involve with multiple events.

6.3. Computational Complexity

We compare the computational time of our model with the STC [18] and the IBC [3] methods based on the UCSDped1 dataset. All of the methods are implemented in the MATLAB environment on a computer with Core i7-2600 CPU and 4GM RAM. No particular programming technique is used, except our method is using the GPML toolbox [17]. As shown in Tab. 3, the high-level codebook construction takes 65.4% of the entire learning time. The processing time of Sparse Cuboids contains interest point detection, feature extraction, vector quantization, and k-NN computation. Our method takes approximately 0.5 seconds for inference, which is time-affordable as there are 300 or so ensembles per test video. For the Sparse IBC method, it requires less learning time at the expense of significant inference time (9 seconds). The Sparse STC is efficient but its dense version requires about five times of the running time required by our method.

Table 3: Computational Time (ms per train/test ensemble)

Models		Learning	Inferring
GPR	Sparse Cuboids	18.4	6.4
	Ensembles acquirement	91.7	88.6
	High-level codebook construction	29.5	-
	Hyper-parameter estimation	0.6	-
	GNLL computation	-	420.3
	Total	140.2	515.3
Sparse IBC [3]		3.6	9818.3
Sparse STC [18]		139.8	96
Dense STC [18]		2432.5	2424.1

7. Conclusions

This paper provides a hierarchical framework for local and global anomalies detection. We rely on a greedy method and Gaussian process regression to cluster, learn, and infer the semantic (appearance) and structural (position) relationships of the nearby STIPs. Our method achieves at least 80% detection rate based on the three challenging datasets and provides competing performance compared with previous works that characterize the relationships of densely-sampled patches while maintaining much lower space and time complexity.

Acknowledgment

This work was supported by the Minister of Science and Technology, R.O.C. under contract MOST 103-2221-E-011-117.

References

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

[2] Y. Benezeth, P.-M. Jodoin, and V. Saligrama. Abnormality detection using low-level co-occurring events. *Pattern Recognit. Lett.*, 32:423 – 431, Feb. 2011.

[3] O. Boiman and M. Irani. Detecting irregularities in images and in video. *Int. J. Comput. Vis.*, 74:17 – 31, Aug. 2007.

[4] A. Chan and N. Vasconcelos. Modeling, clustering, and segmenting video with mixtures of dynamic textures. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30:909 – 926, May 2008.

[5] K.-W. Cheng, Y.-T. Chen, and W.-H. Fang. Abnormal crowd behavior detection and localization using maximum subsequence search. In *ACM/IEEE Int. Workshop Anal. Retr. Track. Event Motion Imag. Stream*, pages 49 – 58, Oct. 2013.

[6] Y. Cong, J. Yuan, and J. Liu. Sparse reconstruction cost for abnormal event detection. In *Conf. Comput. Vis. Pattern Recognit.*, pages 3449 – 3456, June 2011.

[7] X. Cui, Q. Liu, M. Gao, and D. N. Metaxas. Abnormal detection using interaction energy potentials. In *Conf. Comput. Vis. Pattern Recognit.*, pages 3161 – 3167, June 2011.

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Conf. Comput. Vis. Pattern Recognit.*, pages 886 – 893, June 2005.

[9] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *Eur. Conf. Comput. Vis.*, pages 428–441, May 2006.

[10] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *IEEE Int. Workshop Vis. Surveill. Perform. Eval. Track. Surveill.*, pages 65–72, Oct. 2005.

[11] K. Kim, D. Lee, and I. Essa. Gaussian process regression flow for analysis of motion trajectories. In *Int. Conf. Comput. Vis.*, pages 1164 – 1171, Nov. 2011.

[12] C. C. Loy, T. Xiang, and S. Gong. Modelling multi-object activity by gaussian processes. In *Br. Mach. Vis. Conf.*, pages 1–11, 2009.

[13] C. Lu, J. Shi, and J. Jia. Abnormal event detection at 150 fps in matlab. In *Int. Conf. Comput. Vis.*, pages 2720 – 2727, Dec. 2013.

[14] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos. Anomaly detection in crowded scenes. In *Conf. Comput. Vis. Pattern Recognit.*, pages 1975 – 1981, June 2010.

[15] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *Conf. Comput. Vis. Pattern Recognit.*, pages 935–942, June 2009.

[16] O. Popoola and K. Wang. Video-based abnormal human behavior recognitionxa review. *IEEE Trans. Syst. Man Cybern. Soc.*, pages 865–878, Nov. 2012.

[17] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[18] M. J. Rohstkhari and M. D. Levine. Online dominant and anomalous behavior detection in videos. In *Conf. Comput. Vis. Pattern Recognit.*, pages 2611 – 2618, June 2013.

[19] V. Saligrama and Z. Chen. Video anomaly detection based on local statistical aggregates. In *Conf. Comput. Vis. Pattern Recognit.*, pages 2112 – 2119, June 2012.

[20] B. Scholkopt, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson. Estimating the support of a high-dimensional distribution. *Neural Comput.*, 13:1443–1471, July 2001.

[21] P. Scovanner, S. Ali, and M. Shah. A 3-dimensional sift descriptor and its application to action recognition. In *ACM Multimed.*, pages 357–360, Sept. 2007.

[22] E. Tapia. A note on the computation of high-dimensional integral images. *Pattern Recognit. Lett.*, 32:197 – 201, Jan. 2011.

[23] D. Tran, J. Yang, and D. Forsyth. Video event detection: from subvolume localization to spatio-temporal path search. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36:404 – 416, July 2013.

[24] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30:283–298, Feb. 2008.

[25] T. Wang and H. Snoussi. Histograms of optical flow orientation for abnormal events detection. In *IEEE Int. Workshop Perform. Eval. Track. Surveill.*, pages 45 – 52, Jan. 2013.