Anomaly Detection in Video Sequence with Appearance-Motion Correspondence

Trong-Nguyen Nguyen, Jean Meunier
DIRO, University of Montreal
{nguyetn, meunier}@iro.umontreal.ca

Abstract

*Anomaly detection in surveillance videos is currently a challenge because of the diversity of possible events. We propose a deep convolutional neural network (CNN) that addresses this problem by learning a correspondence between common object appearances (*e.g. *pedestrian, background, tree, etc.) and their associated motions. Our model is designed as a combination of a reconstruction network and an image translation model that share the same encoder. The former sub-network determines the most significant structures that appear in video frames and the latter one attempts to associate motion templates to such structures. The training stage is performed using only videos of normal events and the model is then capable to estimate frame-level scores for an unknown input. The experiments on 6 benchmark datasets demonstrate the competitive performance of the proposed approach with respect to state-of-the-art methods.*

1. Introduction

Anomaly detection in video sequences is a necessary functionality for surveillance systems. Because abnormal events rarely occur in real-world videos, this task is significantly time-consuming and may require a large amount of resource (*e.g*. people) to perform manual checking. A method than can automatically determine potential frames of anomalous events is thus crucial.

Our model is a combination of a convolutional auto-encoder (Conv-AE) and a U-Net with skip connections [39] that share the same encoder sub-network. Other related works employed either an AE or a U-Net to perform the anomaly detection in different ways. Hasan *et al*. [11] estimate regularity score for frames in video sequences according to reconstruction models. Their two AEs (with and without convolutional layers) work on two different inputs: hand-crafted features (HOG and HOF with trajectory-based properties [49]) and concatenation of 10 consecutive frames along the temporal axis. The reconstruction error is used to indicate their regularity score. Unlike that work, the input of our Conv-AE is a single frame and the temporal factor is considered in the other stream via U-Net. The purpose of our Conv-AE is to learn only regular appearance structures.

On the contrary, Ravanbakhsh *et al*. [37] employ the U-Net structure proposed in [17] to translate an input from video frame to a corresponding optical flow and vice versa. We argue that the use of two CNNs with the same structure may be redundant and an appropriate modification and/or combination would improve the model ability. Compared with [37], our network keeps the stream translating a video frame to an optical flow (but using our proposed structure instead of [17]) while replaces the other U-Net by a Conv-AE that shares the encoding flow.

Inspired by the good performance of the video prediction model in [32], Liu *et al*. [25] present a model that uses a U-Net structure to predict a frame from a number of recent ones and then estimates the corresponding optical flow. The model is optimized according to the difference between the outputted and original versions of video frame as well as the optical flow together with an adversarial loss. Our work also predicts an optical flow but directly from a single frame in order to determine the association between a scene appearance and its typical motion. Since a fixed procedure of optical flow estimation (FlowNet [8]) is embedded inside the network in [25], the selection of such method is thus limited because the estimator has to be fully differentiable to perform an end-to-end training. Our model, however, has a stream that directly estimates a mapping from input frame to optical flow. We only use a pretrained estimator for ground truth calculation and the model signal does not propagate through it during the training as well as inference stages.

Our main contributions are summarized as follows:

- We design a CNN that combines a Conv-AE and a U-Net, in which each stream has its own contribution for the task of detecting anomalous frames. The model can be trained end-to-end.

- We integrate an Inception module modified from [48] right after the input layer to reduce the effect of network's depth since this depth is considered as a hyper-parameter that requires a careful selection.

- We propose a patch-based scheme estimating frame-level normality score that reduces the effect of noise which appears in the model outputs.

- Experiments on 6 benchmark datasets demonstrate the potential of our model with competitive performance compared with state-of-the-art methods. We also provide discussions for these datasets that should be useful for future works.

The remainder of this paper is organized as follows: a summary of related studies is given in Section 2; Section 3 describes the details of our method; experiments and discussions for the 6 benchmark datasets are presented in Section 4; and Section 5 concludes this work.

2. Related work

We briefly describe the principal categories that lead to very different approaches for anomaly detection in video.

2.1. Trajectory

The diversity of possible anomalous events is the main challenge of the anomaly detection problem. Some researchers simplify this issue by explicitly specifying anomalies (*e.g.* [45]) or particular relevant attributes that can be used effectively for anomaly detection, in which the most common one is motion trajectory. These studies aim to learn patterns of object trajectories determined from normal events [34, 3, 36, 53]. There are four main stages in the methodology including object detection, tracking, trajectory-based feature extraction and classification/detection. The advantages of methods in this category are the simple implementation and fast execution. However, their effectiveness may significantly degrade when working on videos with cluttered background since the trajectory determination depends on the result of object detection and tracking. Moreover, trajectory anomalies do not cover the whole spectrum of anomalies in video surveillance.

2.2. Sparse coding

Instead of explicitly defining and estimating specific anomaly attributes, other researchers consider an input sequence of frames as a collection of small 3D patches. Concretely, a number of consecutive frames are concatenated along the temporal axis and then split into same-size 3D patches according to a window sliding on the image plane. In the inference stage, each 3D patch extracted from unknown inputs is represented as a sparse combination of training samples of normal events. The reconstruction error is considered as the score supporting the final decision. Such sparsity-based methods have achieved state-of-the-art performances [6, 55]. The main drawback is the high computational cost in finding combination coefficients due to sparse representation. Some studies thus attempt to reduce the complexity by modifying the learning algorithms and/or data structures [26, 28]. Beside window-based split, 3D patches are also determined using keypoint detectors [5] while other researchers attempt to learn the relation between training patches according to their distribution [30] or graph-based representation [20].

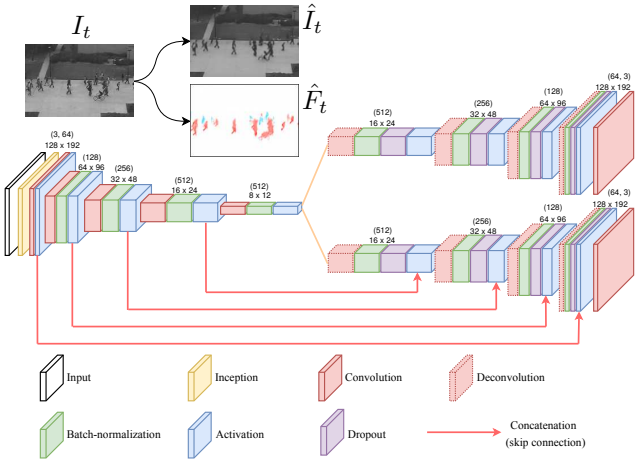


Figure 1. Overview of our model structure together with the spatial resolution of feature maps in each block (*i.e.* a sequence of layers with the same output shape). The number of channels corresponding to each layer in each block is also presented (in parentheses). The input and two output layers have the same size of $128 \times 192 \times 3$. There are three clusters of layers: common encoder (left), appearance decoder (top right) and motion decoder (bottom right). Each concatenation is performed along the channel axis right before operating the next deconvolution. The model input is a single video frame I_t and the outputs from the two decoders are a reconstructed frame $\hat{I}_t$ and an optical flow $\hat{F}_t$ predicting the motion between I_t and I_{t+1}. Best viewed in color.

2.3. Deep learning

Since deep learning models currently achieve top performance in a wide range of vision applications such as image classification [23, 47, 13], object detection [38, 12] and image captioning [18, 19], many CNNs have been proposed to deal with the problem of anomaly detection in videos. Typical structures of image reconstruction and translation are usually employed and the difference between their output and ground truth is used to indicate the frame-level score [11, 37, 25]. Some researchers apply pretrained classification models (such as VGG [41]) to extract useful features from input videos [42, 16]. Results of object detection and/or foreground estimation are also used for the determination of anomalous events in [14, 51].

3. Proposed method

An overview of our model is visualized in Figure 1. The model includes two processing streams. The first one is per-

formed via a Conv-AE to learn common appearance spatial structures in normal events. The second stream is to determine an association between each input pattern and its corresponding motion represented by an optical flow of 3 channels (xy displacements and magnitude). The skip connections in U-Net are useful for image translation since it directly transforms low-level features (*e.g*. edge, image patch) from original domains to the decoded ones. Such connections are not employed in the appearance stream because the network may let the input information go through these connections instead of emphasizing underlying attributes via the bottleneck.

Our model does not use any fully-connected layer, so it can theoretically work on images of any resolution. In order to simplify the model as well as make it be appropriate for possible further extensions, we fixed the size of input layer as $128 \times 192 \times 3$. The image size is set to a ratio of 1:1.5 instead of 1:1 as in related works (*e.g*. [11, 42, 25]) in order to preserve the aspect of objects in surveillance videos.

3.1. Inception module

The Inception module was originally proposed to let a CNN decide its filter size (in a few layers) automatically [47]. A number of convolutional operations with various filter resolutions are performed in parallel and the obtained feature maps are then concatenated along the channel axis. The use of this module in our work can be explained under an alternative perspective as follows. The proposed network has an encoder-decoder structure with bottleneck. A very deep architecture may eliminate the features that are helpful for decoding. On the contrary, a shallow network takes the risk of missing high-level abstractions. Therefore, we apply an Inception module to let the model select its appropriate convolutional operations.

This work focuses on surveillance videos acquired from a fixed position. Given a convolutional layer with a predefined receptive field (*i.e*. filter size) right after the input layer, the information abstraction would be different for the same object captured at various distances. This property is propagated for next layers, we thus expect the model to early determine low-level features by putting the Inception module right after the input layer. We remove the max-pooling in this module since the input is a regular video frame instead of a collection of feature maps. Our Inception module is modified from [48] including 4 streams of convolutions of filter sizes 1×1, 3×3, 5×5 and 7×7. Each convolutional layer of filter larger than 1×1 is factorized into a sequence of layers with smaller receptive fields in order to reduce the computational cost as suggested in [48].

3.2. Appearance convolutional autoencoder

Our Conv-AE supports the detection of strange (abnormal) objects within input frames by learning common appearance templates in normal events. This sub-network consists of the encoder and the top decoder without any skip connection as shown in Figure 1. The encoder is constructed by a sequence of blocks including triple layers: convolution, batch-normalization (BatchNorm) and leaky-ReLU activation [29]. The first block (right after the Inception module) does not contain BatchNorm layer as suggested in [17] for our U-Net task in Section 3.3. Instead of using pooling layer to reduce the resolution of feature maps, we apply strided convolution. Such parametric operation is expected to support the network finding an informative way to downsample the spatial resolution of feature maps as well as learning the further upsampling in decoding stage [43].

The decoder is also a sequence of layer blocks that increases the spatial resolution while reduces the number of feature maps after each deconvolution layer. A dropout layer (with $p_{drop} = 0.3$) is attached before the ReLU activation in each block as a regularization that reduces the risk of overfitting during the training stage [44].

Since the Conv-AE is to learn common appearance patterns of normal events, we consider the l_2 distance between the input image I and its reconstruction $\hat{I}$. The model thus forces to produce an image with similar intensity for each pixel. The intensity loss is estimated as

$$\mathcal{L}_{int}(I, \hat{I}) = \|I - \hat{I}\|_2^2 \tag{1}$$

A drawback of using only l_2 loss is the blur in the output, we thus add a constraint that attempts to preserve the original gradient (*i.e*. the sharpness) in the reconstructed image. The gradient loss is defined as the difference between absolute gradients along the two spatial dimensions as

$$\mathcal{L}_{grad}(I, \hat{I}) = \sum_{d \in \{x,y\}} \left\| \left| g_d(I) \right| - \left| g_d(\hat{I}) \right| \right\|_1 \tag{2}$$

where g_d denotes the image gradient along the d-axis. The final loss function of the appearance Conv-AE is formed as a summation of the intensity and gradient losses.

$$\mathcal{L}_{appe}(I, \hat{I}) = \mathcal{L}_{int}(I, \hat{I}) + \mathcal{L}_{grad}(I, \hat{I}) \tag{3}$$

This loss combination has been reported to give good performance for the task of video prediction [32, 25].

3.3. Motion prediction U-Net

Beside the appearance of strange object structures, unusual motions of typical objects would also be appropriate to provide an assessment of a video frame. Recall that each block in the encoder is to emphasize spatial abstractions of common objects within training frames. Our U-Net subnetwork thus focuses on learning the association between such patterns and corresponding motions. The ground truth

optical flow employed in this work is estimated by a pretrained FlowNet2 [15]. Compared with related models, the optical flow outputted from FlowNet2 is not only much smoother but also preserves motion discontinuities with sharper boundaries. The motion stream is expected to associate typical motions to common appearance objects while ignoring the static background patterns.

The decoder of our U-Net has the same structure as the Conv-AE except for the skip connections. These concatenations are to combine the feature maps upsampled from a higher level of abstraction with the ones containing low-level details. The use of leaky-ReLU activation in the encoder also keeps weak responses that may be informative for the translation in the decoder.

Unlike the Conv-AE in Section 3.2, the loss between an outputted optical flow and its ground truth is measured by l_1 distance. There are two main reasons for this. First, the FlowNet2 model is formed as a fusion of multiple networks providing optical flows from coarse (noisy) to fine (smooth), the result might thus contain noise or even amplify noisy regions during the smoothing procedure. Second, because the selection of optical flow estimation is not limited to FlowNet2, the training ground truth obtained from other algorithms might therefore possibly have small patches of wrong and/or noisy motion measure. In order to reduce the effect of such outliers when learning the motion association, we apply l_1 distance loss

$$\mathcal{L}_{flow}(F_t, \hat{F}_t) = \|F_t - \hat{F}_t\|_1 \tag{4}$$

where F_t is the ground truth optical flow estimated from two consecutive frames I_t and I_{t+1}, and $\hat{F}_t$ is the output of our U-Net given I_t. In summary, this stream attempts to predict instant motions of objects appearing in the video.

3.4. Additional motion-related objective function

Beside the distance-based loss $\mathcal{L}_{flow}$, we also add another loss that penalizes the underlying distribution of predicted optical flow to be similar to ground truth. The generative adversarial network (GAN) [10] was originally introduced to allow a CNN learning an implicit distribution of patterns. The model consists of a generator that creates fake samples from noise and a discriminator that attempts to distinguish such outputs from the real patterns. Many modified GAN versions have been proposed for the task of data generation. The discriminator also plays the role of a regularization in many models. Inspired by [32] where using a GAN loss is reported to provide better results compared with employing only distance-based ones, we apply such strategy as an additional objective function.

Our generator is the entire network in Figure 1 while the discriminator conditionally performs the classification on predicted optical flow. A visualization of our discriminator architecture is shown in Figure 2. Notice that the discriminator is not employed in the inference stage. Although the recent study [25] employed a Least Square GAN [31] and achieved state-of-the-art performance in detecting anomalous video frames, our model follows the strategy of typical conditional GAN (cGAN) where both the ground truth video frame and its corresponding optical flow are fed into the discriminator. There are two reasons leading to this decision. First, the cGAN theoretically avoids the problem of mode collapse in vanilla GAN since ground truth information (*i.e.* labels, real samples) is fed into the discriminator. The model is thus expected to efficiently learn the distribution of training samples. Second, cGAN is appropriate for a CNN of image translation as demonstrated in [17].

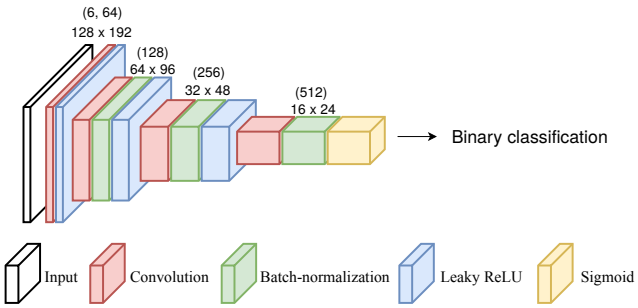


Figure 2. The architecture of our discriminator. The input layer of shape $128 \times 192 \times 6$ is fed by the concatenation of a video frame and its optical flow (that is either ground truth or outputted from the U-Net). The output layer is sigmoid activation of 512 feature maps of spatial resolution 16×24. Best viewed in color.

Finally, the adversarial loss is directly computed on the last layer containing activated feature maps in the discriminator. This calculation is different from [17, 25] where a convolutional layer is employed to collapse previous feature channels into a 2D map. The common sense of our model and the two others is the structural penalization where the classification is performed according to image patches instead of the whole image. However, we strictly constrain patches at feature-level so that each feature map must attempt to provide a classification result. This design is inspired from the study [4] demonstrating that each convolutional channel attends to particular semantic patterns.

Given an input video frame I and its associated optical flow F obtained from FlowNet2, the proposed network in Figure 1 (the generator denoted as $\mathcal{G}$) produces a reconstructed frame $\hat{I}$ and a predicted optical flow $\hat{F}$, while the discriminator $\mathcal{D}$ estimates a probability that the optical flow associated to I is the ground truth F. The GAN objective function consists of two loss functions:

$$\begin{aligned}\mathcal{L}_{\mathcal{D}}(I, F, \hat{F}) = & \frac{1}{2}\sum_{x,y,c} -\log \mathcal{D}(I, F)_{x,y,c} \\ & + \frac{1}{2}\sum_{x,y,c} -\log[1 - \mathcal{D}(I, \hat{F})_{x,y,c}]\end{aligned} \tag{5}$$

$$\mathcal{L}_{\mathcal{G}}(I, \hat{I}, F, \hat{F}) = \lambda_{\mathcal{G}} \sum_{x,y,c} -\log \mathcal{D}(I, \hat{F})_{x,y,c} + \lambda_a \mathcal{L}_{appe}(I, \hat{I}) + \lambda_f \mathcal{L}_{flow}(F, \hat{F}) \quad (6)$$

where x, y and c respectively indicate the spatial position and the corresponding channel of a unit in the feature maps outputted from $\mathcal{D}$, and λ values are the weights associated to partial losses within our proposed model. Our GAN is optimized by alternately minimizing the two GAN losses. In our experiments (see Section 4), we assigned 0.25 for $\lambda_{\mathcal{G}}$, 1 for λ_a and 2 for λ_f. This GAN aims to emphasize the efficiency of motion prediction.

3.5. Anomaly detection

Our model aims to provide a score of normality for each frame. In related studies, such scores are usually quantities measuring the similarity between a ground truth and the reconstructed/predicted output. There are two common scores employed in CNN approaches: L_p distance and Peak Signal to Noise Ratio (PSNR). The normality of each video frame is decided by comparing its score with a threshold. It is obvious that an anomalous event occurring within a small image region may be missed due to the summation and/or average operations over all pixel positions. We hence propose another score estimation scheme considering only a small patch instead of the entire frame.

First, we define partial scores individually estimated on the two model streams sharing the same patch position as

$$\begin{cases} \mathcal{S}_I(P) = \frac{1}{|P|} \sum_{i,j \in P} (I_{i,j} - \hat{I}_{i,j})^2 \\ \mathcal{S}_F(P) = \frac{1}{|P|} \sum_{i,j \in P} (F_{i,j} - \hat{F}_{i,j})^2 \end{cases} \quad (7)$$

where P indicates an image patch and $|P|$ is its number of pixels. Our frame-level score is then computed as a weighted combination of the two partial scores as follows:

$$\mathcal{S} = \log[w_F \mathcal{S}_F(\tilde{P})] + \lambda_{\mathcal{S}} \log[w_I \mathcal{S}_I(\tilde{P})] \quad (8)$$

where w_F and w_I are the weights calculated according to the training data, $\lambda_{\mathcal{S}}$ is to control the contribution of partial scores to the summation, and $\tilde{P}$ is the patch providing the highest value of $\mathcal{S}_F$ in the considering frame, *i.e.*

$$\tilde{P} \leftarrow \underset{P \text{ slides on frame}}{\operatorname{argmax}} \mathcal{S}_F(P) \quad (9)$$

The weights w_F and w_I are estimated as the inverse of average scores obtained on the training data of n images:

$$\begin{cases} w_F = \left[\frac{1}{n} \sum_{i=1}^{n} \mathcal{S}_{F_i}(\tilde{P}_i) \right]^{-1} \\ w_I = \left[\frac{1}{n} \sum_{i=1}^{n} \mathcal{S}_{I_i}(\tilde{P}_i) \right]^{-1} \end{cases} \quad (10)$$

This helps to normalize the two scores on the same scale. The size of P was set to 16×16 in our experiments. Typically, such patches are determined by a sliding window. In realistic implementation, it can be performed using a convolutional operation with a filter of size 16×16. $\lambda_{\mathcal{S}}$ was empirically set to 0.2 since the model focuses on motion prediction efficiency.

Figure 3. Examples of normal (top) and abnormal (bottom) frames in the CUHK Avenue, UCSD Ped2, Exit Gate, and Entrance Gate (from left to right) datasets. Anomalous events are highlighted including a man picking a bag, bicycle appearance, and loitering.

Finally, we perform a normalization on frame-level scores in each evaluated video as suggested in related studies such as [11, 37, 25]. Our final frame-level score is

$$\hat{\mathcal{S}}_t = \frac{\mathcal{S}_t}{\max(\mathcal{S}_{1..m})} \quad (11)$$

where t is the frame index in a video containing m frames. The score estimated from a frame of abnormal event is expected to be higher compared with the ones of normal event.

4. Experiments

We performed experiments on various benchmark datasets of anomaly detection including CUHK Avenue [26], UCSD Ped2 [24], Subway Entrance Gate and Exit Gate [1], Traffic-Belleview and Traffic-Train [52]. Their training data contain only normal events. Some examples of normal and abnormal frames in the first 4 datasets are shown in Figure 3. The first two datasets are provided with frame-level ground truth, we thus employ area under curve (AUC) of the receiver operating characteristic (ROC) curve measured according to frame-level scores outputted from the proposed model to indicate the performance. The next two Subway datasets are evaluated on event-level that requires some additional operations described below. The last two datasets are evaluated according to the average precision (AP) since the precision-recall (PR) curve was usually used for their assessment [52, 51]. We used the FlowNet2 pretrained on FlyingThing3D [33] and ChairsSDHom [15] datasets as the ground truth optical flow estimator. The GAN was trained using Adam algorithm [21] where the initial learning rates were set to 2×10^{-4} for the generator $\mathcal{G}$ and 2×10^{-5} for the discriminator $\mathcal{D}$. The description, experimental results and a discussion corresponding to each evaluation are presented in the remaining of this section.

Method	Avenue	Ped2
Conv-AE [11]	0.702	0.900
Discriminative learning [7]	0.783	-
Hashing filters [54]	-	0.910
Unmask late fusion [16]	0.806	0.822
AMDN (double fusion) [51]	-	0.908
ConvLSTM-AE [27]	0.770	0.881
DeepAppearance [42]	0.846	-
FRCN action [14]	-	0.922
TSC [28]	0.806	0.910
Stacked RNN [28]	0.817	0.922
AbnormalGAN [37]	-	0.935
GrowingGas [46]	-	0.941
Future frame prediction [25]	0.851	0.954
Our proposed method	0.869	0.962

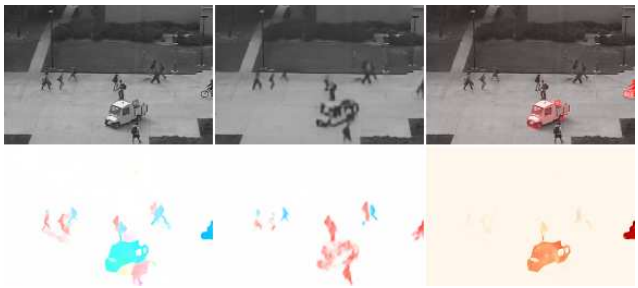
Table 1. Frame-level performance (AUC) of anomaly detection on the CUHK Avenue and UCSD Ped2 datasets. The methods are ordered according to the year of publication.

4.1. CUHK Avenue and UCSD Ped2

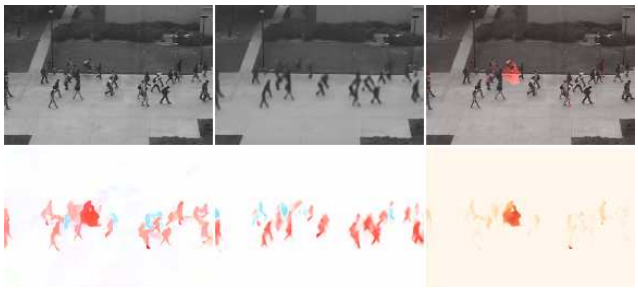
The Avenue dataset consists of 30652 frames that are split into 16 clips for training and 21 clips for testing. This dataset was captured in a campus avenue and contains various types of anomaly such as unusual action (*e.g.* running), wrong moving direction and abnormal object (*e.g.* bicycle). This also provides some challenges for evaluation such as slight camera shake and the occurrence of a few outliers.

The UCSD anomaly dataset includes two subsets Ped1 and Ped2 acquired from static cameras overlooking pedestrian walkways. The anomalies are the appearance of non-pedestrian object (*e.g.* vehicle) and strange pedestrian motion. The difference between the two subsets is the walking direction (toward and away from the camera in Ped1, parallel to the camera plane in Ped2). We select only the Ped2 dataset for two reasons. First, our optical flow estimator (FlowNet2) does not work well on very small and thin pedestrians appearing too far from the camera. Nevertheless, examples of people walking towards and away from the camera are available in the CUHK Avenue dataset allowing to evaluate performance in this situation. Second, we observed that some events were labeled as normality in the training data but were considered as anomalous in the test data (*e.g.* people walking on grass). Therefore, the Ped2 dataset (16 training and 12 testing clips) was used in our experiments.

The frame-level assessment results in Table 1 show that our model outperforms all other recent methods in the task of anomaly detection. Examples of reconstructed frames and predicted optical flows obtained from the appearance and motion streams are given in Figure 4. Considering the first example, the truck was reconstructed as a collection of pedestrian patterns since it is a new object observed by the model. The corresponding predicted motion was thus completely different from the ground truth. The processing of the bicycle on the right image edge was also similar. The second scene shows that the model still worked well on a crowded scene with many pedestrians and an anomalous object having similar intensities with the background. In the



(a) The appearance of a truck and a bicycle. (Ped2)



(b) A bicycle is running in a low contrast region. (Ped2)

(c) A man is running. (Avenue)

(d) A man is tossing papers. (Avenue)

Figure 4. (Best viewed in color) Results on the Ped2 and Avenue datasets. Each example consists of 3 image columns that are input frame and its optical flow (left), reconstructed frame and predicted motion (middle), and the frame superimposed by the motion error map below (right). The flow field color coding is the same as [15].

Method	Entrance (66)		Exit (19)	
	TP	FA	TP	FA
Subspace [9]	46	7	14	4
MPPCA [20]	57	6	19	3
DSC [55]	60	5	19	2
Sparse dict. [26]	57	4	19	2
Conv-AE [11]	61	15	17	5
IT-AE [11]	55	17	17	9
Hashing filters [54]	61	4	19	2
Early fusion [51]	56	8	15	4
Late fusion [51]	58	6	17	2
AMDN [51]	61	4	19	1
Our method	61	18	17	5

Table 2. Our results of anomaly detection on the Subway datasets. In the ground truth, the numbers of abnormal events in the Entrance and Exit are respectively 66 and 19. The term TP indicates the number of true positive detections while FA is the counting of false alarms. The methods are listed in temporal order.

next two Avenue frames, the model expected slower moving speed and another motion direction as observed in the training data. In addition, notice that the reconstructed man's trouser color was slightly different from the input frame while the back ground was well restored. This demonstrates that the model reasonably determined the low-significance relation between the color of a pattern and its movement.

4.2. Subway Entrance and Exit gates

This dataset contains videos capturing the entrance gate and exit gate of a subway station. Their lengths are respectively 96 and 43 minutes. The anomalous events in these two videos are wrong direction (*e.g*. passenger exits through the entrance gate), no payment, loitering, irregular interaction (*e.g*. a person walks awkwardly to avoid another) and miscellaneous (*e.g*. sudden changing of walking speed).

We performed the evaluation according to the ground truth of events with the training and test sets provided in [20], in which the normal events in the first 15 minutes of the Entrance Gate video and 5 minutes of the Exit Gate were used in training stage. Notice that the experiments were performed individually for the two videos.

Since the dataset does not provide the frame-level ground truth, we employ the assessment scheme in [11] to determine anomalous events in the experiments. In detail, the persistence algorithm [22] is applied on the sequence of scores to locate local maxima, in which each maximum point indicates an anomalous event. In order to reduce the effect of possible noisy detected extrema, nearby events are combined to provide only an anomalous one.

Our event-based assessment results are presented in Table 2. It shows that our model detected most anomalous events but also generated more false alarm than other recent studies. By taking a closer look at these false alarms, we determined that some events denoted as normal in the test set can be considered as anomaly under other circumstances. A visualization of some false alarms and missed anomaly detections in the Entrance dataset is given in Figure 5.

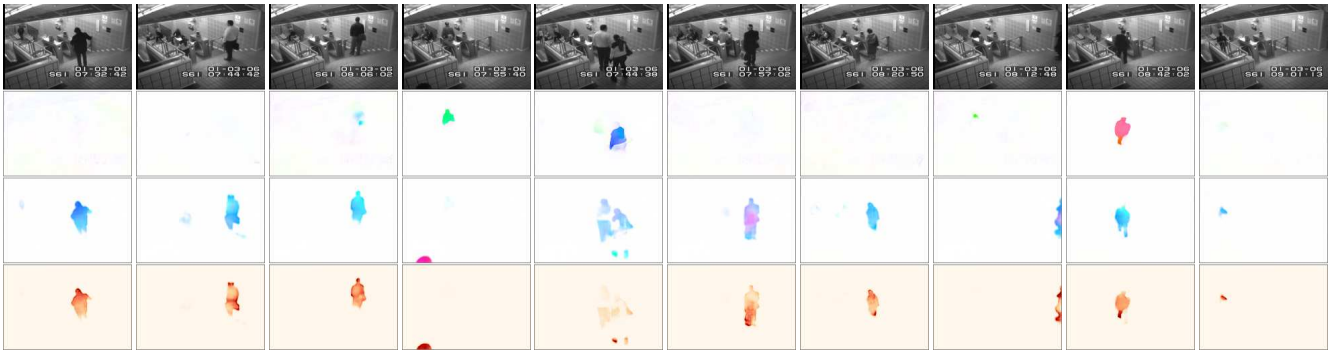
Figure 5 shows that the normality decision of movement stopping and loitering was unstable since the cases (a)-(e) were missed while (f)-(h) were wrongly detected. There are two possible reasons: (1) the use of maximum localization as in [11] is not ideal when the anomaly score smoothly and/or slowly changes, and (2) the training set (according to [20]) contains loitering event [caused by the man in (b) and (e)]. The ambiguity in ground truth annotation is also shown in the event (h) where a loitering man appeared on the right side but was not labeled as anomaly. In the event (i), the model predicted that the man would go through the left gate but he suddenly changed to the right one (the color indicates the motion direction). Since this action does not occur in the training data, the model determined it as an anomalous event. Regarding the last example (j), the motion stream expected the passenger to go to the train because most people at this location move to the left side in the training data. In other words, the model may *forget* training patterns moving to the right side. In this case, using sparse coding approaches [20, 55, 26] can be appropriate since the effect of the frequency of training patterns is reduced.

4.3. Traffic-Belleview and Traffic-Train

The Traffic-Belleview dataset was acquired by a surveillance camera looking at the traffic on a road intersection from a high viewpoint. In the training data (300 frames), vehicles only run on the main street. The appearance and movement of vehicles from/to left or right roads is defined as anomaly in the test set containing a total of 2618 frames. The video is gray-scale and has a low quality.

Unlike the previous benchmark datasets, the Traffic-Train can be considered as the most challenging dataset since the lighting conditions vary drastically together with camera jitter. The camera was mounted in a train and people movement is defined as anomaly. The training and test sets consist of 800 and 4160 frames, respectively.

Our average precision of frame-level assessment is presented in Table 3. Figure 6 shows examples of problems that the model encountered when dealing with the traffic datasets as well as illustrates the change of lighting conditions in the Train dataset. In Figure 6(b), the predicted motion was very noisy and the passenger at the frame center was missed in the error map. The effect of optical flow estimator is illustrated in Figure 6(c) where two cars were combined to be a big blob. This bad estimation significantly affected the error map though the three cars running on other way were correctly determined. The results may thus be improved by choosing another optical flow estimator or tuning the



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 5. Examples of missed detections (a)-(e) and false alarms (f)-(j) in our experiments on the Entrance dataset. Each example consists of 4 images that are (from top to bottom) the input frame, ground truth optical flow, predicted motion and the corresponding motion error map. The missed detections are: (a)-(c) movement stopping, (d) loitering, and (e) loitering (man) and movement stopping (woman). The false alarms are: (f)-(g) movement stopping, (h) loitering, (i) changing gate, and (j) passenger going near the railway. Best viewed in color.

Method	Belleview	Train
GANomaly [2]	0.735	0.194
AEs + local feature [35]	0.748	0.171
AEs + global feature [35]	0.776	0.216
ALOCC $\mathcal{D}(X)$ [40]	0.734	0.182
ALOCC $\mathcal{D}(\mathcal{R}(X))$ [40]	0.805	0.237
Our proposed method	0.751	0.490
SSIM on appearance stream	0.830	0.798

Table 3. The average precision of frame-level anomaly detection on the Traffic-Belleview and Traffic-Train datasets.

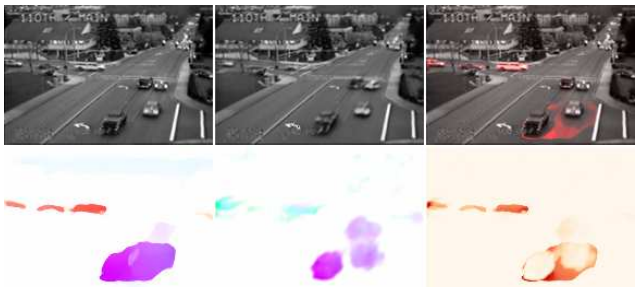
pretrained FlowNet2 by a more appropriate dataset.

As an attempt to reduce the effect of such factors, we estimated another frame-level score without the support of motion as in section 3.5. Concretely, we used the Structural Similarity Index (SSIM) [50] to compute the similarity between an input frame and its reconstruction provided by the appearance stream. Compared with other common measures such as MSE or PSNR, SSIM can work well on jitter images where pixel by pixel comparison is not appropriate. Table 3 shows that this modification improved the anomaly detection results, especially with the Train dataset.

Further details including ROC and PR curves, visualization of some feature maps and evaluation results of each single stream are provided in the supplementary materials.

(a) The change of lighting in the Traffic-Train dataset.

(b) Passengers moving in the stopping train.



(c) Cars turning to the left way.

Figure 6. (Best viewed in color) Some testing results on the two traffic datasets. Each example consists of 6 images as in Figure 4.

5. Conclusion

This paper presents an anomaly detection approach that exploits the correspondence between pattern appearances and their motions. The model is designed as a combination of two streams. The first one attempts to reconstruct the appearance according to its auto-encoder architecture while the second stream uses a U-Net structure to predict the instant motion given an input video frame. By sharing the same encoder, the model is forced to learn the correspondence. A patch-based scheme of anomaly score estimation is proposed to reduce the effect of noise in model outputs. Experiments on 6 benchmark datasets demonstrated the potential of our method. Detailed discussions are also presented to provide improvement suggestions for further works.

References

[1] Amit Adam, Ehud Rivlin, Ilan Shimshoni, and David Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(3):555–560, 2008.

[2] Samet Akcay, Amir Atapour-Abarghouei, and Toby P. Breckon. Ganomaly: Semisupervised anomaly detection via adversarial training. In *Computer Vision – ACCV 2018*, Cham, 2018. Springer International Publishing.

[3] Arslan Basharat, Alexei Gritai, and Mubarak Shah. Learning object motion patterns for anomaly detection and improved object detection. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[4] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[5] Kai-Wen Cheng, Yie-Tarng Chen, and Wen-Hsien Fang. Video anomaly detection and localization using hierarchical feature representation and gaussian process regression. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2909–2917, June 2015.

[6] Yang Cong, Junsong Yuan, and Ji Liu. Sparse reconstruction cost for abnormal event detection. In *CVPR 2011*, pages 3449–3456, June 2011.

[7] Allison Del Giorno, J. Andrew Bagnell, and Martial Hebert. A discriminative framework for anomaly detection in large videos. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 334–349, Cham, 2016. Springer International Publishing.

[8] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Häusser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2758–2766, Dec 2015.

[9] Ehsan Elhamifar and Rene Vidal. Sparse subspace clustering. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2790–2797, June 2009.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[11] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K. Roy-Chowdhury, and Larry S. Davis. Learning temporal regularity in video sequences. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 733–742, June 2016.

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, Oct 2017.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016.

[14] Ryota Hinami, Tao Mei, and Shin'ichi Satoh. Joint detection and recounting of abnormal events by learning deep generic knowledge. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3639–3647, Oct 2017.

[15] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[16] Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, and Marius Popescu. Unmasking the abnormal events in video. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2914–2922, Oct 2017.

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, July 2017.

[18] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4565–4574, June 2016.

[19] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):664–676, April 2017.

[20] Jaechul Kim and Kristen Grauman. Observe locally, infer globally: A space-time mrf for detecting abnormal activities with incremental updates. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2928, June 2009.

[21] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[22] Yeara Kozlov and Tino Weinkauf. Extracting and filtering minima and maxima of 1d functions. `https://www.csc.kth.se/~weinkauf/notes/persistence1d.html`. [Accessed 15-Feb-2019].

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[24] Weixin Li, Vijay Mahadevan, and Nuno Vasconcelos. Anomaly detection and localization in crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):18–32, Jan 2014.

[25] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection a new baseline. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[26] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *2013 IEEE International Conference on Computer Vision*, pages 2720–2727, Dec 2013.

[27] Weixin Luo, Wen Liu, and Shenghua Gao. Remembering history with convolutional lstm for anomaly detection. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 439–444, July 2017.

[28] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 341–349, Oct 2017.

[29] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

[30] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. Anomaly detection in crowded scenes. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1975–1981, June 2010.

[31] Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2813–2821, Oct 2017.

[32] Michaël Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *CoRR*, abs/1511.05440, 2015.

[33] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[34] Gérard Medioni, Isaac Cohen, François Bremond, Somboon Hongeng, and Ramakant Nevatia. Event detection and analysis from video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):873–889, Aug 2001.

[35] Medhini G. Narasimhan and Sowmya Kamath S. Dynamic video anomaly detection and localization using sparse denoising autoencoders. *Multimedia Tools and Applications*, 77(11):13173–13195, Jun 2018.

[36] Claudio Piciarelli, Christian Micheloni, and Gian Luca Foresti. Trajectory-based anomalous event detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1544–1554, Nov 2008.

[37] Mahdyar Ravanbakhsh, Moin Nabi, Enver Sangineto, Lucio Marcenaro, Carlo Regazzoni, and Nicu Sebe. Abnormal event detection in videos using generative adversarial nets. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 1577–1581, Sep. 2017.

[38] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015.

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.

[40] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. Adversarially learned one-class classifier for novelty detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[41] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[42] Sorina Smeureanu, Radu Tudor Ionescu, Marius Popescu, and Bogdan Alexe. Deep appearance features for abnormal behavior detection in video. In Sebastiano Battiato, Giovanni Gallo, Raimondo Schettini, and Filippo Stanco, editors, *Image Analysis and Processing - ICIAP 2017*, pages 779–789, Cham, 2017. Springer International Publishing.

[43] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. *CoRR*, abs/1412.6806, 2014.

[44] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

[45] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6479–6488, June 2018.

[46] Qianru Sun, Hong Liu, and Tatsuya Harada. Online growing neural gas for anomaly detection in changing surveillance scenes. *Pattern Recognition*, 64:187 – 201, 2017.

[47] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, June 2015.

[48] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, June 2016.

[49] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *2013 IEEE International Conference on Computer Vision*, pages 3551–3558, Dec 2013.

[50] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004.

[51] Dan Xu, Yan Yan, Elisa Ricci, and Nicu Sebe. Detecting anomalous events in videos by learning deep representations of appearance and motion. *Computer Vision and Image Understanding*, 156:117 – 127, 2017. Image and Video Understanding in Big Data.

[52] Andrei Zaharescu and Richard Wildes. Anomalous behaviour detection using spatiotemporal oriented energies, subset inclusion histogram comparison and event-driven processing. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, pages 563–576, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[53] Tianzhu Zhang, Hanqing Lu, and Stan Z. Li. Learning semantic scene models by object classification and trajectory clustering. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1940–1947, June 2009.

[54] Ying Zhang, Huchuan Lu, Lihe Zhang, Xiang Ruan, and Shun Sakai. Video anomaly detection based on locality sensitive hashing filters. *Pattern Recognition*, 59:302 – 311,

2016. Compositional Models and Structured Learning for Visual Recognition.

[55] Bin Zhao, Li Fei-Fei, and Eric P. Xing. Online detection of unusual events in videos via dynamic sparse coding. In *CVPR 2011*, pages 3313–3320, June 2011.