Object-centric Auto-encoders and Dummy Anomalies for Abnormal Event Detection in Video

Radu Tudor Ionescu[1,2,3], Fahad Shahbaz Khan[1], Mariana-Iuliana Georgescu[2,3], Ling Shao[1]
[1]Inception Institute of Artificial Intelligence (IIAI), Abu Dhabi, UAE
[2]University of Bucharest, 14 Academiei, Bucharest, Romania
[3]SecurifAI, 21 Mircea Vodă, Bucharest, Romania

Abstract

Abnormal event detection in video is a challenging vision problem. Most existing approaches formulate abnormal event detection as an outlier detection task, due to the scarcity of anomalous data during training. Because of the lack of prior information regarding abnormal events, these methods are not fully-equipped to differentiate between normal and abnormal events. In this work, we formalize abnormal event detection as a one-versus-rest binary classification problem. Our contribution is two-fold. First, we introduce an unsupervised feature learning framework based on object-centric convolutional auto-encoders to encode both motion and appearance information. Second, we propose a supervised classification approach based on clustering the training samples into normality clusters. A one-versus-rest abnormal event classifier is then employed to separate each normality cluster from the rest. For the purpose of training the classifier, the other clusters act as dummy anomalies. During inference, an object is labeled as abnormal if the highest classification score assigned by the one-versus-rest classifiers is negative. Comprehensive experiments are performed on four benchmarks: Avenue, ShanghaiTech, UCSD and UMN. Our approach provides superior results on all four data sets. On the large-scale ShanghaiTech data set, our method provides an absolute gain of 8.4% *in terms of frame-level AUC compared to the state-of-the-art method [34].*

1. Introduction

Abnormal event detection in video has drawn a lot of attention in the past couple of years [7, 11, 12, 13, 14, 21, 22, 24, 27, 28, 31, 33, 34, 36, 37, 38], perhaps because it is considered a challenging task due to the commonly accepted definition of abnormal events, which relies on context. An example that illustrates the importance of context is a scenario in which a truck is being driven on the street (normal event) versus a scenario in which a truck is being driven in a pedestrian area (abnormal event). In addition to the reliance on context, abnormal events rarely occur and are generally dominated by more familiar (normal) events. Therefore, it is difficult to obtain a sufficiently representative set of anomalies, making it hard to employ traditional supervised learning methods.

Most existing anomaly detection approaches [2, 5, 15, 18, 23, 25, 26, 37, 39] are based on outlier detection and learn a model of normality from training videos containing only familiar events. During inference, events are labeled as abnormal if they deviate from the normality model. Different from these approaches, we address abnormal event detection by formulating the task as a multi-class classification problem instead of an outlier detection problem. Since the training data contains only normal events, we first apply k-means clustering in order to find clusters representing various types of normality (see Figure 1). Next, we train a binary classifier following the one-versus-rest scheme in order to separate each normality cluster from the others. During training, normality clusters are treated as different categories, leading to the synthetic generation of abnormal training data. During inference, the highest classification score corresponding to a given test sample represents the normality score of the respective sample. If the score is negative, the sample is labeled as abnormal (since it does not belong to any normality class). To our knowledge, we are the first to treat the abnormal event detection task as a discriminative multi-class classification problem.

In general, existing abnormal event detection frameworks extract features at a local level [7, 9, 15, 22, 23, 24, 25, 31, 32, 38], global (frame) level [21, 26, 27, 28, 33], or both [5, 6, 11]. All these approaches extract features without explicitly taking into account the objects of interest. In this paper, we propose an object-centric approach by applying a fast yet powerful single-shot detector (SSD) [19] on each frame, and learning deep unsupervised features using convolutional auto-encoders (CAE) on top of the detected objects, as shown in Figure 1. This enables us to explicity focus only on the objects present in the scene. In addition, it allows us to accurately localize the anomalies in each frame. Although auto-encoders have been used before for abnormal event detection [11, 31, 37], to our knowledge, we are the first to train object-centric auto-encoders.

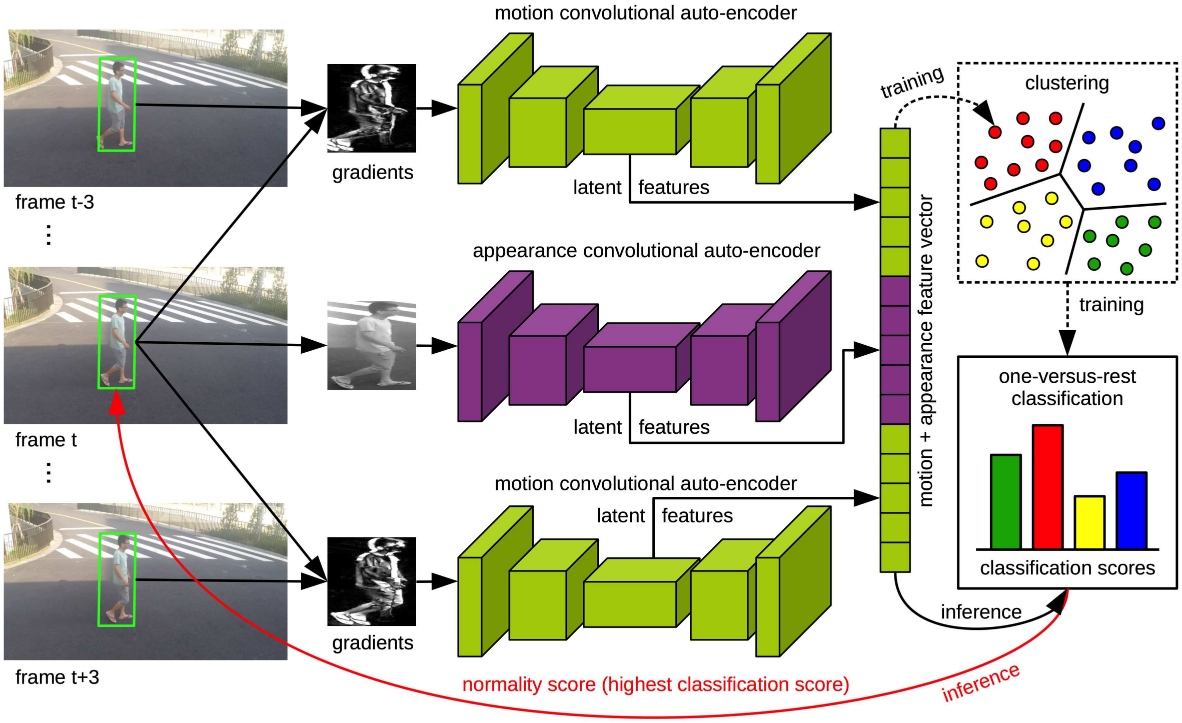
motion convolutional auto-encoder
gradients
latent
features
frame t-3
appearance convolutional auto-encoder
latent
features
frame t
motion convolutional auto-encoder
latent
features
gradients
frame t+3
training
clustering
motion + appearance feature vector
training
one-versus-rest classification
classification scores
inference
normality score (highest classification score)
inference

Figure 1. Our anomaly detection framework based on training convolutional auto-encoders on top of object detections. In the training phase (represented in dashed lines), the concatenated motion and appearance latent representations are clustered and a one-versus-rest classifier is trained to discriminate between the formed clusters. In the inference phase, we label a test sample as abnormal if the highest classification score is negative, i.e. the sample is not attributed to any class. Best viewed in color.

In summary, the novelty of our paper is two-fold. First, we train object-centric convolutional auto-encoders for both motion and appearance. Second, we propose a supervised learning approach by formulating the abnormal event detection task as a multi-class problem. We conduct experiments on the Avenue [23], the ShanghaiTech [24], the UCSD [25] and the UMN [26] data sets, and compare our approach with the state-of-the-art abnormal event detection methods [6, 7, 9, 11, 12, 13, 14, 15, 21, 22, 23, 24, 25, 26, 27, 28, 31, 32, 33, 34, 36, 37, 38]. The empirical results clearly show that our approach achieves superior performance compared to the state-of-the-art methods on all data sets. Furthermore, on the Avenue and the ShanghaiTech data sets, our approach provides considerable absolute gains of 1.5% and 8.4%, respectively, over the state-of-the-art methods [14, 34].

We organize the paper as follows. We present related work on abnormal event detection in Section 2. We describe our approach in Section 3. We present the abnormal event detection experiments in Section 4. We draw our final conclusions in Section 5.

2. Related Work

Abnormal event detection is commonly formalized as an outlier detection task [2, 5, 6, 9, 14, 15, 18, 23, 25, 26, 29, 36, 37, 38, 39], in which the main approach is to learn a model of familiarity from training videos and label the detected outliers as abnormal. Several abnormal event detection approaches [5, 6, 9, 23, 29] learn a dictionary of atoms representing normal events during training, then label the events not represented in the dictionary as abnormal. Some recent approaches have employed locality sensitive hashing [38] and deep learning [11, 12, 21, 24, 27, 28, 31, 33, 36, 37] to achieve better results. For instance, Smeureanu et al. [33] employed a one-class Support Vector Machines (SVM) model based on deep features provided by convolutional neural networks (CNN) pre-trained on the ILSVRC benchmark [30], while Ravanbakhsh et al. [27] combined pre-trained CNN models with low-level optical-flow maps.

Similar to our own approach, which learns features in an unsupervised fashion, there are a few works that have employed unsupervised steps for abnormal event detection [9, 11, 29, 31, 36, 37]. Interestingly, some recent works do not require training data at all, in order to detect abnormal events [7, 13, 22]. More closely-related to our work are methods that employ features learned with auto-encoders [11, 31, 36, 37] or extracted from the classification branch of Fast R-CNN [12]. In order to learn deep features without supervision, Xu et al. [36, 37] used Stacked Denoising Auto-Encoders on multi-scale patches. To detect abnormal events, Xu et al. [36, 37] used one-class SVM on top of the deep features. Hasan et al. [11] employed two auto-encoders, one that is learned on conventional handcrafted features, and another one that is learned in an end-to-end fashion using a fully convolutional feed-forward network. On the other hand, Sabokrou et al. [31] combined 3D deep auto-encoders and 3D convolutional neural networks into a cascaded framework.

Differences of our approach. Different from these recent related works [11, 31, 36, 37], we propose to train auto-encoders on object detections provided by a state-of-the-art detector [19]. The most similar work to ours is that of Hinami et al. [12]. They also proposed an object-centric approach, but our detection, feature extraction and training stages are different. While Hinami et al. [12] used geodesic [17] and moving object proposals [10], we employ a single-shot detector [19] based on Feature Pyramid Networks (FPN). In the feature extraction stage, Hinami et al. [12] fine-tuned the classification branch of the Fast R-CNN model on multiple visual tasks to exploit semantic information that is useful for detecting and recounting abnormal events. In contrast, we learn unsupervised deep features with convolutional auto-encoders. Also differing from Hinami et al. [12] and all other works, we formalize the abnormal event detection task as a multi-class problem and propose to train a one-versus-rest SVM on top of k-means clusters. A similar approach was adopted by Caron et al. [4] in order to train deep generic visual features in an unsupervised manner.

3. Method

Motivation. Since the training data contains only normal events, supervised learning methods that require both positive (normal) and negative (abnormal) samples cannot be directly applied for the abnormal event detection task. However, we believe that including any form of supervision is an important step towards obtaining better performance in practice. Motivated by this intuition, we conceive a framework that incorporates two approaches for including supervision. The first approach consists of employing a single-shot object detector [19], which is trained in a supervised fashion, in order to obtain object detections that are subsequently used throughout the rest of the processing pipeline. The second approach consists of training supervised one-versus-rest classifiers on artificially-generated classes representing different kinds of normality. The classes are generated by previously clustering the training samples. Our entire framework is composed of four sequential stages that are described in detail below. These are the object detection stage, the feature learning stage, the model training stage, and the inference stage.

Object detection. We propose to detect objects using a single-shot object detector based on FPN [19], which offers an optimal trade-off between accuracy and speed. This object detector is specifically chosen because (i) it can accurately detect smaller objects, due to the FPN architecture, and (ii) it can process about 13 frames per second on a GPU. These advantages are of utter importance for developing a practical abnormal event detection framework. The object detector is applied on a frame by frame basis in order to obtain a set of bounding boxes for the objects in each frame t. We use the bounding boxes to crop the objects. The resulting images are converted to grayscale. Next, the images are directly passed to the feature learning stage, in order to learn object-centric appearance features. At the same time, we use the images containing objects in order to compute gradients representing motion. For this step, we additionally consider the images cropped from a previous and a subsequent frame. As illustrated in Figure 1, we choose the frames at index $t-3$ and $t+3$, with respect to the current frame t. Since the temporal distance between the frames is not significant, we do not need to track the objects. Instead, we simply consider the bounding boxes determined at frame t in order to crop the objects at frames $t-3$ and $t+3$. For each object, we obtain two image gradients, one representing the change in motion from frame $t-3$ to frame t and one representing the change in motion from frame t to frame $t+3$. Finally, the image gradients are also passed to the feature learning stage, in order to learn object-centric motion features.

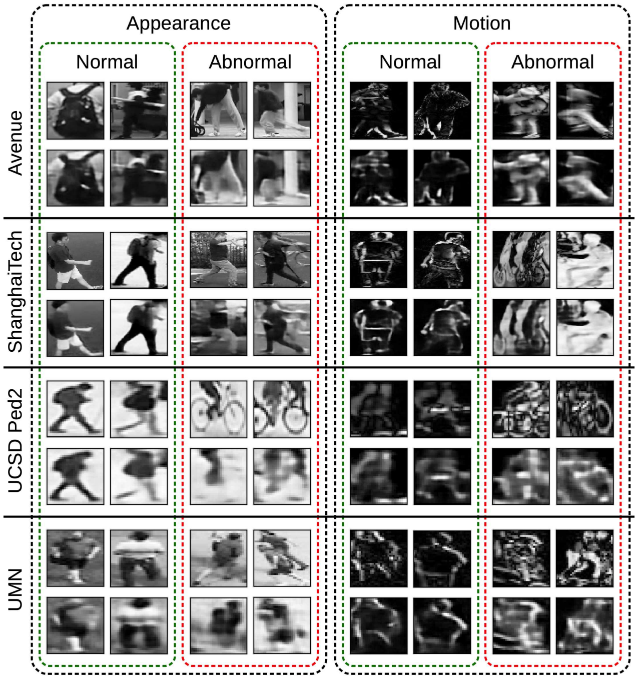

Figure 2. Normal and abnormal objects (left) and gradients (right) with reconstructions provided by the appearance (left) and the motion (right) convolutional auto-encoders. The samples are selected from the Avenue [23], the ShanghaiTech [24], the UCSD Ped2 [25] and the UMN [26] test videos, and are not seen during training the auto-encoders.

Feature learning. In order to obtain a feature vector for each object detection, we train three convolutional auto-encoders. One auto-encoder takes as input cropped images containing objects, and it inherently learns latent appearance features. The other two auto-econders take as input the gradients that capture how the object moved before and after the detection moment, respectively. These auto-encoders learn latent motion features. All three auto-encoders are based on the same lightweight architecture, which is composed of an encoder with 3 convolutional and max-pooling blocks, and a decoder with 3 upsampling and convolutional blocks and an additional convolutional layer for the final output. For each CAE, the size of the input is $64 \times 64 \times 1$, and the size of the output is the same. All convolutional layers are based on 3×3 filters. Each convolutional layer, except the very last one, is followed by ReLU activations. The first two convolutional layers of the encoder contain 32 filters each, while the third convolutional layer contains 16 filters. The max-pooling layers of the encoder are based on 2×2 filters with stride 2. The resulting latent feature representation of each CAE is composed of 16 activation maps of size 8×8. In the decoder, each resize layer upsamples the input activations by a factor of two, using the nearest neighbor approach. The first convolutional layer in the decoder contains 16 filters. The following two convolutional layers of the decoder contain 32 filters each. The fourth (and last) convolutional layer of the decoder contains a single filter of size 3×3. The main purpose of the last convolutional layer is to reduce the output depth from $64 \times 64 \times 32$ to $64 \times 64 \times 1$. The auto-encoders are trained with the Adam optimizer [16] using the pixel-wise mean squared error as loss function:

$$\mathcal{L}(I, O) = \frac{1}{h \cdot w} \sum_{i=1}^{h} \sum_{j=1}^{w} \left(I_{ij} - O_{ij}\right)^2, \tag{1}$$

where I and O are the input and the output images, each of size $h \times w$ pixels (in our case, $h = w = 64$).

The auto-encoders learn to represent objects detected in the training video containing only normal behavior. When we provide as input objects with abnormal behavior, the reconstruction error of the auto-encoders is expected to be higher. Furthermore, the latent features should represent known (normal) objects in a different and better way than unknown (abnormal) objects. Some input-output CAE pairs selected from the test videos in each data set considered in the evaluation are shown in Figure 2. We notice that the auto-encoders generally provide better reconstructions for normal objects, confirming our intuition. The final feature vector for each object detection sample is a concatenation of the latent appearance features and the latent motion features. Since the latent activation maps of each CAE are $8 \times 8 \times 16$, the final feature vectors have 3072 dimensions.

Model training. We propose a novel training approach by formalizing the abnormal event detection task as a multi-class classification problem. The proposed approach aims to compensate for the lack of truly abnormal training samples, by constructing a context in which a subset of normal training samples can play the role of *dummy abnormal samples* with respect to another subset of normal training samples. This is achieved by clustering the normal training samples into k clusters using k-means. We consider that each cluster represents a certain kind of normality, different from the other clusters. From the perspective of a given cluster i, the samples belonging to the other clusters (from the set $\{1, 2, ..., k\} \setminus i$) can be viewed as (dummy) abnormal samples. Therefore, we can train a binary classifier g_i, in our case an SVM, to separate the positively-labeled data points in a cluster i from the negatively-labeled data points in clusters $\{1, 2, ..., k\} \setminus i$, as follows:

$$g_i(x) = \sum_{j=1}^{m} w_j \cdot x_j + b, \tag{2}$$

where $x \in \mathbb{R}^m$ is a test sample that must be classified either as normal or abnormal, w is the vector of weights and b is the bias term. We note that the negative samples can actually be considered as more closely-related to the samples in cluster i than truly abnormal samples. Hence, the discrimination task is more difficult, and it can help the SVM to select better support vectors. For each cluster i, we train an independent binary classifier g_i. The final classification score for one data sample is the highest score among the scores returned by the k classifiers. In other words, the classification score for one data sample is selected according to the one-versus-rest scheme, commonly used when binary classifiers are employed for solving multi-class problems.

Inference. In the inference phase, each test sample x is classified by the k binary SVM models. The highest classification score is used (with a change of sign) as the abnormality score s for the respective test sample x:

$$s(x) = -\max_{i}\{g_i(x)\}, \forall i \in \{1, 2, ..., k\}. \tag{3}$$

By putting together the scores of the objects cropped from a given frame, we obtain a pixel-level anomaly prediction map for the respective frame. If the bounding boxes of two objects overlap, we keep the maximum abnormality score for the overlapping region. To obtain frame-level predictions, we take the highest score in the prediction map as the anomaly score of the respective frame. Finally, we apply a Gaussian filter to temporally smooth the frame-level anomaly scores.

4. Experiments

4.1. Data Sets

Avenue. The Avenue data set [23] consists of 16 training videos with a total 15328 frames and 21 test videos with a total of 15324. The resolution of each video frame is 360×640 pixels. For each test frame, ground-truth locations of anomalies are provided using pixel-level masks.

ShanghaiTech. The ShanghaiTech Campus data set [24] is among the largest data sets for abnormal event detection. Unlike other data sets, it contains 13 different scenes with various lighting conditions and camera angles. There are 330 training videos and 107 test videos. The test set contains a total of 130 abnormal events annotated at the pixel-level. There are 316154 frames in the whole data set. The resolution of each video frame is 480 × 856 pixels.

UCSD. The UCSD Pedestrian data set [25] is composed of two subsets, namely Ped1 and Ped2. As Hinami et al. [12], we exclude Ped1 from the evaluation, because it has a significantly lower frame resolution of 158 × 238. Another problem with Ped1 is that some recent works report results only on a subset of 16 videos [27, 28, 36], while others [13, 25, 21, 22] report results on all 36 test videos. We thus consider only UCSD Ped2, which contains 16 training and 12 test videos. The resolution of each frame is 240 × 360 pixels. There are 2550 frames for training and 2010 for testing. The videos illustrate various crowded scenes, and anomalies include bicycles, vehicles, skateboarders and wheelchairs crossing pedestrian areas.

UMN. The UMN Unusual Crowd Activity data set [26] consists of three independent crowded scenes of different lengths. The three scenes consist of 1453 frames, 4144 frames and 2144 frames, respectively. The resolution of each video frame is 240 × 320 pixels. The normal behavior is represented by people walking around, while the abnormal behavior is represented by people running in different directions.

4.2. Evaluation

As evaluation metric, we employ the *area under the curve* (AUC) computed with regard to ground-truth annotations at the frame-level. The frame-level AUC metric used in most previous works [6, 7, 13, 14, 21, 22, 23, 24, 25, 34, 36] considers a frame as being a correct detection, if it contains at least one abnormal pixel. We adopt the same frame-level AUC definition as these previous works. In order to obtain the final abnormality maps, our pixel-level detection maps are smoothed using a similar technique as [7, 13, 23].

4.3. Parameter and Implementation Details

In the object detection stage, we employ a single-shot detector based on FPN [19] that is pre-trained on the COCO data set [20]. The detector is downloaded from the TensorFlow detection model zoo. For the training set, we keep the detections with a confidence level higher than 0.5, and for the test set, we keep those with a confidence level higher than 0.4. The convolutional auto-encoders used in the feature learning stage are implemented in TensorFlow [1]. We train the auto-encoders for 100 epochs with the learning rate set to 10^{-3}, and for another 100 epochs with the learning rate set to 10^{-4}. We use mini-batches of 64 samples. We train independent auto-encoders for each of the four data sets considered in the evaluation. To cluster the training samples with k-means, we employ the VLFeat [35] implementation, which is based on the Lloyd algorithm [8]. We adopt k-means++ [3] initialization. We repeat the clustering 10 times, selecting the partitioning with the minimum energy. In all the experiments, we set the number of k-means clusters to $k = 10$. We set the regularization parameter of the linear SVM (implemented in VLFeat [35]) to $C = 1$.

Method	Avenue	Shanghai Tech	UCSD Ped2	UMN
Kim et al. [15]	-	-	69.3	-
Mehran et al. [26]	-	-	55.6	96.0
Mahadevan et al. [25]	-	-	82.9	-
Cong et al. [6]	-	-	-	97.8
Saligrama et al. [32]	-	-	-	98.5
Lu et al. [23]	80.9	-	-	-
Dutta et al. [9]	-	-	-	99.5
Xu et al. [36, 37]	-	-	90.8	-
Hasan et al. [11]	70.2	60.9	90.0	-
Del Giorno et al. [7]	78.3	-	-	91.0
Zhang et al. [38]	-	-	91.0	98.7
Smeureanu et al. [33]	84.6	-	-	97.1
Ionescu et al. [13]	80.6	-	82.2	95.1
Luo et al. [24]	81.7	68.0	92.2	-
Hinami et al. [12]	-	-	92.2	-
Ravanbakhsh et al. [28]	-	-	93.5	99.0
Sabokrou et al. [31]	-	-	-	**99.6**
Ravanbakhsh et al. [27]	-	-	88.4	98.8
Liu et al. [21]	85.1	72.8	95.4	-
Liu et al. [22]	84.4	-	87.5	96.1
Sultani et al. [34]	-	76.5	-	-
Ionescu et al. [14]	88.9	-	-	99.3
Ours	**90.4**	**84.9**	**97.8**	**99.6**

Table 1. Abnormal event detection results (in %) in terms of frame-level AUC on the Avenue [23], the ShanghaiTech [24], the UCSD Ped2 [25] and the UMN [26] data sets. Our framework is compared with several state-of-the-art approaches [6, 7, 9, 11, 12, 13, 14, 15, 21, 22, 23, 24, 25, 26, 27, 28, 31, 32, 33, 34, 36, 37, 38], which are listed in temporal order. The results of Sultani et al. [34] on ShanghaiTech are based on their pre-trained model.

4.4. Results

We evaluate our approach in comparison with a series of state-of-the-art methods [6, 7, 9, 11, 12, 13, 15, 21, 22, 23, 24, 25, 26, 27, 28, 31, 32, 33, 36, 37, 38] on the Avenue, the ShanghaiTech, the UCSD Ped2 and the UMN data sets. The corresponding results are presented in Table 1.

Avenue. On the Avenue data set, we are able to surpass the results reported in all previous works. Compared to most of the recent works [13, 21, 22, 24, 33], our method provides an absolute gain of more than 5% in terms of frame-level AUC. With a frame-level AUC of 88.9%, Ionescu et al. [14] is the best and most recent baseline. We surpass their score by 1.5%. Remarkably, with a frame-level AUC of 90.4%, our approach is the only method that surpasses the threshold

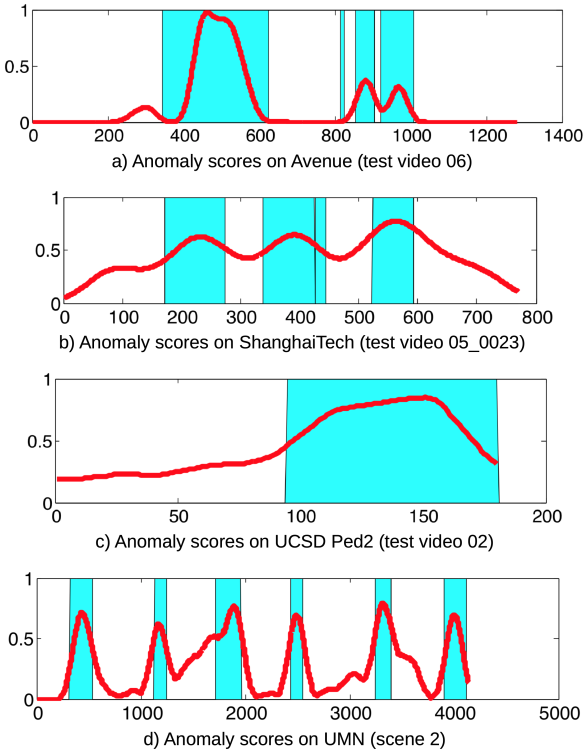

Figure 3. Frame-level anomaly detection scores between 0 and 1 (on the horizontal axis) provided by our approach, for various test videos selected from the Avenue [23], the ShanghaiTech [24], the UCSD Ped2 [25] and the UMN [26] data sets. Ground-truth abnormal events are represented in cyan and our scores are illustrated in red. Best viewed in color.

of 90% on the Avenue data set.

Notably, Hinami et al. [12] did not compare with other approaches on the official Avenue test set, arguing that there are five test videos (01, 02, 08, 09 and 10) containing static abnormal objects that are not properly labeled. Therefore, they only evaluated their method on Avenue17, a subset that excludes the respective five videos. We also compare our performance with that reported by Hinami et al. [12], being sure to exclude the same five test videos for a fair comparison. Our frame-level AUC score on the Avenue17 subset is 91.6%, which is almost 2% better than the frame-level AUC of 89.8% reported in [12]. With respect to the complete Avenue test set, we note that our framework attains a better frame-level AUC score on the Avenue17 subset, suggesting that the removed test videos are indeed more problematic than the videos left in Avenue17. As observed by Hinami et al. [12], the removed videos include some abnormal objects that are not labeled accordingly. Methods detecting these objects as abnormal are destined to reach higher false positive rates, which is unfair.

In Figure 3 (a), we present the frame-level anomaly scores provided by our method on test video 06 from Avenue. According to the ground-truth labels, which are also illustrated in Figure 3 (a), we note that there are four abnormal events in the respective test video. Our approach seems to be able to identify three of the four events, without including any false positive detections. Figure 4 (top row) illustrates a few examples of true positive and false positive abnormal event detections. From left to right, the true positive detections are *a person running*, *a person walking in the wrong direction*, *a person picking up an object* and *a person throwing an object*. The first false positive example consists of *two people that are detected in the same bounding box* by the object detector. The other false positive detection is *a person walking in the wrong direction* that is labeled as abnormal too soon.

ShanghaiTech. Since ShanghaiTech is the newest data set for abnormal event detection, there are only a few recent approaches reporting results on this data set [21, 24]. Besides these, Luo et al. [24] additionally evaluated a previously published method [11] when they introduced the data set. On the ShanghaiTech data set, the state-of-the-art performance of 72.8% is reported by Liu et al. [21]. We outperform their approach by a large margin of 12.1%. In order to compare with Sultani et al. [34] in standard formulation of the abnormal event detection task, we used the open source code provided by Sultani et al. [34] to compute their anomaly scores for the large-scale ShanghaiTech data set. As shown in Table 1, the approach of Sultani et al. [34] obtains a frame-level AUC of 76.5%, outperforming the best existing method [21]. Our approach significantly outperforms both Sultani et al. [34] and Liu et al. [21], achieving a frame-level AUC of 84.9%. With a frame-level AUC of 84.9%, our approach is the only one to surpass the 80% threshold on ShanghaiTech.

In Figure 3 (b), we display our frame-level anomaly scores against the ground-truth labels on a ShanghaiTech test video with three abnormal events. On this video, we can clearly observe a strong correlation between our anomaly scores and the ground-truth labels. Some localization results from different scenes in the ShanghaiTech data set are illustrated in the second row of Figure 4. The true positive abnormal events detected by our framework are (from left to right) *two bikers in a pedestrian area*, *a person robbing another person*, *a person jumping* and *two people fighting*. The false positive abnormal events are triggered because, in each case, there are *two people in the same bounding box* and our system labels the unusual appearance and motion generated by the two objects as abnormal.

UCSD Ped2. While older approaches [15, 26] report frame-level AUC scores under 70%, most approaches proposed in the last three years [11, 12, 21, 22, 24, 27, 28, 37, 38] reach frame-level AUC scores between 87% and 94% on UCSD Ped2. For instance, the frameworks based on auto-encoders [11, 36, 37] attain results of around 90%. Liu et al. [21] recently outperformed the previous works, reporting a frame-level AUC of 95.4%. We further surpass their

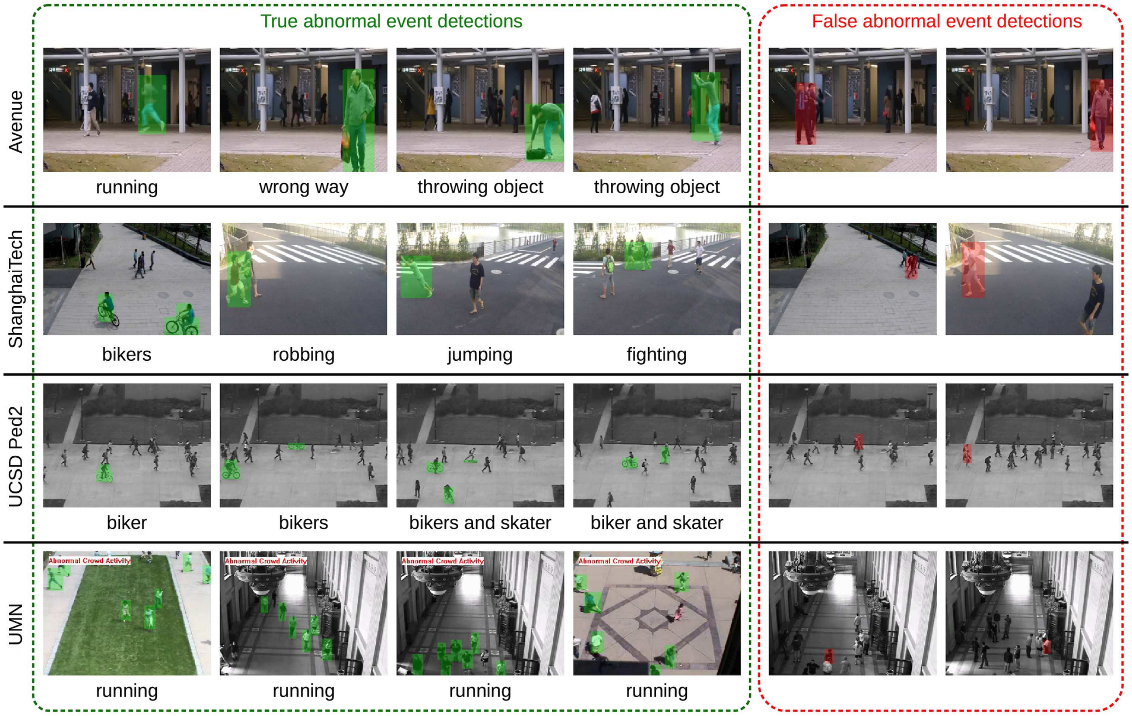

Figure 4. True positive (left) versus false positive (right) detections of our framework. Examples are selected from the Avenue [23] (first row), the ShanghaiTech [24] (second row), the UCSD Ped2 [25] (third row) and the UMN [26] (fourth row) data sets. Best viewed in color.

state-of-the-art result, reaching the top frame-level AUC of 97.8% on UCSD Ped2. Our score is 2.4% above the score reported by Liu et al. [21], 4.3% above the second-best score reported by Ravanbakhsh et al. [28], and more than 7% higher than the scores reported by other frameworks based on auto-encoders [11, 36, 37].

As for the other data sets, we compare our frame-level anomaly scores against the ground-truth labels on a test video from UCSD Ped2 in Figure 3 (c). On this particular video, our frame-level AUC is above 99%, indicating that our approach can precisely detect the abnormal event. Furthermore, the qualitative results presented in the third row of Figure 4, show that our approach can also localize the abnormal events from UCSD Ped2. From left to right, the true positive detections are *a biker in a pedestrian area*, *two bikers in a pedestrian area*, *two bikers and a skater in a pedestrian area* and *a biker and a skater in a pedestrian area*. As for ShanghaiTech, the false positive abnormal detections are caused by *two people in the same bounding box*.

UMN. It appears that UMN is the easiest abnormal event detection data set, because almost all works report frame-level AUC scores higher than 95%, with some works [9, 28, 31] even surpassing 99%. The top score of 99.6% is reported by Sabokrou et al. [31], and we reach the same performance on the UMN data set. We note that the second scene seems to be slightly more difficult than the other two scenes, since our frame-level AUC score on this scene is 99.1%, while the frame-level AUC scores on the other scenes are 99.9% and 99.8%, respectively. For this reason, we choose to illustrate the frame-level anomaly scores against the ground-truth labels for the second scene from UMN in Figure 3 (d). Overall, our anomaly scores correlate well with the ground-truth labels, but there are some normal frames with high abnormality scores just before the third abnormal event in the scene.

In the fourth row of Figure 4, we present some localization results provided by our framework. The true positive examples represent *people running around in different directions*, while the false positive detections are triggered by *two people in the same bounding box* and *a person bending down to pick up an object*. We note that the false positive examples are selected from the second scene, as we did not find false positive detections in the other two scenes.

4.5. Discussion

While the results presented in Table 1 show that our approach can outperform the state-of-the-art methods on four evaluation sets, we also aim to address questions about the robustness of our features and parameter choices, and to discuss the running time of our framework.

Parameter selection. We present results with various parameter choices on the largest and most difficult evaluation set, namely ShanghaiTech. We first variate the number of clusters k by selecting values in the set $\{5, 10, 15, 20, 25, 30\}$. The corresponding frame-level AUC scores are presented in Figure 5. The results presented in Figure 5 indicate that the number of clusters does

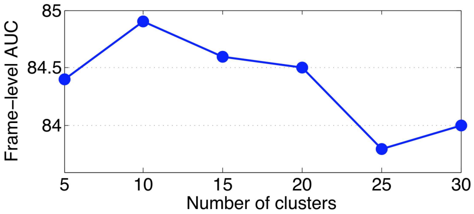

Figure 5. Frame-level AUC scores on ShanghaiTech obtained by selecting values for the number of clusters k from the set $\{5, 10, 15, 20, 25, 30\}$.

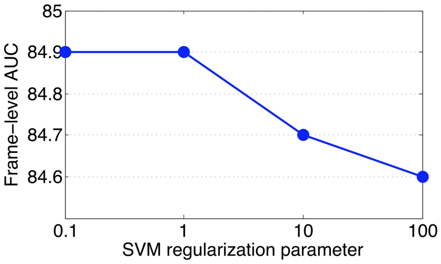

Figure 6. Frame-level AUC scores on ShanghaiTech obtained by selecting values for the SVM regularization parameter C from the set $\{0.1, 1, 10, 100\}$.

not play a significant role in our multi-class classification framework, since the accuracy variations are lower than 1.1%. With only one exception (for $k = 25$), our results are always higher than 84%. We also variate the regularization parameter of the SVM, by considering values in the set $\{0.1, 1, 10, 100\}$. The corresponding frame-level AUC scores are presented in Figure 6. The results presented in Figure 6 show that the performance variation is lower than 0.3%, and the frame-level AUC scores are always higher than 84.6%. We believe that this happens because the classes are linearly separable, since they are generated by clustering the samples with k-means into disjoint clusters. Overall, we conclude that our high improvement (12.1%) over the state-of-the-art approach [21], cannot be explained by a convenient choice of parameters.

Ablation results. In Table 2, we present feature ablation results, as well as results for a one-class SVM based on our full object-centric feature set, on the ShanghaiTech data set. When we remove the object detector and train auto-encoders at the frame-level, we obtain a frame-level AUC of 72.4%, which demonstrates the importance of extracting object-centric features and using the one-versus-rest SVM. We note that the frame-level auto-encoders have an additional convolutional layer and the input resolution is increased to 192×192. When we replace the one-class SVM with our multi-class approach based on k-means and one-versus-rest SVM, while keeping the features computed on full frames, the frame-level AUC grows to 78.7%. This demonstrates that our approach based on k-means and one-versus-rest SVM is indeed helpful. When we replace the object-centric CAE features with pre-trained SSD features (extracted right before the SSD class predictor), the frame-level AUC is only 81.3%, which shows the importance of learning features with auto-encoders. By removing either the appearance or the motion object-centric CAE features from our model, the results drop by less than 3%. This shows that both appearance and motion features are relevant for the abnormal event detection task. By replacing our multi-class approach based on k-means and one-versus-rest SVM with a one-class SVM, while keeping the combined object-centric CAE features, the performance drops by 5.7%. This result indicates that formalizing the abnormal event detection task as a multi-class problem is indeed useful. We conclude that both of our contributions are crucial to obtain superior results.

Method	Score
Frame-level CAE features + one-class SVM (baseline)	72.4
Frame-level CAE features + one-versus-rest SVM	78.7
Pre-trained SSD features + one-versus-rest SVM	81.3
CAE appearance features + one-versus-rest SVM	82.2
CAE motion features + one-versus-rest SVM	83.0
Combined CAE features + one-class SVM	79.2
Combined CAE features + one-versus-rest SVM	**84.9**

Table 2. Frame-level AUC scores (in %) on ShanghaiTech [24] obtained by removing various components from our framework versus a baseline based on frame-level features and one-class SVM.

Running time. The single-shot object detector [19] requires about 74 milliseconds to process a single frame. Hence, it can run at about 13.5 frames per second (FPS). With a reasonable average of 5 objects per frame, our feature extraction and inference stages require about 16 milliseconds per frame. Thus, we can process about 62.5 frames per second. However, the entire pipeline requires about 90 milliseconds to infer the anomaly scores for a single frame, which translates to 11 FPS. We note that more than 80% of the processing time is spent detecting objects on a frame by frame basis. The running time can be improved by replacing the current object detector with a faster one. We note that all running times were measured on an Nvidia Titan Xp GPU with 12 GB of RAM.

5. Conclusion and Future Work

We introduced a novel method for abnormal event detection in video, which is based on (i) training object-centric convolutional auto-encoders and on (ii) formalizing abnormal event detection as a multi-class problem. The empirical results obtained on four data sets indicate that our approach outperforms a series of state-of-the-art approaches [6, 7, 9, 11, 12, 13, 14, 15, 21, 22, 23, 24, 25, 26, 27, 28, 31, 32, 33, 34, 36, 37, 38]. In future work, we aim to improve our framework by segmenting and tracking objects.

Acknowledgments. The work of Radu Tudor Ionescu was partially supported through grants PN-III-P1-1.1-PD-2016-0787 and PN-III-P2-2.1-PED-2016-1842.

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: A system for large-scale machine learning. In *Proceedings of OSDI*, pages 265–283, 2016.

[2] B. Antic and B. Ommer. Video parsing for abnormality detection. In *Proceedings of ICCV*, pages 2415–2422, 2011.

[3] D. Arthur and S. Vassilvitskii. k-means++: The Advantages of Careful Seeding. In *Proceedings of SODA*, pages 1027–1035, 2007.

[4] M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep Clustering for Unsupervised Learning of Visual Features. In *Proceedings of ECCV*, volume 11218, pages 139–156, 2018.

[5] K.-W. Cheng, Y.-T. Chen, and W.-H. Fang. Video anomaly detection and localization using hierarchical feature representation and Gaussian process regression. In *Proceedings of CVPR*, pages 2909–2917, 2015.

[6] Y. Cong, J. Yuan, and J. Liu. Sparse reconstruction cost for abnormal event detection. In *Proceedings of CVPR*, pages 3449–3456, 2011.

[7] A. Del Giorno, J. Bagnell, and M. Hebert. A Discriminative Framework for Anomaly Detection in Large Videos. In *Proceedings of ECCV*, pages 334–349, 2016.

[8] Q. Du, V. Faber, and M. Gunzburger. Centroidal Voronoi Tessellations: Applications and Algorithms. *SIAM Review*, 41(4):637–676, 1999.

[9] J. K. Dutta and B. Banerjee. Online Detection of Abnormal Events Using Incremental Coding Length. In *Proceedings of AAAI*, pages 3755–3761, 2015.

[10] K. Fragkiadaki, P. Arbelaez, P. Felsen, and J. Malik. Learning to Segment Moving Objects in Videos. In *Proceedings of CVPR*, pages 4083–4090, 2015.

[11] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis. Learning temporal regularity in video sequences. In *Proceedings of CVPR*, pages 733–742, 2016.

[12] R. Hinami, T. Mei, and S. Satoh. Joint Detection and Recounting of Abnormal Events by Learning Deep Generic Knowledge. In *Proceedings of ICCV*, pages 3639–3647, 2017.

[13] R. T. Ionescu, S. Smeureanu, B. Alexe, and M. Popescu. Unmasking the abnormal events in video. In *Proceedings of ICCV*, pages 2895–2903, 2017.

[14] R. T. Ionescu, S. Smeureanu, M. Popescu, and B. Alexe. Detecting abnormal events in video using narrowed normality clusters. In *Proceedings of WACV*, pages 1951–1960, 2019.

[15] J. Kim and K. Grauman. Observe locally, infer globally: A space-time MRF for detecting abnormal activities with incremental updates. In *Proceedings of CVPR*, pages 2921–2928, 2009.

[16] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of ICLR*, 2015.

[17] P. Krähenbühl and V. Koltun. Geodesic Object Proposals. In *Proceedings of ECCV*, volume 8693, pages 725–739, 2014.

[18] W. Li, V. Mahadevan, and N. Vasconcelos. Anomaly detection and localization in crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):18–32, 2014.

[19] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *Proceedings of CVPR*, pages 2117–2125, 2017.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *Proceedings of ECCV*, pages 740–755, 2014.

[21] W. Liu, W. Luo, D. Lian, and S. Gao. Future Frame Prediction for Anomaly Detection – A New Baseline. In *Proceedings of CVPR*, pages 6536–6545, 2018.

[22] Y. Liu, C.-L. Li, and B. Póczos. Classifier Two-Sample Test for Video Anomaly Detections. In *Proceedings of BMVC*, 2018.

[23] C. Lu, J. Shi, and J. Jia. Abnormal Event Detection at 150 FPS in MATLAB. In *Proceedings of ICCV*, pages 2720–2727, 2013.

[24] W. Luo, W. Liu, and S. Gao. A Revisit of Sparse Coding Based Anomaly Detection in Stacked RNN Framework. In *Proceedings of ICCV*, pages 341–349, 2017.

[25] V. Mahadevan, W.-X. LI, V. Bhalodia, and N. Vasconcelos. Anomaly Detection in Crowded Scenes. In *Proceedings of CVPR*, pages 1975–1981, 2010.

[26] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *Proceedings of CVPR*, pages 935–942, 2009.

[27] M. Ravanbakhsh, M. Nabi, H. Mousavi, E. Sangineto, and N. Sebe. Plug-and-Play CNN for Crowd Motion Analysis: An Application in Abnormal Event Detection. In *Proceedings of WACV*, pages 1689–1698, 2018.

[28] M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. Regazzoni, and N. Sebe. Abnormal Event Detection in Videos using Generative Adversarial Nets. In *Proceedings of ICIP*, pages 1577–1581, 2017.

[29] H. Ren, W. Liu, S. I. Olsen, S. Escalera, and T. B. Moeslund. Unsupervised Behavior-Specific Dictionary Learning for Abnormal Event Detection. In *Proceedings of BMVC*, pages 28.1–28.13, 2015.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, K. A., A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[31] M. Sabokrou, M. Fayyaz, M. Fathy, and R. Klette. Deep-cascade: Cascading 3D Deep Neural Networks for Fast Anomaly Detection and Localization in Crowded Scenes. *IEEE Transactions on Image Processing*, 26(4):1992–2004, 2017.

[32] V. Saligrama and Z. Chen. Video anomaly detection based on local statistical aggregates. In *Proceedings of CVPR*, pages 2112–2119, 2012.

[33] S. Smeureanu, R. T. Ionescu, M. Popescu, and B. Alexe. Deep Appearance Features for Abnormal Behavior Detection in Video. In *Proceedings of ICIAP*, volume 10485, pages 779–789, 2017.

[34] W. Sultani, C. Chen, and M. Shah. Real-World Anomaly Detection in Surveillance Videos. In *Proceedings of CVPR*, pages 6479–6488, 2018.

[35] A. Vedaldi and B. Fulkerson. VLFeat: An Open and Portable Library of Computer Vision Algorithms. http://www.vlfeat.org/, 2008.

[36] D. Xu, E. Ricci, Y. Yan, J. Song, and N. Sebe. Learning Deep Representations of Appearance and Motion for Anomalous Event Detection. In *Proceedings of BMVC*, pages 8.1–8.12, 2015.

[37] D. Xu, Y. Yan, E. Ricci, and N. Sebe. Detecting Anomalous Events in Videos by Learning Deep Representations of Appearance and Motion. *Computer Vision and Image Understanding*, 156:117–127, 2017.

[38] Y. Zhang, H. Lu, L. Zhang, X. Ruan, and S. Sakai. Video anomaly detection based on locality sensitive hashing filters. *Pattern Recognition*, 59:302–311, 2016.

[39] B. Zhao, L. Fei-Fei, and E. P. Xing. Online Detection of Unusual Events in Videos via Dynamic Sparse Coding. In *Proceedings of CVPR*, pages 3313–3320, 2011.