ComplexVAD: Detecting Interaction Anomalies in Video

Mumcu, Furkan; Jones, Michael J.; Yilmaz, Yasin; Cherian, Anoop

TR2025-016 February 19, 2025

Abstract

Existing video anomaly detection datasets are inadequate for representing complex anomalies that occur due to the interactions between objects. The absence of complex anomalies in previous video anomaly detection datasets affects research by shifting the focus onto simple anomalies. To ad- dress this problem, we introduce a new large-scale dataset: ComplexVAD. In addition, we propose a novel method to detect complex anomalies via modeling the interactions be- tween objects using a scene graph with spatio-temporal attributes. With our proposed method and two other state- of-the-art video anomaly detection methods, we obtain base- line scores on ComplexVAD and demonstrate that our new method outperforms existing works.

IEEE Winter Conference on Applications of Computer Vision (WACV) Workshop 2025

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

ComplexVAD: Detecting Interaction Anomalies in Video

Furkan Mumcu[*]
University of South Florida
furkan@usf.edu

Michael J. Jones
Mitsubishi Electric Research Labs (MERL)
mjones@merl.com

Yasin Yilmaz
Univ. of South Florida
yasiny@usf.edu

Anoop Cherian
Mitsubishi Electric Research Labs (MERL)
cherian@merl.com

Abstract

Existing video anomaly detection datasets are inadequate for representing complex anomalies that occur due to the interactions between objects. The absence of complex anomalies in previous video anomaly detection datasets affects research by shifting the focus onto simple anomalies. To address this problem, we introduce a new large-scale dataset: ComplexVAD. In addition, we propose a novel method to detect complex anomalies via modeling the interactions between objects using a scene graph with spatio-temporal attributes. With our proposed method and two other state-of-the-art video anomaly detection methods, we obtain baseline scores on ComplexVAD and demonstrate that our new method outperforms existing works.

1. Introduction

Video anomaly detection (VAD) has become a popular research area with important security and public safety applications due to the massive amount of video surveillance data being generated which humans cannot effectively monitor. Video anomaly detection algorithms are crucial for flagging unusual activity in surveillance video for further review by human operators. Various formulations of the video anomaly detection problem have been studied by the research community. In this paper, we focus on the formulation in which nominal videos (also called training videos) containing only normal activities in a particular scene are provided for learning a model. The goal is to temporally and spatially localize anomalous activity occurring in test video of the same scene.

We are focused on single-scene video anomaly detection because it corresponds to a very common use-case: using a camera to monitor activity at a particular location and alert someone when unusual activity occurs. In such scenarios, what is normal in one scene may not be normal in another. For a new scene, the idiosyncrasies of that scene would need to be learned from video of the new scene and not generalized from other scenes. For example, something as innocuous as walking across the grass may be anomalous in one scene, but perfectly normal in another. One can only know this by viewing normal video of a particular scene. Another important difference between single-scene and multi-scene VAD is the presence of location-specific anomalies in single-scene VAD (i.e. activity that is anomalous in some locations but not others in a scene). Because there is no overlap in locations for the difference scenes in multi-scene VAD, such datasets do not include location-specific anomalies. Thus, multi-scene VAD is not a generalization of single-scene VAD. This is an important point that is often overlooked by researchers in this field.

There have been many datasets introduced for the single-scene version of video anomaly detection, including UCSD Ped1 and Ped2 [22], CUHK Avenue [19], Street Scene [26], NOLA [8], and IITB-Corridor [27]. All of these datasets contain anomalous activity that mainly involves a single object or actor, such as a golf cart driving on a pedestrian walkway, a jaywalker, or a person running, etc. In the real world, anomalous activity is often not this simple. In this paper, we introduce the idea of complex anomalies which are anomalies that involve the interaction of two or more objects/actors. Some examples of complex anomalies include a cyclist running into a car, a person falling off of a skateboard, and a person sitting on a car. Because existing datasets have very few complex anomalies, we introduce a new dataset, called ComplexVAD, with many different types of anomalies involving interactions between two objects. By introducing this new dataset, we hope to encourage more complex models of scenes that include modeling of object interactions. We expect such models to expand the types of anomalies that can be reliably detected.

In addition to introducing a new dataset and a new direc-

[*] Furkan Mumcu did part of this work as an intern at MERL.

tion for video anomaly detection research, we also propose a novel method for detecting complex anomalies in video. In our method, we generate scene graphs by turning frames into graph representations. Each object of each frame is extracted (using a multi-class object detector) and treated as a node in the graph where node features are represented by the current location, bounding box, motion trajectory for the next T frames, object class identifier, and skeletal pose if the object is a person. Each node is then connected with nearby nodes in the frame if the 3D spatial distance between objects is below a threshold.

At the end of this process, we have a graph representation for each frame. We group node-to-node connections which we simply call node pairs or just pairs, into a set of normal pairs. We also collect isolated nodes into another set to detect simple anomalies. Then, we reduce both sets to smaller sets which we call exemplars by removing redundant instances. The details of exemplar selection are given in Section 4. For a given test video, we again compute scene graphs for test frames and compare node pairs and isolated nodes to the appropriate exemplar set using distance functions between object attributes which are explained in Section 4. Any test instance with a high distance to every nominal exemplar is considered anomalous.

On the ComplexVAD dataset, we compare our proposed method against two state-of-the-art video anomaly detection methods using the frame-level criterion [16], the region-based detection criterion [26] and the track-based detection criterion [26]. Our experimental results show that while our method performs better than existing methods, complex anomaly detection is a difficult problem in needs of further investigation.

In summary, we make the following key contributions:

- We introduce a new large-scale video anomaly detection dataset, named ComplexVAD, to encourage further research on detecting more difficult complex anomalies.
- We propose a novel video anomaly detection method based on scene graphs to detect complex anomalies.
- We demonstrate improved results for our proposed method over two state-of-the-art video anomaly detection methods which establishes a baseline for the new ComplexVAD dataset.

2. Related Work

There have been many datasets introduced for the problem of video anomaly detection. UCSD Ped1 and Ped2 [22] are early datasets with simple anomalies such as cyclists, golf carts and people walking in unusual places. CUHK Avenue [19] is another popular dataset whose anomalies include people running or walking in unusual directions or throwing things into the air. Street Scene [26] emphasizes location-dependent anomalies such as people jaywalking, cars parked illegally or cars/bikes moving outside of their designated lanes. IITB-Corridor [27] is a dataset with anomalies such as loitering, left-behind luggage, people running and people fighting. NOLA [8] is another dataset proposed to study continual learning in VAD. These datasets are all single-scene datasets which is our main focus in this paper.

There are also a number of datasets intended for anomaly detection across multiple scenes. ShanghaiTech [21] includes videos of 13 different scenes with anomalies such as cyclists, people with strollers, and people fighting. UCF-Crime [30] is another multi-scene dataset intended for weakly supervised version of video anomaly detection in which anomalous videos are used in addition to normal videos during training. Anomaly types include people fighting and explosions. UBnormal [1] is a multi-scene dataset consisting of synthetically generated scenes that include annotated anomalies in the training videos. Anomalous activity includes people running, falling, dancing and jaywalking.

The vast majority of anomalies in all of these datasets (with the exception of UCF-Crime) involve a single object, for example, a person walking in an unusual place, the appearance of a golf cart, a cyclist, or a person running. Such anomalies can be detected well (at least for temporal localization) by models that fundamentally work at the pixel level as evidenced by so many models that use pixel reconstruction error as a loss function for training [3,13,14,17,18,20,25,28,32,34,37,38]. Concerning the UCF-Crime dataset [30], it is designed for a very different version of video anomaly detection (multi-scene and weakly supervised) which does not correspond to the most common real-world surveillance application that we are most interested in. We hope to encourage methods that try to understand a scene at a higher level such as methods that model objects and their motions. Toward that end, a dataset that has more complex anomalies such as those involving the interaction of multiple objects will require modeling a scene at a higher level to be successful. This is the main motivation for introducing our new ComplexVAD dataset.

The novel algorithm we propose for detecting complex anomalies uses a scene graph to represent objects and their interactions in a video. A number of recent papers have also focused on addressing anomaly localization at the object-level [2, 5–7, 10, 11, 14, 35, 39]. These methods utilize pre-trained object detectors to first localize objects and then estimate if the detected objects are anomalous or not. There are many differences in the details of these methods compared to ours, especially in the representation of motion, but the most important difference is that these methods do not model the interactions among people/objects.

There have been a few past approaches that did model interactions among objects. Many of these methods also employed scene graphs to represent the interactions [4,9,31].

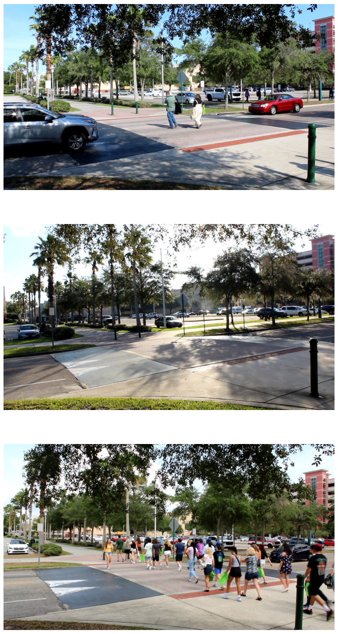

Figure 1. Nominal frame samples from the ComplexVAD dataset.

In [4], a simple model of object-relation-object triplets is used to model a scene, but unlike our method, there is no modeling of motion or trajectories. In our approach, the trajectory for each object is computed which allows unusual trajectories to be detected as anomalous. The approach of [9] used a scene graph to represent subject-predicate-object triplets in normal video and then compare those to ones found in test video. The main difference compared with our proposed approach is our use of an exemplar-based model of normal pairs of objects and our inclusion of object trajectories in the representation of objects in the scene graph. In [31], a scene graph was also used to represent objects and their interactions. The main difference with our approach is the method for computing distances between pairs of graph nodes and the specific attributes that are stored in the representation of each object.

The work of [33] modeled interactions between a person and an object using human-object interaction (HOI) vectors that does not use scene graphs. Normal HOI vectors are modeled with a Gaussian mixture. Low probability HOI vectors from test video can then be detected as anomalous.

Most of the methods that use object-level representations including our method are also interpretable. They can provide human-understandable explanations for detected anomalies. Explainability is a very important property for VAD methods to be adopted for real-world use.

3. ComplexVAD

To address the absence of complex anomalies in existing datasets, we introduce the ComplexVAD dataset. The dataset has 104 training and 113 test video sequences. All videos are recorded at the same location in a university campus showing a crosswalk, pedestrian sidewalks, and a two-lane street. Figure 1 shows some nominal frames. The video collection process lasted 5 months and videos were recorded during different periods including morning, noon, and afternoon. Since it is a campus environment, the scene tends to change frequently depending on the time and day. For this reason, for each day of the week, there is at least one hour of recording for morning, noon, and afternoon to represent the different states of the scene. It is a highly active and complex scene with people who walk, jog, or run; bikers, skateboarders, and scooter riders using the crosswalk and sidewalks; cars, buses, and golf carts using the car lanes. In addition, the background is not static among videos due to changing shadows, trees blowing in the wind or parking lots with varying numbers of parked cars. All faces were blurred using a face detector and Gaussian blurring to remove personally identifiable information.

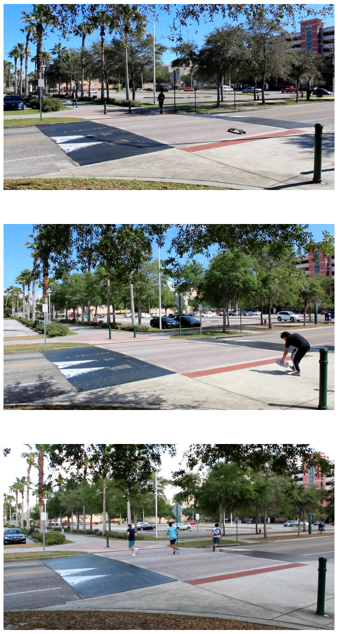

Figure 2. Samples of complex anomaly from the ComplexVAD dataset. In all samples, objects are nominal, but the interactions are anomalous. (Top) A skateboard moving alone violates the expectation of a nominal interaction between a person and skateboard. (Middle) Person carries an object and then leaves it on the ground. (Bottom) Three people walk together and then suddenly start to run in different directions.

ComplexVAD is a large dataset consisting of videos recorded in 1080x1920 resolution and at a rate of 30 frames per second. The training set includes videos ranging from 2.5 minutes to 13 minutes, with an average duration of 11 minutes. In the test set, the longest duration is 12.8 minutes, the shortest is 1.5 minutes, and the average duration is 7.9 minutes. When considering frames extracted from the original videos at 30 frames per second, there are 2,069,941 RGB

Dataset	Total Frames	Training Frames	Testing Frames	Anomalous Events	Anomaly Types	Ground Truth	Resolution	Complex Anomalies
UCSD Ped1	14,000	6800	7200	54	5	S, T	238 x 158	No
UCSD Ped2	4560	2550	2010	23	5	S, T	360x240	No
CUHK Avenue	30,652	15,328	15,324	47	5	S, T	640 x 360	No
IITB-Corridor	483,566	301,999	181,567	?	~10	T	1920x1080	No
NOLA	1,440,000	450,000	990,000	50	~10	S, T	1280 x 720	No
Street Scene	203,257	56,847	146,410	205	17	S, T	1280 x 720	No
ComplexVAD	**3,681,438**	**2,069,941**	**1,611,497**	**118**	**40**	**S, T**	**1920x1080**	**Yes**

Table 1. Characteristics of existing single scene video anomaly detection datasets compared to ComplexVAD. S and T denote Spatial and Temporal ground truth labels respectively.

frames for training and 1,611,497 RGB frames for testing, totaling 3,681,438 RGB video frames for the entire dataset. Due to potential complications and challenges in storage and distribution, ComplexVAD is publicly shared in video format. The comparison with existing VAD datasets can be seen in Table 1.

The aim of the ComplexVAD dataset is to showcase complex anomalies. We define a complex anomaly as an anomalous event resulting from the interaction between objects. Compared to anomalies presented in previous datasets, objects in a complex anomaly should be considered normal within the scene until their interaction occurs. Some examples of complex anomaly are presented in Figure 2. For instance, a person and a backpack are common objects in our dataset, but a person leaving their backpack on a sidewalk constitutes an anomaly resulting from the "leaving" action. Another example is a skateboard moving autonomously (due to a remote control), on a crosswalk. While skateboards are typically found in crosswalks with someone riding them, in this case, the usual interaction between the skateboard and a rider is absent.

Additionally, changes in interactions can lead to complex anomalies, such as a biker slowing down and stopping briefly in the middle of a crosswalk, where the typical interaction involves passing by without any interruption. The ComplexVAD dataset includes complex anomalies resulting from interactions between various objects such as pedestrians, cars, bikes, scooters, skates, sports balls, dogs, baseball bats, and trees. ComplexVAD includes 118 anomalies from 40 diverse types of complex anomalies, which are detailed in the supplementary document.

The ComplexVAD dataset is publicly available under the CC-BY-SA-4.0 license.[1] We provide ground truth annotations in a form which can easily be used for several types of evaluation criteria such as region-based and track-based as well as frame-level. Annotations are provided for each testing video in the form of bounding boxes around each object that is a part of the anomalous event in each frame. In addition, a track id is assigned to each bounding box so that each anomalous event can be represented as a track of bounding boxes. Due to the nature of our dataset, each frame can have more than one anomaly labeled.

4. Detecting Complex Anomalies

We propose a novel method to detect complex anomalies. Our method can be divided into three stages. First, we derive graph representations of all frames in the training dataset. Second, for all pairs of nodes (i.e., pairs of objects that are close in terms of 3D distance) and isolated nodes (i.e., objects that are not close to any other object) in the training set, we use an exemplar selection algorithm to select a subset of unique node pairs and isolated nodes to form an exemplar set. Third, we compare the distances between node pairs in the test set and node pairs in the exemplar set. The same is done between isolated test nodes and isolated nodes in the exemplar set. Any test instance with a high distance to every exemplar is considered anomalous. In the following sections, we will discuss the stages of our method in detail.

4.1. Frame to graph

For a given dataset, we transform each frame of each video into an undirected graph. The pipeline of our frame to graph transformation is depicted in Figure 3. Our first step is to use an object detector to extract objects. Note that, for our approach, the object detector plays a fundamental role. It is important to evaluate the object detector's capability in the scene and choose the most suitable one. In our initial experiments, we found that Detectron2 [36] has the most accurate object detection. Hence, Detectron2, which is trained on the COCO dataset, is used in our implementation.

A video V is a collection of M frames $\{F_i\}_{i=1}^{M}$, such that $V = [F_1, F_2, ..., F_M]$. We send each frame F_i to the object detector O, which returns X number of detected objects. For each object o, the location $l = (x_o, y_o)$ which is the x and y coordinates of the center of the object, $b = (w_o, h_o)$ which is the width and height of the bounding box for the object, and class id c. The output of the object detector is then $O(F) = [o_1, o_2, ..., o_X]$, where each object o_i is represented by $o_i = [b, c, l]$.

After detecting objects in a frame, they are then tracked using an object tracker, namely ByteTrack [40]. Each detected object o is sent to the object tracker, which returns x and y coordinates for that object in the subsequent frames. In our method, we track objects for 30 frames. Therefore, for every object, we acquire the trajectory $\theta = \{(x_1, y_1), (x_2, y_2), ..., (x_{30}, y_{30})\}$.

[1] www.merl.com/research/downloads/ComplexVAD

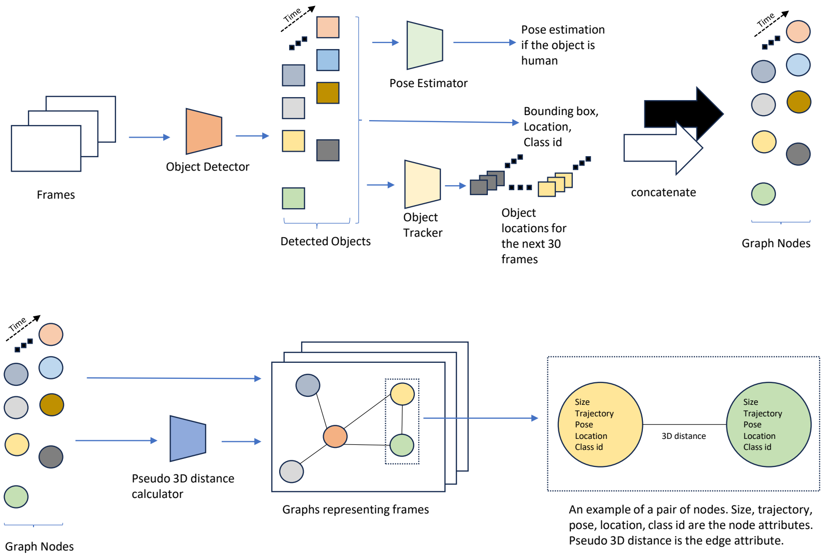
Frames
Object Detector
Time
Detected Objects
Pose Estimator
Pose estimation if the object is human
Bounding box, Location, Class id
Object Tracker
Object locations for the next 30 frames
concatenate
Time
Graph Nodes
Time
Graph Nodes
Pseudo 3D distance calculator
Graphs representing frames
Size
Trajectory
Pose
Location
Class id
3D distance
Size
Trajectory
Pose
Location
Class id
An example of a pair of nodes. Size, trajectory, pose, location, class id are the node attributes. Pseudo 3D distance is the edge attribute.

Figure 3. The pipeline of our method for frame to graph generation with the help of an object detector, object tracker and pose estimator.

In addition to the object detector and object tracker, we also use a pose estimator to obtain the pose information of human objects. Any object o identified as human is sent to the pose estimator to obtain the pose vector $p = \{(x_1, y_1), (x_2, y_2), ..., (x_{17}, y_{17})\}$, which contains the locations of 17 key points on the human body. In our experiments, we use the human pose estimation method included in Detectron2 [36].

We concatenate the features that we extract from an object o to form a graph node $n = [b, c, l, \theta, p]$ where node attribute b is the bounding box size, c is the class id, l is the location of the center of the object, θ is the trajectory vector and p is the pose vector.

Next, we need to find edges between nodes/objects that are likely to be interacting. Past approaches to building scene graphs [4, 31] have used a deep network, usually trained on the Visual Genome dataset [15], to estimate relations between objects. We found such approaches to produce too many inconsistent relations which can cause false positive anomalous detections. Instead we use the simple and more robust method of assigning an edge between objects if they are close to each other (if their distance in 3D space is below a threshold). Thus, to determine which nodes to connect in the graph we need to calculate the 3D distances between each pair of nodes. To calculate the 3D distance we need to derive the 3D coordinates of the node locations by estimating a pseudo-depth since we do not have access to actual depth estimates. Given two nodes n_1 and n_2, we have 2D coordinates $l_1 = (x_1, y_1)$ and $l_2 = (x_2, y_2)$. Then we define a relative depth, z, between two nodes by taking the absolute difference of y values such that $z = |y_1 - y_2|$. This estimate of pseudo-depth assumes that objects are resting on the ground plane and the ground plane is farther from the camera the closer it is to the top of the image. The 3D distance d can then be calculated by taking the Euclidean distance between 3D coordinates (x_1, y_1, z) and $(x_2, y_2, 0)$. Any node pair that has a 3D distance d smaller than a predetermined threshold h is connected with an edge E. Due to applying the threshold, not every single node is necessarily connected to another node, which leads to having isolated nodes in addition to node pairs.

At the end of our frame to graph transformation, a frame F which is a collection of objects $\{o_1, o_2, ..., o_X\}$ where X is the total number of objects extracted by the object detector, can be represented as a graph $G = (N, E)$ where N is the collection of graph nodes $[n_1, n_2, ..., n_X]$, and E is the graph edges between connected nodes. Similarly, a video $V = [F_1, F_2, ..., F_M]$ which contains M number of frames F, can be represented as collection of graphs: $V = [G_1, G_2, ..., G_M]$.

4.2. Model building from nominal video

For a given nominal video, frames are processed using our method described in 4.1 and transformed into graphs. For all frames in a video, we collect all pairs of nodes that are connected by an edge into one set and all isolated nodes (not connected to any other node) into another set. Then for each of the sets, independently, we run an exemplar selection algorithm which selects a subset of the elements of the set such that no two members of the subset are near each other according to a distance function (described below). The intuition behind exemplar selection is to simply remove redundant (or nearly redundant) elements from the set leaving behind a compact, representative subset of exemplars. We use the same exemplar selection algorithm as described in [26]. Given a set S, the exemplar selection algorithm proceeds as follows: (1) Initialize the exemplar set to NULL. (2) Add the first element of S to the exemplar set. (3) For each subsequent element of S, find its distance to the nearest instance in the exemplar set. If this distance is above a threshold, th, then add the element to the exemplar set.

As mentioned before, we run exemplar selection separately on the set of all isolated nodes found in the graphs of all frames and the set of all pairs of nodes found in the graphs of all frames. To use the exemplar selection algorithm we need to define a distance between two isolated nodes and a distance between two node pairs. We will start with the distance between two isolated nodes.

A graph node, n, is a high-level representation of an object which includes the attributes $[b, c, l, \theta, p]$ where b is the bounding box size, c is the class identifier, l is the location, θ is the trajectory vector and p is the pose vector. For two given nodes n_1 and n_2 with attributes $[b_1, c_1, l_1, \theta_1, p_1]$ and $[b_2, c_2, l_2, \theta_2, p_2]$, we define a distance between each node attribute as follows.

The location distance is the Euclidean distance between $l_1 = (x_1, y_1)$ and $l_2 = (x_2, y_2)$:

$$L(n_1, n_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{1}$$

The distance between bounding box sizes $b_1 = (w_1, h_1)$ and $b_2 = (w_2, h_2)$ is calculated by taking the Euclidean distance between each bounding box width and height normalized by the minimum width and height:

$$S\left(n_1, n_2\right) = \sqrt{\frac{\left(w_1 - w_2\right)^2}{\min\left(w_1, w_2\right)} + \frac{\left(h_1 - h_2\right)^2}{\min\left(h_1, h_2\right)}} \tag{2}$$

The class distance is set to 0 if the nodes have the same class id; otherwise, it is set to 1:

$$C(n_1, n_2) = \begin{cases} 0 & \text{if } c_1 = c_2 \\ 1 & \text{if } c_1 \neq c_2, \end{cases} \tag{3}$$

For two pose vectors, $P_1 = \{(x_{1,1}, y_{1,2}), (x_{1,2}, y_{1,2}), ..., (x_{1,17}, y_{1,17})\}$ and $P_2 = \{(x_{2,1}, y_{2,2}), (x_{2,2}, y_{2,2}), ..., (x_{2,17}, y_{2,17})\}$, The pose distance is

$$P\left(n_1, n_2\right) = \sum_{t=2}^{17} \frac{\left|dp_{1,t} - dp_{2,t}\right|}{\max\left(\min\left(dp_{1,t}, dp_{2,t}\right), 1\right)} \tag{4}$$

where

$$dp_{1,t} = \sqrt{(x_{1,1} - x_{1,t})^2 + (y_{1,1} - y_{1,t})^2} \tag{5}$$

and

$$dp_{2,t} = \sqrt{(x_{2,1} - x_{2,t})^2 + (y_{2,1} - y_{2,t})^2} \tag{6}$$

are the distances from the first pose keypoint to the tth pose keypoint, for each pose vector, respectively. The max function in the denominator of Equation 4 insures that the denominator is not less than 1 to prevent division by zero.

For two node trajectories $\theta_1 = \{(x_{1,1}, y_{1,1}), (x_{1,2}, y_{1,2}), ..., (x_{1,30}, y_{1,30})\}$ and $\theta_2 = \{(x_{2,1}, y_{2,1}), (x_{2,2}, y_{2,2}), ..., (x_{2,30}, y_{2,30})\}$, the trajectory distance is the sum of the L1 distances between the displacements of the first node and the displacements of the second node normalized by the minimum displacement:

$$\begin{aligned} \Theta\left(\theta_1, \theta_2\right) = \sum_{t=1}^{T-1} & \frac{\left|dx_{1,t} - dx_{2,t}\right|}{\max\left(\min\left(dx_{1,t}, dx_{2,t}\right), 1\right)} + \\ & \frac{\left|dy_{1,t} - dy_{2,t}\right|}{\max\left(\min\left(dy_{1,t}, dy_{2,t}\right), 1\right)} \end{aligned} \tag{7}$$

where $dx_1(t) = x_{1,t} - x_{1,t+1}$, $dx_2(t) = x_{2,t} - x_{2,t+1}$, $dy_1(t) = y_{1,t} - y_{1,t+1}$, $dy_2(t) = y_{2,t} - y_{2,t+1}$. T is the number of frames in a track which is set to 30 in our experiments. The max function in the denominator is used to avoid division by zero.

Given these distances between attributes of two nodes, the final distance between two isolated nodes is calculated as follows:

$$\begin{aligned} D(n_1, n_2) = \max(& \frac{L(n_1, n_2) - \mu_L}{\sigma_L}, \\ & \frac{S(n_1, n_2) - \mu_S}{\sigma_S}, \frac{C(n_1, n_2) - \mu_C}{\sigma_C}, \\ & \frac{P(n_1, n_2) - \mu_P}{\sigma_P}, \frac{\Theta(n_1, n_2) - \mu_\Theta}{\sigma_\Theta}) \end{aligned} \tag{8}$$

where the μ and σ parameters are normalization constants for each distance which make all the distances comparable. We discuss how these normalization constants are chosen in the supplementary material.

A node pair N is a combination of two nodes which are connected with an edge. Between two node pairs $N_1 = (n_1, n_2)$ and $N_2 = (n_3, n_4)$, the distance is calculated as follows:

$$\begin{aligned} & D_{pair}(N_1, N_2) = \\ & \quad \min(\max(D(n_1, n_3), D(n_2, n_4)), \\ & \qquad \max(D(n_1, n_4), D(n_2, n_3))). \end{aligned} \tag{9}$$

The intuition behind this distance is firstly that we do not know whether n_1 corresponds to n_3 or n_4 (and similarly whether n_2 corresponds to n_3 or n_4) so we need to try both pairings and take the minimum distance. This corresponds to the outer min function. For a given correspondence, the overall distance between the two node pairs is the maximum distance between the corresponding nodes from each pair. This is represented by the inner max functions. Further details on distance normalization and exemplar selection provided in the supplementary document.

4.3. Complex anomaly detection in test video

After the first stage of obtaining exemplar sets from nominal videos, the next step is detecting anomalies in testing video of the same scene. As with nominal videos, the pipeline that is described in 4.1 is also followed for test videos, to generate graphs from objects detected in each frame. The same object attributes are computed for each object: location, bounding box size, class ID, trajectory and if the object is a person, a pose vector. Given a scene graph for a test frame, anomaly scores are computed for every pair of connected nodes and for every isolated node. The anomaly score, AS, for a test isolated node, n, is the distance to the nearest exemplar in the isolated node exemplar set:

$$AS(n, \mathcal{E}_{iso}) = \min_{n_e \in \mathcal{E}_{iso}} D\left(n, n_e\right) \tag{10}$$

Similarly, the anomaly score (AS) for a pair of nodes $N = (n_1, n_2)$ is the distance to the nearest pair of nodes from the node-pair exemplar set:

$$AS(N, \mathcal{E}_{pair}) = \min_{N_e \in \mathcal{E}_{pair}} D_{pair}\left(N, N_e\right) \tag{11}$$

The nearest neighbor searches in Equations 10 and 11 are generally fast because the number of exemplars is typically small, but can easily be sped up with one of the many efficient nearest neighbor techniques [23].

5. Experiments

5.1. Experimental settings and evaluation criteria

We evaluate our proposed method and two other state-of-the-art video anomaly detection methods, namely Memory-augmented Deep Autoencoder (MemAE) [12] and Explainable Video Anomaly Localization (EVAL) [29], on the ComplexVAD dataset.

For our method, ByteTrack [40] and Detectron2 [36] were used as object tracker and object detector. The pose estimator module of Detectron2 is also used for pose estimations. Using the method described in Section 4, exemplar sets are extracted for all of the training videos. We choose a threshold $th = 0.65$ for exemplar selection that resulted in a modest number of total exemplars selected. From past work that used exemplar-based models, this threshold mainly effects model size and has a small effect on test accuracy.

To test MemAE on the ComplexVAD dataset, we used the same proposed hyper-parameter and model structure settings as described in [12]. The ComplexVAD dataset is resized to 256x256 to be compatible with the existing settings. Additionally, since the ComplexVAD dataset has an extensive number of frames, we sub-sampled every third frame of the dataset for training and testing to gain computational speed during training and testing. Finally, the MemAE model is trained with the training split on NVIDIA 4090.

Method	RBDC	TBDC	Frame
MemAE [12]	0.0005	0	0.58
EVAL [29]	0.10	0.62	0.54
Ours	**0.19**	**0.64**	**0.60**

Table 2. The table reports the area under the curve (AUC) for our method and two recent VAD methods using the RBDC, TBDC and Frame-Level evaluation criteria on ComplexVAD.

To test EVAL on the ComplexVAD dataset, we subsampled every other frame (for an effective frame rate of 15 fps), and used 10 frame video volumes with 256x256 pixel spatial region sizes which roughly corresponds to the average height of a person in this dataset. The remainder of the setup and parameters were exactly as described in [29].

We use the Region-Based Detection Criterion (RBDC) and the Track-Based Detection Criterion (TBDC) as proposed in [26] as our primary evaluation criteria and report the area under the curve (AUC) for false positive rates per frame from 0 to 1. We also report frame-level AUC [22] scores. As highlighted in previous works [26] frame-level AUC only evaluates temporal accuracy and disregards spatial localization of anomalies. Whereas, RBDC and TBDC measure a method's capacity to accurately identify anomalous spatio-temporal regions within a given video sequence, however, we also report frame-level AUC scores of the methods for completeness, as well as comparisons to older methods.

In order to get RBDC and TBDC numbers for MemAE we used the following procedure. For each anomaly score threshold, we create a mask of all pixels with anomaly scores above threshold. We then find connected components of anomalous pixels. This give us anomalous regions. For each connected component with at least 10 pixels, we compute the minimum bounding box encompassing that component. This yields a set of anomalous bounding boxes that can be used for computing RBDC and TBDC numbers.

5.2. Results

The main results of our method as well as the EVAL [29] and MemAE [12] methods using the three different evaluation criteria described above are reported in Table 2. We can see that our scene-graph based method outperforms the other two recent methods under all criteria. The MemAE method does very poorly for the two criteria that measure spatial localization. This implies that the regions of an image that MemAE predicts as anomalous are usually normal.

We also show some visualizations of the output of our method on some frames from ComplexVAD in Figures 4 and 5. Figure 4 shows 5 frames from a test video in ComplexVAD in which a person carrying an object places the object on the ground and continues walking. This is an example of a "left-behind object" anomaly and is correctly detected by our method. Figure 5 shows frames from three other anomalies, including a dog walking without a person holding its leash,

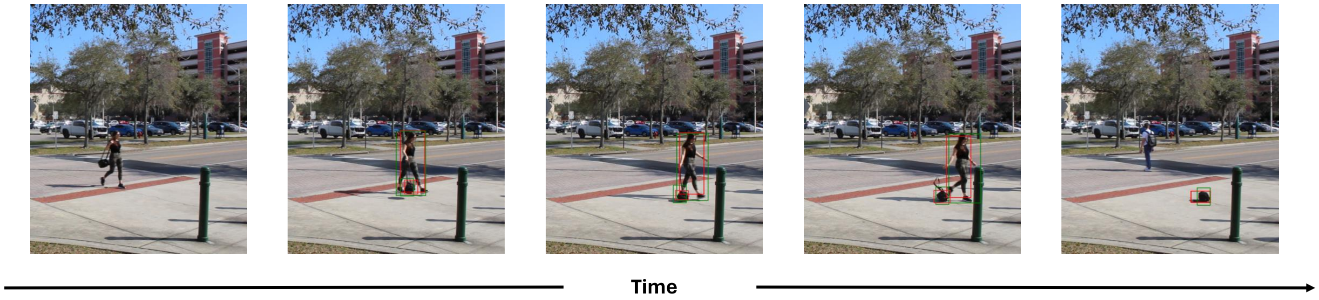

Figure 4. A person who drops a bag on the street is detected as an anomaly with our method. The detection starts with the action of "drop". After the object interaction ends, the dropped object continues to be detected as an anomaly. Ground truth labels and detection boxes are represented with green and red colors, respectively.

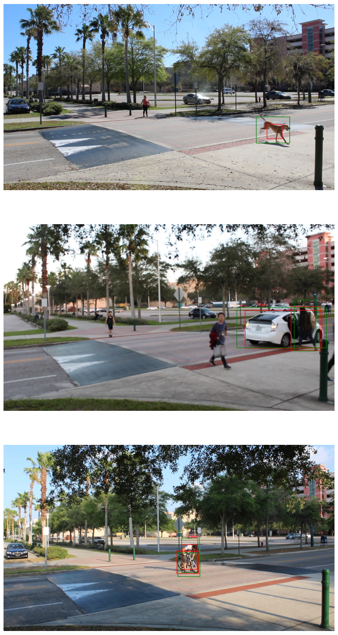

Figure 5. Detected interaction anomalies with our method. (Top) A dog without a walker. (Middle) Car picks up a passenger on crosswalk. (Bottom) Bicycle stops briefly in the middle of the road. Ground truth labels and detection boxes are represented with green and red colors, respectively.

a person getting into a car in the middle of a crosswalk, and a person stopping on a bicycle in the middle of a crosswalk. These are all successfully detected by our method. The first anomaly is particularly interesting because it required the system to notice that in the nominal training videos, dogs always appeared with a person walking them on a leash. It is the lack of the expected interaction that is anomalous here.

6. Future Work and Discussions

Complex video anomaly detection is a new direction in research and according to the baseline results, there is plenty of room for improvement for this difficult problem. Since the limitations of the object detector directly affect our method's accuracy, investigating the effects of different object detectors may lead to improved accuracy. Also, because our method only models the interactions of pairs of objects, expanding this to modeling three or more objects interacting may also lead to accuracy gains. Another interesting direction for further research is explainability. As shown by other papers [4, 9, 29, 31], the use of object-level models and scene graphs allow for human-understandable explanations to be automatically generated to explain why certain activities are detected as anomalous.

Our interest in introducing complex video anomaly detection is to make this research area more applicable in the real world. An important practical issue that real systems must handle is adversarial attacks which have been demonstrated to effectively deceive video anomaly detection systems [24]. Therefore, robustness against such attacks should be a major concern in this new field.

7. Conclusion

Existing video anomaly detection datasets demonstrate anomalous activities that mainly involve a single object or actor. However, in the real world, anomalies are often caused by the interactions between objects. In this work, we introduce a new video anomaly detection dataset, ComplexVAD, with many diverse types of interaction-based anomalies. With the introduction of ComplexVAD, we anticipate that more research will be directed towards detecting complex anomalies in video.

In addition to a new dataset, we also introduce a novel method to detect complex anomalies. With our method and two other state-of-the art video anomaly detection methods, we provide baseline scores on ComplexVAD. Results indicate that our method outperforms the existing methods but there is still room for improvement with further research.

Acknowledgement

We sincerely thank Lokman Bekit *(lbekit@usf.edu)* for his valuable contributions to the data collection process.

References

[1] Andra Acsintoae, Andrei Florescu, Mariana-Iuliana Georgescu, Tudor Mare, Paul Sumedrea, Radu Tudor Ionescu, Fahad Shahbaz Khan, and Mubarak Shah. Ubnormal: New benchmark for supervised open-set video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[2] Antonio Barbalau, Radu Tudor Ionescu, Mariana-Iuliana Georgescu, Jacob Dueholm, Bharathkumar Ramachandra, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund, and Mubarak Shah. Ssmtl++: Revisiting self-supervised multi-task learning for video anomaly detection. *Computer Vision and Image Understanding*, 229:103656, 2023. 2

[3] Yunpeng Chang, Zhigang Tu, Wei Xie, and Junsong Yuan. Clustering driven deep autoencoder for video anomaly detection. In *European Conference on Computer Vision*, pages 329–345. Springer, 2020. 2

[4] Nicholas F.Y. Chen, Zhiyuan Du, and Khin Hua Ng. Scene graphs for interpretable video anomaly classification. In *NIPS*, 2018. 2, 3, 5, 8

[5] Keval Doshi and Yasin Yilmaz. Any-shot sequential anomaly detection in surveillance videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 934–935, 2020. 2

[6] Keval Doshi and Yasin Yilmaz. An efficient approach for anomaly detection in traffic videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4236–4244, 2021. 2

[7] Keval Doshi and Yasin Yilmaz. A modular and unified framework for detecting and localizing video anomalies. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3982–3991, 2022. 2

[8] Keval Doshi and Yasin Yilmaz. Rethinking video anomaly detection-a continual learning approach. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3961–3970, 2022. 1, 2

[9] Keval Doshi and Yasin Yilmaz. Towards interpretable video anomaly detection. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2023. 2, 3, 8

[10] Mariana-Iuliana Georgescu, Antonio Barbalau, Radu Tudor Ionescu, Fahad Shahbaz Khan, Marius Popescu, and Mubarak Shah. Anomaly detection in video via self-supervised and multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12742–12752, 2021. 2

[11] Mariana Iuliana Georgescu, Radu Ionescu, Fahad Shahbaz Khan, Marius Popescu, and Mubarak Shah. A background-agnostic framework with adversarial training for abnormal event detection in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[12] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 7

[13] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 733–742, 2016. 2

[14] Radu Tudor Ionescu, Fahad Shahbaz Khan, Mariana-Iuliana Georgescu, and Ling Shao. Object-centric auto-encoders and dummy anomalies for abnormal event detection in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7842–7851, 2019. 2

[15] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123, 2017. 5

[16] 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, 2013. 2

[17] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection–a new baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6536–6545, 2018. 2

[18] Zhian Liu, Yongwei Nie, Chengjiang Long, Qing Zhang, and Guiqing Li. A hybrid video anomaly detection framework via memory-augmented flow reconstruction and flow-guided frame prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13588–13597, 2021. 2

[19] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2720–2727, 2013. 1, 2

[20] Yiwei Lu, Frank Yu, Mahesh Kumar Krishna Reddy, and Yang Wang. Few-shot scene-adaptive anomaly detection. In *European Conference on Computer Vision*, pages 125–141. Springer, 2020. 2

[21] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. In *Proceedings of the IEEE international conference on computer vision*, pages 341–349, 2017. 2

[22] 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. IEEE, 2010. 1, 2, 7

[23] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *International Conference on Computer Vision Theory and Applications (VISAPP'09)*, 2009. 7

[24] Furkan Mumcu, Keval Doshi, and Yasin Yilmaz. Adversarial machine learning attacks against video anomaly detection systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 206–213, 2022. 8

[25] Trong-Nguyen Nguyen and Jean Meunier. Anomaly detection in video sequence with appearance-motion correspondence.

In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1273–1283, 2019. 2

[26] Bharathkumar Ramachandra and Michael Jones. Street scene: A new dataset and evaluation protocol for video anomaly detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2569–2578, 2020. 1, 2, 5, 7

[27] Royston Rodrigues, Neha Bhargava, Rajbabu Velmurugan, and Subhasis Chaudhuri. Multi-timescale trajectory prediction for abnormal human activity detection. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020. 1, 2

[28] Chenrui Shi, Che Sun, Yuwei Wu, and Yunde Jia. Video anomaly detection via sequentially learning multiple pretext tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 2

[29] Ashish Singh, Michael J. Jones, and Erik Learned-Miller. Eval: Explainable video anomaly localization. In *CVPR*, 2023. 7, 8

[30] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6479–6488, 2018. 2

[31] Che Sun, Yunde Jia, Yao Hu, and Yuwei Wu. Scene-aware context reasoning for unsupervised abnormal event detection in videos. In *Proceedings of the 28th ACM International Conference on Multimedia (ACMMM)*, 2020. 2, 3, 5, 8

[32] Shengyang Sun and Xiaojin Gong. Hierarchical semantic contrast for scene-aware video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[33] Stanislaw Szymanowicz, James Charles, and Roberto Cipolla. X-man: Explaining multiple sources of anomalies in video. In *CVPR*, 2021. 3

[34] Chenxu Wang, Yanxin Yao, and Han Yao. Video anomaly detection method based on future frame prediction and attention mechanism. In *IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2021. 2

[35] Guodong Wang, Yunhong Wang, Jie Qin, Dongming Zhang, Xiuguo Bao, and Di Huang. Video anomaly detection by solving decoupled spatio-temporal jigsaw puzzles. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X*, pages 494–511. Springer, 2022. 2

[36] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 4, 5, 7

[37] Zhiwei Yang, Jing Liu, Zhaoyang Wu, Peng Wu, and Xiaotao Liu. Video event restoration based on key frames for video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[38] Zhiwei Yang, Jing Liu, Zhaoyang Wu, Peng Wu, and Xiaotao Liu. Video event restoration based on keyframes for video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[39] Guang Yu, Siqi Wang, Zhiping Cai, En Zhu, Chuanfu Xu, Jianping Yin, and Marius Kloft. Cloze test helps: Effective video anomaly detection via learning to complete video events. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 583–591, 2020. 2

[40] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. 2022. 4, 7