6-2023

Unbiased multiple instance learning for weakly supervised video anomaly detection

Hui LYU
Singapore Management University, huilyu@smu.edu.sg

Zhongqi YUE

Qianru SUN
Singapore Management University, qianrusun@smu.edu.sg

Bin LUO

Zhen CUI

See next page for additional authors

Unbiased Multiple Instance Learning for Weakly Supervised Video Anomaly Detection

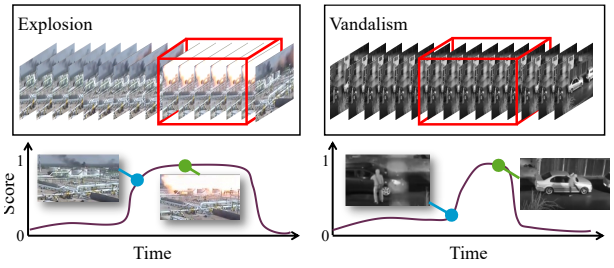
Hui Lv[1,3,4], Zhongqi Yue[4], Qianru Sun[3], Bin Luo[2], Zhen Cui[1]*, Hanwang Zhang[4]
[1]PCAlab, Nanjing University of Science and Technology [2]Alibaba Group
[3]Singapore Management University [4]Nanyang Technological University
[1]{hubrthui, zhen.cui}@njust.edu.cn, [2]luwu.lb@alibaba-inc.com
[3]qianrusun@smu.edu.sg, [4]{yuez0003, hanwangzhang}@ntu.edu.sg

Abstract

Weakly Supervised Video Anomaly Detection (WSVAD) is challenging because the binary anomaly label is only given on the video level, but the output requires snippet-level predictions. So, Multiple Instance Learning (MIL) is prevailing in WSVAD. However, MIL is notoriously known to suffer from many false alarms because the snippet-level detector is easily biased towards the abnormal snippets with simple context, confused by the normality with the same bias, and missing the anomaly with a different pattern. To this end, we propose a new MIL framework: Unbiased MIL (UMIL), to learn unbiased anomaly features that improve WSVAD. At each MIL training iteration, we use the current detector to divide the samples into two groups with different context biases: the most confident abnormal/normal snippets and the rest ambiguous ones. Then, by seeking the invariant features across the two sample groups, we can remove the variant context biases. Extensive experiments on benchmarks UCF-Crime and TAD demonstrate the effectiveness of our UMIL. Our code is provided at https://github.com/ktr-hubrt/UMIL.

1. Introduction

Video Anomaly Detection (VAD) aims to detect events among video sequences that deviate from expectation, which is widely applied in real-world tasks such as intelligent manufacturing [8], TAD surveillance [9, 22] and public security [25, 30]. To learn such a detector, conventional fully-supervised VAD [1] is impractical as the scattered but diverse anomalies require extremely expensive labeling cost. On the other hand, unsupervised VAD [3, 11, 13, 35, 42] by only learning on normal videos to detect open-set anomalies often triggers false alarms, as it is essentially ill-posed to define what is normal and abnormal by giving only normal videos without any prior knowledge. Hence, we are interested in a more practical setting: Weakly Supervised VAD (WSVAD) [12, 43], where only video-level binary labels (*i.e.*, normal *vs.* abnormal) are available.



(a) (b)

Figure 1. Two anomalies of Explosion and Vandalism are illustrated. Among each video sequence, we use red boxes to highlight the ground-truth anomaly regions as in the first row. The corresponding anomaly curves of an MIL-based model are depicted below. False alarms and real anomalies are linked to the curves with blue arrows and green arrows respectively. Best viewed in color.

In WSVAD, each video sequence is partitioned into multiple snippets. Hence, all the snippets are normal in a normal video, and at least one snippet contains the anomaly in an abnormal one. The goal of WSVAD is to train a snippet-level anomaly detector using video-level labels. The mainstream method is Multiple Instance Learning (MIL) [22, 30]—multiple instances refer to the snippets in each video, and learning is conducted by decreasing the predicted anomaly score for each snippet in a normal video, and increasing that only for the snippet with the largest anomaly score in an abnormal video. For example, Figure 1a shows an abnormal video containing an explosion scene, and the detector is trained by MIL to increase the anomaly score for the most anomalous explosion snippet (green link).

However, MIL is easily biased towards the simplest context shortcut in a video. We observe in Figure 1a that the de-

*Corresponding author

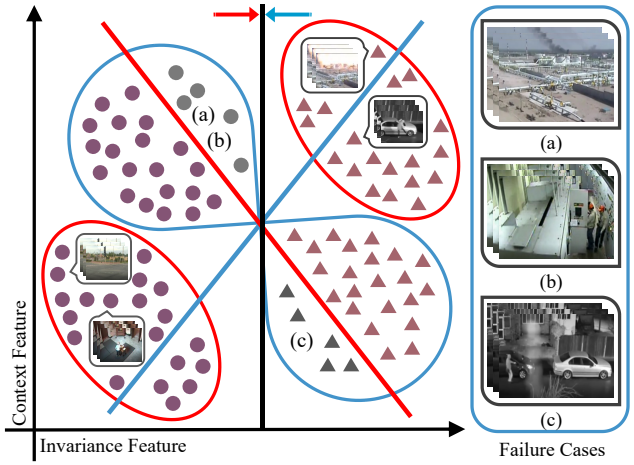


Figure 2. Red: Confident Set, Blue: Ambiguous Set. ●: Normal sample, ▲: Abnormal sample, Gray instances: Failure cases. The red line denotes the classifier trained under MIL. The invariant classifier (black line) can be learned by combining confident snippets learning in MIL (red line) and the ambiguous snippets clustering (blue line). Best viewed in color.

tector is biased to smoke, as the pre-explosion snippet with only smoke is also assigned a large anomaly score (blue link). This biased detector can trigger false alarms on smoke snippets without anomaly, *e.g.*, a smoking chimney. Moreover, it could also fail in videos with multiple anomalies of different contexts. In Figure 1b, the video records two men vandalizing a car, where only the second one has substantial motions. We notice that the two snippets of them have large differences in the anomaly scores, and only the latter is predicted as an anomaly. This shows that the detector is biased to the drastic motion context while being less sensitive to the subtle vandalism behavior, which is the true anomaly.

The root of MIL's biased predictions lies in its training scheme with biased sample selection. As shown in Figure 2, the bottom-left cluster (denoted as the red ellipse) corresponds to the confident normal snippets, *e.g.*, an empty crossroad or an old man standing in a room, which are either from normal videos as the ground truth or from abnormal videos but visually similar to the ground-truth ones. On the contrary, the top-right cluster denotes the confident abnormal ones, which not only contain the true anomaly features (*e.g.*, explosion and vandalism) but also include the context features commonly appearing with anomaly under a context bias (*e.g.*, smoke and motions). In MIL, the trained detector is dominated by the confident samples, corresponding to the top-right cluster with the abnormal representation and the bottom-left cluster with the normal representation. Hence the learned detector (red line) inevitably captures the context bias in the confident samples. Consequently, the biased detector generates ambiguous predictions on snippets with a different context bias (the red line mistakenly crossing the blue points), *e.g.*, smoke but normal (industrial exhaust in Figure 2a), substantial motion but normal (equipment maintenance in Figure 2b), or subtle motion but abnormal (vandalizing the rear-view mirror in Figure 2c), leading to the aforementioned failure cases.

To this end, we aim to build an unbiased MIL detector by training with both the confident abnormal/normal and the ambiguous ones. Specifically, at each UMIL training iteration, we divide the snippets into two sets using the current detector: 1) the confident set with abnormal and normal snippets and 2) the ambiguous set with the rest snippets, *e.g.*, the two sets are enclosed with red circles and blue circles in Figure 2, respectively. The ambiguous set is grouped into two unsupervised clusters (*e.g.*, the two blue circles separated by the blue line) to discover the intrinsic difference between normal and abnormal snippets. Then, we seek an invariant binary classifier between the two sets that separate the abnormal/normal in the confident set and the two clusters in the ambiguous one. The rationale of the proposed invariance pursuit is that the snippets in the ambiguous set must have a different context bias from the confident set, otherwise, they will be selected into the same set. Therefore, given a different context but the same true anomaly, the invariant pursuit will turn to the true anomaly (*e.g.*, the black line).

Overall, we term our approach as **Unbiased MIL (UMIL)**. Our contributions are summarized below:

- UMIL is a novel WSVAD method that learns an unbiased anomaly detector by pursuing the invariance across the confident and ambiguous snippets with different context biases.
- Thanks to the unbiased objective, UMIL is the first WSVAD method that combines feature fine-tuning and detector learning into an end-to-end training scheme. This leads to a more tailored feature representation for VAD.
- UMIL is equipped with a fine-grained video partitioning strategy for preserving the subtle anomaly information in video snippets.
- These contribute to the improved performance over the current state-of-the-art methods on UCF-Crime [30] (1.4% AUC) and TAD [22] (3.3% AUC) benchmarks. Note that UMIL brings more than 2% AUC gain compared with the MIL baseline on both datasets, which justifies the effectiveness of UMIL.

2. Related Work

The research lineup of video anomaly detection falls into two classes: unsupervised and weakly-supervised settings. **Unsupervised methods** include the ones that only use unlabelled training data or directly train and test on testing data. Del *et al*. [5] proposed to detect changes on a sequence of video data to detect unique frames. Tudor *et al*. [32] intro-

duced unmasking technology [10] to iteratively train a binary classifier to distinguish the most discriminant features. Lately, Zaheer *et al*. [40] exploited the low frequency of anomalies by building a cross-supervision between a generator and a discriminator. There are also One-Class Classification (OCC) methods assume the availability of normal training data only and approach the problem in an unsupervised manner. Typically, researchers fit a model with only normal data, then detect anomalies by distinguishing the events that deviate from the model. Early works used handcrafted appearance and motion features [2, 3, 18, 23, 24]. Thanks to the impressive progress of deep learning, recent works used the features from pre-trained deep neural networks and built an anomaly classifier upon them [6, 27]. There are also methods for self-supervised feature learning [28, 37], where a popular approach is by temporal prediction [15, 20, 36]. However, unsupervised methods suffer from false alarms for unseen normal patterns, since it is impossible to collect all kinds of normality in one dataset.

Weakly-supervised methods exploit both normal and abnormal training data with weak annotations only on the video-level [30]. Multiple instance learning (MIL) is the mainstream paradigm that uses video-level labels for training snippet-level anomaly detectors [7, 30, 44]. Generally, they embrace the two-stage anomaly detection pipeline, which performs anomaly detection upon pre-extracted features. In particular, Zhong *et al*. [43] considered the WSVAD task as supervised learning under noise labels and they designed an alternate training procedure to enhance the discrimination of action classifiers. Lv *et al*. [22] focused on anomaly localization and proposed a higher-order context model as well as a margin-based MIL loss. Tian *et al*. [31] investigated the feature magnitude to facilitate anomaly detection and selected the instances of top-k scores to better represent the video for MIL. Li *et al*. [12] proposed multiple sequence learning, where consecutive snippets with high anomaly scores are selected in MIL learning. They attempted to improve the sample selection for improving MIL, whose biased nature is not changed yet. In this paper, our unbiased MIL framework is the first effort on removing the context bias [38, 39] in WSVAD. In addition, we integrate feature representation fine-tuning and anomaly detector learning into an end-to-end training fashion.

3. Method

In Weakly Supervised Video Anomaly Detection (WSVAD), each training video is annotated with a binary anomaly label $y \in \{0, 1\}$ (*i.e*., normal or abnormal) and partitioned into m snippets. We denote $\mathbf{x}_i, i \in \{1, \dots, m\}$ as the feature of the i-th snippet in the video extracted by a backbone parameterized by θ. The goal of WSVAD is to train a snippet-level anomaly classifier $f(\mathbf{x}_i)$ predicting the probability of the snippet being positive (abnormal).

3.1. From MIL to Unbiased MIL

The mainstream method in WSVAD is Multiple Instance Learning (MIL). In MIL, the backbone θ is pre-trained (*e.g*., on Kinetics400 [4]) and frozen in training. It aims to learn f so as to predict the most anomalous snippet in a normal video (*i.e*., $y = 0$) as normal, and that in an abnormal video (*i.e*., $y = 1$) as abnormal. Specifically, for each video, MIL creates a tuple containing the prediction of f on the most anomalous snippet and the video's anomaly label, *i.e*., $(\max\{f(\mathbf{x}_i)\}_{i=1}^m, y)$. Then MIL aggregates the tuple for all videos to construct a labeled confident snippet set $\mathcal{C}$, and trains f by minimizing the binary cross-entropy (BCE) loss:

$$\text{BCE}(\mathcal{C}) = - \mathop{\mathbb{E}}_{(\hat{y},y)\sim\mathcal{C}} \left[y\log(\hat{y}) + (1-y)\log(1-\hat{y}) \right], \quad (1)$$

where $\hat{y} = \max\{f(\mathbf{x}_i)\}_{i=1}^m$. Note that some methods [30] use the mean squared error loss, which achieves the same outcome as Eq. (1). In this way, for a normal video with $y = 0$, by *minimizing* $\max\{f(\mathbf{x}_i)\}_{i=1}^m$, f must assign low abnormal probability for all the snippets. For an abnormal video with $y = 1$, by *maximizing* $\max\{f(\mathbf{x}_i)\}_{i=1}^m$, f is trained to output an even larger probability for the most confident abnormal snippet. However, the MIL training scheme suffers from biased sample selection: as f is trained to further increase $\max\{f(\mathbf{x}_i)\}_{i=1}^m$ in an abnormal video, the rest ambiguous snippets become even less likely to be selected by $\max$. Hence MIL essentially discards the ambiguous snippets and only trains on the confident ones, which leads to a biased detector (*e.g*., Figure 2).

In contrast, our proposed Unbiased MIL (UMIL) leverages both the confident and ambiguous snippets to train the anomaly classifier f. Specifically, in Step 1, we divide the snippets into a labeled confident snippet set $\mathcal{C}$ and an unlabeled ambiguous snippet set $\mathcal{A}$. In Step 2, we cluster $\mathcal{A}$ into 2 groups in an unsupervised fashion to distinguish the normal and abnormal snippets. Finally, in Step 3, f is supervised by both $\mathcal{C}$ and $\mathcal{A}$ to simultaneously predict the binary labels in $\mathcal{C}$ and separate the clusters in $\mathcal{A}$.

3.2. Step 1: Divide Snippets

Based on the predictions from f, we divide the snippets into the confident set $\mathcal{C}$ and the ambiguous one $\mathcal{A}$:

Constructing $\mathcal{C}$. During training, we track the history of the last 5 predictions from f for each snippet. Then, at the start of every epoch, we select N snippets $\mathbf{x}_1, \dots, \mathbf{x}_N$ with the least prediction variance, and the confident set $\mathcal{C}$ is given by $\{f(\mathbf{x}_i), y_i\}_{i=1}^N$. The rationale is that for the apparent normal or abnormal snippets (*e.g*., enclosed in red in Figure 2), their predictions tend to quickly converge to confident normal or abnormal with small predictive variance over time. This approach is empirically validated in Appendix, and we point out similar method shows promising results in [43].

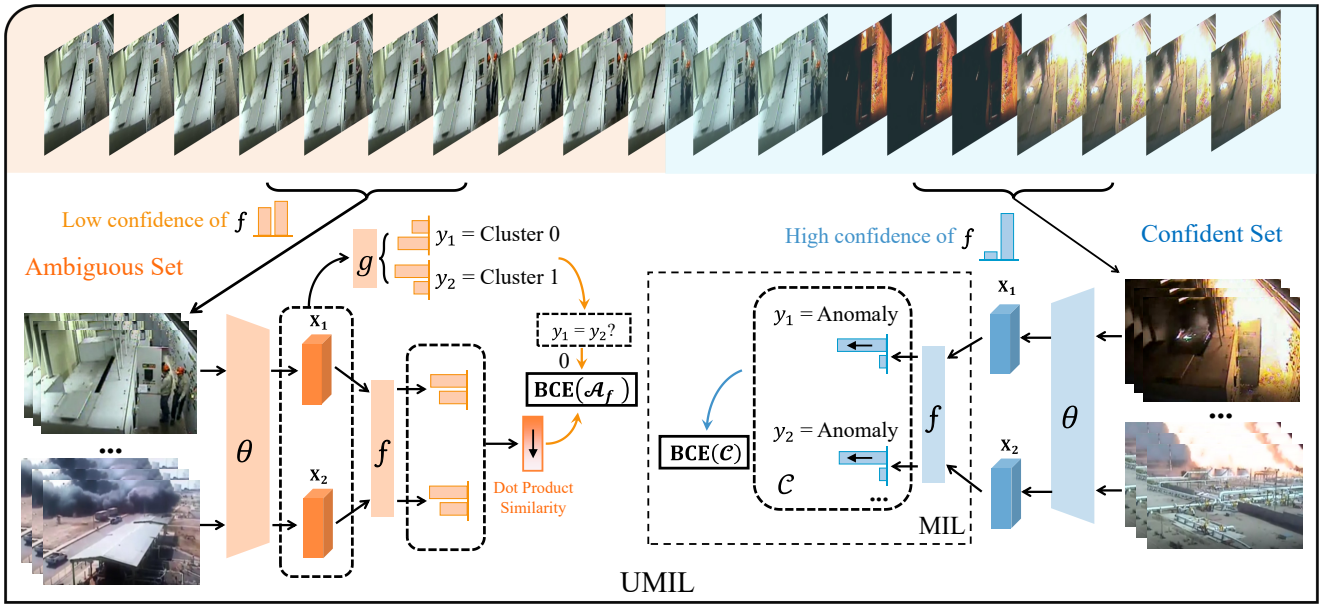


Figure 3. The proposed UMIL framework for WSVAD consists of a backbone model θ, an anomaly head f, and a cluster head g. We use the predictions by f to divide the snippets into a confident set $\mathcal{C}$ and an ambiguous set $\mathcal{A}$. In MIL, the model is only supervised by the confident snippets to further increase the confidence of anomaly prediction (the black arrows on the probability bar). In UMIL, f is additionally supervised by $\mathcal{A}$ to separate the two clusters identified by g for removing the context bias in $\mathcal{C}$. The black arrow on the similarity bar denotes that minimizing the BCE losses in $\mathcal{A}$ will decrease the dot-product similarity of the predictions on the pair, as they are from different clusters ($y_1 \neq y_2$). Best viewed in color.

Constructing $\mathcal{A}$. The rest of the M snippets have large prediction fluctuations, showing that f is still uncertain about them. They are collected as the ambiguous set $\mathcal{A} = \{\mathbf{x}_i\}_{i=1}^{M}$. Note that $\mathcal{A}$ is a set of features at this point, awaiting the next clustering step.

3.3. Step 2: Clustering Ambiguous Snippets

While the prediction from f is ambiguous on $\mathcal{A}$, the feature distribution can still reflect the intrinsic differences between normal and abnormal snippets. Hence we aim to cluster $\mathcal{A}$ into 2 groups to distinguish them. Specifically, we learn a cluster head g that takes the snippet feature $\mathbf{x} \in \mathcal{A}$ as input and outputs the softmax-normalized probabilities for being in each of the 2 clusters. The head g is trained in a pair-wise manner such that a pair of similar features have similar predictions from g (*i.e.*, from the same cluster), and vice versa for dissimilar. To accomplish this, we denote the pair-wise form of $\mathcal{A}$ based on cluster prediction from g as:

$$\mathcal{A}_g = \{g(\mathbf{x}_i)^\intercal g(\mathbf{x}_j), \mathbb{1}(\mathbf{x}_i \sim \mathbf{x}_j) \mid \mathbf{x}_i, \mathbf{x}_j \in \mathcal{A}\}, \quad (2)$$

where the dot-product is used to measure the prediction similarity, and $\mathbb{1}(\cdot)$ is an indicator function that returns 1 if the cosine similarity between $\mathbf{x}_i, \mathbf{x}_j$ is larger than a threshold τ (*i.e.*, $\mathbf{x}_i \sim \mathbf{x}_j$), and returns 0 otherwise. This allows us to train g by minimizing BCE($\mathcal{A}_g$).

With the optimized g, each feature $\mathbf{x}_i$ in $\mathcal{A}$ is assigned a cluster label $y_i = \operatorname{argmax} g(\mathbf{x}_i)$ as the cluster with the highest predicted probability. Next, we supervise f by $\mathcal{A}$ to separate the clusters and form our overall objective.

3.4. Step 3: Overall Objective

Note that unlike the sample-wise supervision provided by labels in $\mathcal{C}$, *i.e.*, whether a feature is normal or abnormal, the cluster labels in $\mathcal{A}$ only provide pair-wise supervision, *i.e.*, whether a feature pair is from the same cluster. Hence we supervise f with $\mathcal{A}$ using a pair-wise loss: f is trained to produce similar anomaly prediction on feature pairs with the same cluster label, and push away predictions for those in different clusters. This corresponds to minimizing BCE($\mathcal{A}_f$) with $\mathcal{A}_f$ based on the pair-wise prediction similarity of f:

$$\mathcal{A}_f = \{f(\mathbf{x}_i)^\intercal f(\mathbf{x}_j), \mathbb{1}(y_i = y_j) \mid \mathbf{x}_i, \mathbf{x}_j \in \mathcal{A}\}, \quad (3)$$

where $f(\mathbf{x}_i)^\intercal f(\mathbf{x}_j)$ denotes the dot-product similarity of the binary probabilities (*i.e.*, normal or abnormal)[1] with slight abuse of notation. The overall objective of UMIL is given by:

$$\min_{\theta,f,g} \overbrace{\text{BCE}(\mathcal{C})}^{\mathcal{C}\text{ supervision}} + \overbrace{\alpha\text{BCE}(\mathcal{A}_f)}^{\mathcal{A}\text{ supervision}} + \overbrace{\beta\text{BCE}(\mathcal{A}_g)}^{\text{Clustering in }\mathcal{A}}, \quad (4)$$

where α, β are trade-off parameters with ablations in Section 4.4. Hence in addition to the supervision from $\mathcal{C}$ as in

[1] While f only outputs the probability of being abnormal as p, the probability of being normal is easily computed as $1 - p$.

Algorithm 1 UMIL Training (1 epoch)

Input: $N+M$ video snippets, backbone parameterized by θ, classifier f and cluster head g, batch size b.
Output: θ, f, g trained for 1 epoch.
Compute $\{\mathbf{x}_i\}_{i=1}^{N+M}$ (features extracted by θ)
Update prediction history $\mathcal{H} \leftarrow \mathcal{H} \cup \{f(\mathbf{x}_i)\}_{i=1}^{N+M}$
Construct $\mathcal{C} = \{f(\mathbf{x}_i), y_i\}_{i=1}^{N}, \mathcal{A} = \{\mathbf{x}_i\}_{i=1}^{M}$ from $\mathcal{H}$
repeat
  Sample a batch $\{f(\mathbf{x}_i), y_i\}_{i=1}^{b}$ from $\mathcal{C}$
  Compute BCE($\mathcal{C}$) for the batch with Eq. (1)
  Sample a batch $\{\mathbf{x}_i\}_{i=1}^{b}$ from $\mathcal{A}$
  Assign $y_i \leftarrow \operatorname{argmax} g(\mathbf{x}_i)$ for $i \in \{1, \ldots, b\}$
  Construct $\mathcal{A}_g, \mathcal{A}_f$ with Eq. (2), (3) for the batch
  Compute BCE($\mathcal{A}_g$), BCE($\mathcal{A}_f$)
  Optimize θ, f, g with Eq. (4)
until end of epoch

MIL, f in UMIL is additionally supervised by $\mathcal{A}$ to separate its 2 clusters identified by g to remove the context bias in $\mathcal{C}$ (Figure 2). This unbiased objective allows us to train not only f, but also to fine-tune the backbone θ to get a tailored representation for VAD.

Training and Testing. Before training, the backbone θ is first pre-trained with MIL, and f, g are randomly initialized. Then the models are trained with our proposed UMIL by iterating Algorithm 1 until convergence. In testing, anomalies are labeled on the frame level. The model is evaluated with a non-overlapping sliding window of frames (*i.e.*, each window of frames is a snippet) to predict anomaly whenever the window intersects with any anomaly frame.

4. Experiments

4.1. Datasets and Evaluation Metrics

We conducted extensive experiments and ablations on two standard WSVAD evaluation datasets [22, 30]. As per standard in WSVAD, the training videos only have video-level labels, and the test videos have frame-level labels. Other details are given below:

UCF-Crime [30] is a large-scale dataset that contains 1,900 untrimmed real-world outdoor and indoor surveillance videos. The total length of the videos is 128 hours, which contains 13 classes of anomalous events. We follow the standard split: the training set contains 1,610 videos, and the test set contains 290 videos.

TAD dataset [22] contains real-world videos of traffic scenes with a total length of 25 hours and 1,075 average frames per video. The videos contain more than 7 categories of anomalies that are common on roads. The dataset is partitioned as a training set with 400 videos, and a test set with 100 videos.

Evaluation Metrics. Following previous works [30, 43], we used the Area Under the Curve (AUC) of the frame-level ROC (Receiver Operating Characteristic) as the main evaluation metric for TAD and UCF-Crime. Intuitively, a larger AUC means a larger margin between the normal and abnormal snippet predictions, hence indicating a better anomaly classifier. Inspired by Lv *et al*. [22], besides evaluating AUC on the overall test set with normal and abnormal videos, denoted as AUC_O, we also computed the AUC on abnormal ones alone, denoted as AUC_A. The rationale is to remove normal videos where all snippets are normal (label 0), and keep only the abnormal ones with both kinds of snippets (label 0,1), which truly challenges a classifier's capability of localizing anomalies.

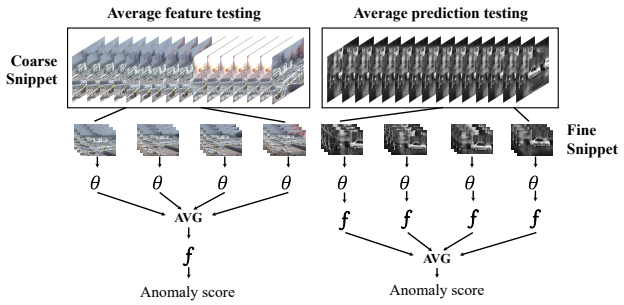


Figure 4. Average feature versus average prediction testing. θ, f: the feature backbone and anomaly classifier, respectively.

4.2. Implementation Details

Video Sequence Partition. Existing works partition each video into multiple coarse snippets, and use the *average feature* in each one as the input to their classifiers (Figure 4 left). However, we find that the subtle anomaly feature is often diluted by averaging features over the coarse snippets (see Appendix). This has less impact on the traditional MIL compared to our UMIL, as MIL only leverages the confident snippets with apparent anomalies. Therefore, in UMIL training, we used fine-grained snippets with one-second lengths. In testing, to generate the prediction for a coarse snippet, we used the *average predictions* over the fine snippets inside the coarse one (Figure 4 right).

Baseline. We built a baseline to validate that the improvements of UMIL are indeed from the unbiased training scheme (Section 4.4), rather than the above testing scheme based on average predictions. Specifically, the baseline has exactly the same model design as UMIL, and we trained it with the MIL objective in Eq. (1) on fine snippets and tested it by averaging predictions. Hence the only difference between the baseline and UMIL is the training objective.

Model Training. We implemented the backbone θ with the X-CLIP-B/32 model [26] fine-tuned on Kinectics-400 [4] to improve its capabilities in action recognition. We used the fully connected layer to implement the anomaly classifier f and the cluster head g. We trained our model with the AdamW optimizer [17] using an initial learning rate of 8e-

Category	Method	AUC_O (%)	AUC_A (%)
UVAD	SVM Baseline	50.00	50.00
	Conv-AE [6]	50.60	-
	Sohrab et al. [29]	58.50	-
	Lu et al. [18]	65.51	-
	BODS [33]	68.26	-
	GODS [33]	70.46	-
WSVAD	Sultani et al. [30]	75.41	54.25
	Zhang et al. [41]	78.66	-
	Motion-Aware [44]	79.10	62.18
	GCN-Anomaly [43]	82.12	59.02
	Wu et al. [34]	82.44	-
	RTFM [31]	84.30	-
	WSAL [22]	85.38	67.38
	Baseline	80.67	60.57
	UMIL	**86.75**	**68.68**

Table 1. Frame-level AUC performance on UCF-Crime. Best results in bold. AUC_O and AUC_A denote that the AUC computed on the overall test set and only abnormal test videos, respectively. "UVAD" and "WSVAD" under category denote Unsupervised VAD and Weakly-Supervised VAD, respectively.

6, weight decay of 0.001, and batch size of 8. We utilized the cosine annealing scheduler and warmed up the learning rate for 5 epochs. Our UMIL model was pre-trained with MIL for 30 epochs, followed by 10 epochs of UMIL training. We conducted all experiments on 4 TITAN RTX GPUs. We implement the max value scores as well as max margin scores [22] in $\mathcal{C}$ supervision of Eq 4. We also incorporated entropy minimization as a standard auxiliary objective [14, 16], and added the self-training loss, which leverages the learned unbiased anomaly classifier f to generate accurate pseudo-labels on samples in the ambiguous set $\mathcal{A}$ for additional supervision. Details in Appendix.

4.3. Main Results

UCF-Crime and TAD. In Table 1, we compared our UMIL with other state-of-the-art (SOTA) methods in both Unsupervised VAD (UVAD) and WSVAD. On UCF-Crime [30], UMIL achieves the best AUC_O and AUC_A among all the methods, with an improvement of +1.37% and +1.3%, respectively. UMIL also significantly outperforms all methods in TAD [22] by +3.3% on AUC_O and +4.2% on AUC_A.
Overall Observations. 1) Notice that our baseline performs similarly (*e.g.*, AUC_O on TAD) or even worse (*e.g.*, 60.57% versus 67.38% on UCF-Crime AUC_O) compared to existing MIL-based methods. This validates that the improvements from UMIL are not from the test scheme of averaging predictions. 2) In particular, our improvement in AUC_A indicates that the superior performance of UMIL on AUC_O is not merely from easy normal videos, but also from improved capabilities to identify anomalous snippets in abnormal videos. 3) Moreover, on both datasets, WSVAD significantly improves over UVAD on AUC_O, which empirically validates that detecting open-set anomalies in UVAD is ill-posed (Section 1). However, the improvements

Category	Method	AUC_O (%)	AUC_A (%)
UVAD	SVM Baseline	50.00	50.00
	Luo *et al.* [19]	57.89	55.84
	Liu *et al.* [15]	69.13	55.38
WSVAD	Sultani *et al.* [30]	81.42	55.97
	Motion-Aware [44]	83.08	56.89
	GIG [21]	85.64	58.65
	WSAL [22]	89.64	61.66
	Baseline	89.10	56.47
	Ours	**92.93**	**65.82**

Table 2. Frame-level AUC performance on TAD benchmark.

Baseline	ST	RTFM*	UMIL	AUC_O (%) - UCF	AUC_O (%) - TAD
✓				80.67	89.10
✓	✓			82.01	90.80
✓	✓	✓		83.45	91.28
✓			✓	83.66	91.74
✓	✓		✓	**86.75**	**92.93**

Table 3. Ablation studies of the components in UMIL on UCF-Crime and TAD. *: we re-implemented RTFM with our backbone and average-prediction-based testing scheme for fair comparison.

Threshold(%)	10	**30**	50	70	90
AUC_O (%) - UCF	86.8	**86.8**	85.9	84.3	83.1
AUC_O (%) - TAD	92.7	**93.0**	92.8	91.5	91.1

Table 4. Ablation on the threshold to divide the confident/ambiguous snippet set on UCF-Crime and TAD.

in AUC_A are much smaller (*e.g.*, 54.25% over 50.00% on UCF-Crime). This shows that the existing WSVAD methods are still biased toward the apparent normal/abnormal, causing many false positives and negatives on ambiguous snippets from the abnormal videos. 4) Our UMIL significantly improves the AUC_A over MIL (*e.g.*, +4.2% on TAD), which demonstrates the effectiveness of using ambiguous snippets in UMIL to learn an unbiased invariant classifier. 5) Interestingly, TAD tends to have larger AUC_O but lower AUC_A, *e.g.*, from UCF-Crime to TAD, UMIL's AUC_O is 6.2% higher, but AUC_A is 2.8% lower. The improved overall performance suggests that TAD has stronger context bias in the confident set, *i.e.*, more apparent normal/abnormal snippets, and the dropped AUC_A indicates that it contains more subtle anomalies in the ambiguous snippets that are hard to detect and localize. This also explains why our UMIL improves AUC_A more on TAD than UCF-Crime by incorporating ambiguous snippets to remove the context bias from the confident set.

4.4. Ablations

Components. Our approach has 2 main components: 1) the self-training objective; 2) the UMIL objective in Eq. (4). We validate the effectiveness of each component in Table 3 with AUC_O. All ablations in the table are on the equal ground—using average prediction instead of average feature for anomaly detection (*i.e.*, Baseline). By comparing

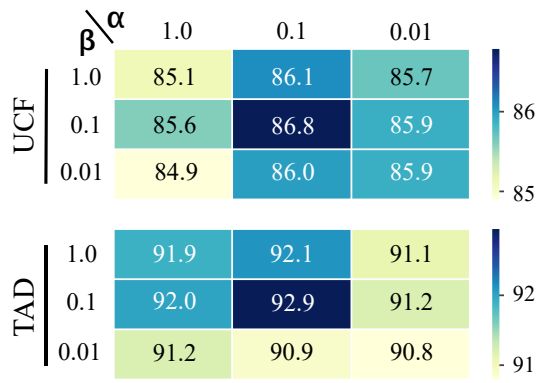


Figure 5. Ablations on the trade-off parameters.

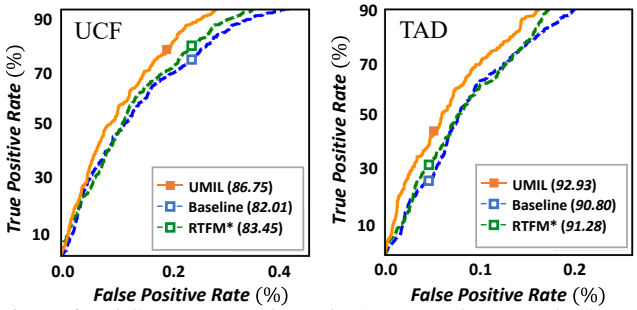


Figure 6. ROC curves on UCF and TAD. Note that we only show part of the curves for visual clarity, as the other part of the methods have a large overlap when the true positive rate approaches 100%.

the first two lines, we observe that self-training can improve AUC_O from 80.67% to 82.01% on UCF-crime and 89.10% to 90.80% on TAD. To independently evaluate the effectiveness of UMIL objective, we re-implement the SOTA RTFM [31] using our backbone and add the self-training objective, namely RTFM*. The result is listed in line 3. Our UMIL in line 4 still significantly outperforms RTFM* (+3.3% on UCF-crime and +1.7% on TAD), hence validating the effectiveness of our unbiased learning objectives.

Confident Threshold. We then conducted experiments to analyze the effects of the variance threshold for dividing confident and ambiguous snippets as in Section 3.2. Specifically, we selected k (%) training snippets with the minimum variance on their prediction history with varying k as in Table 4. Overall the threshold is easy to determine, *i.e.*, 10-50% is a reasonable range with 30% being the best.

Trade-off Parameters. Recall that we use α and β in Eq. (4) as the trade-off for the supervision from the ambiguous set $\mathcal{A}$ and clustering, respectively. We empirically find in Figure 5 that $\alpha, \beta = 0.1$ are suitable across the two datasets, hence we used this setting in the experiments by default. In general, the choice of α depends on the strength of the context bias in the confident set, *e.g.*, TAD has strong bias as analyzed in Section 4.3, which cannot be overcome with a small α (*e.g.*, α=0.01 has low performance).

Class-wise AUC. On UCF-Crime dataset, the class of anomaly in each test video is given. This allows us to plot

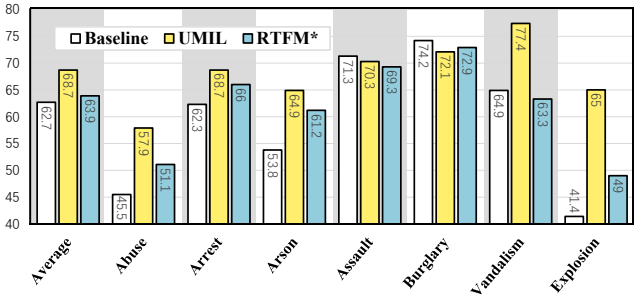


Figure 7. Class-wise AUC_A of three methods on UCF-Crime.

the class-wise AUC_A to examine models' capabilities to detect subtle abnormal events. In Figure 7, we compared UMIL with baseline and RTFM*, where "Average" shows the overall AUC_A and the rest shows the class-wise one. We have the following observation: 1) Both of the two MIL-based methods perform well on human-centric anomaly classes with drastic motions, *e.g.*, "Assault" and "Burglary". These classes correspond to apparent anomalies as the backbone expresses the human action feature well (fine-tuned on the action recognition Kinetics400 dataset [4]). 2) However, we notice that they easily fail to distinguish anomalies with subtle motions, *e.g.*, "Arson" and "Vandalism", as well as non-human-centric anomalies, *e.g.*, "Explosion". These classes correspond to ambiguous anomalies discarded by the biased training in MIL. 3) Our UMIL performs similarly on the above apparent anomaly classes and much better on the other subtle anomalies, which largely contributes to the superior anomaly detection and localization performance. Overall, observation 1 and 2 empirically verifies the biased prediction situation of MIL in Figure 1 and Figure 2. In contrast, our UMIL convincingly improves the performance on ambiguous anomalies with almost no sacrifice on the confident ones, which validates the effectiveness of our approach, *i.e.*, identifying the invariance between the two types of anomalies to remove the bias in MIL.

ROC Curve. In Figure 6, we draw the ROC Curve on the overall test set for our baseline, the re-implemented RTFM* and UMIL, which shows the true and false positive rate for detecting anomaly on a sweeping threshold over the predictions. VAD is evaluated using the area under this curve to demonstrate the overall separation of normal and abnormal snippet predictions. However, when applying a detector for real-world usage, we need to choose a specific threshold (*e.g.*, with a maximum tolerable false positive rate). We observe from Figure 6 that our UMIL outperforms the two MIL baselines in every inch, which further shows the strength of our proposed unbiased training.

Qualitative Analysis. In Figure 8, we show the continuous predictions of anomaly probabilities from our baseline, RTFM*, and our UMIL on 4 test videos on UCF-crime. We summarize the observations: 1) For the MIL baseline (2nd column), we observe that it assigns a larger probability on the pre-explosion snippets from B1 and B2 (top two videos),

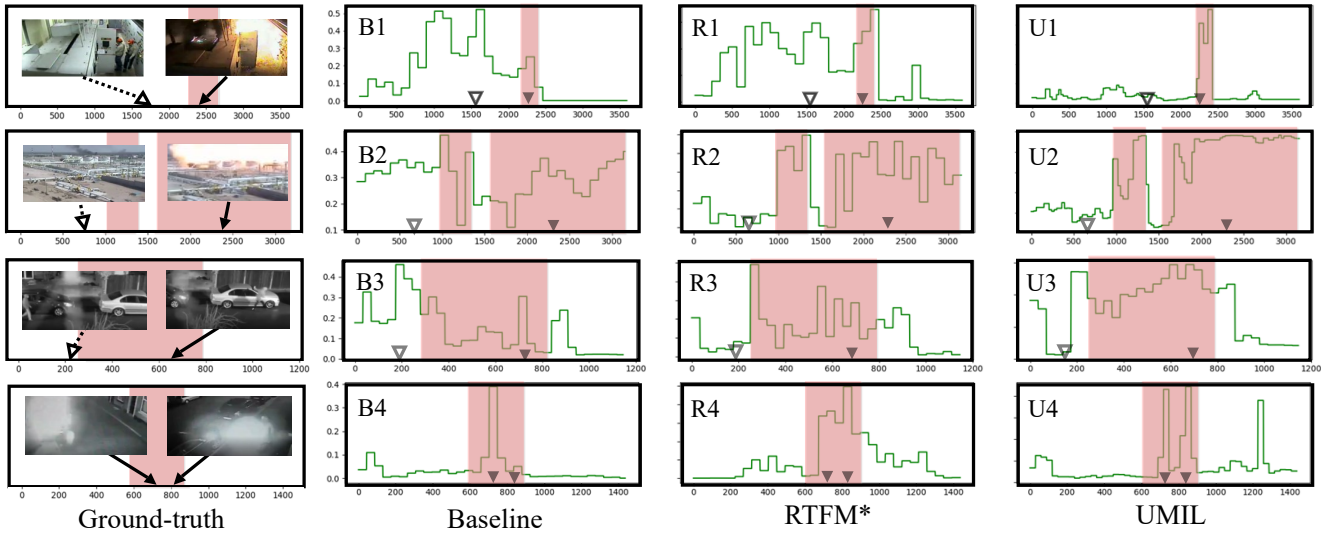


Figure 8. Visualization cases of ground-truth and anomaly score curves of various approaches. The white and black triangles denote the location of the normal and abnormal frame displayed on the left, respectively. The green curves represent the anomaly predictions of various methods. The pink background corresponds to the ground-truth abnormal regions.

e.g., workers performing maintenance and snippets with smoke, yet the actual explosion may have a lower prediction (*e.g.*, comparing the height of the green lines on the white and black triangle locations). Similarly, on B3, the running person (white triangle) triggers a larger anomaly prediction than the actual vandalism (black triangle). This further illustrates the biased prediction problem in MIL. 2) RTFM (3rd column) uses feature magnitudes to assist anomaly detection by assuming anomalous snippets have larger magnitudes, which indeed improves over the baseline sometimes, *e.g.*, R2 is no longer biased to smoke. However, its assumption has no guarantee to hold and hence the failure on subtle anomalies persists, *e.g.*, false alarm in R1 white triangle location and low prediction in R3 black triangle location. 3) In contrast, our UMIL localizes the anomalies accurately in U1-U3, *e.g.*, having consistently high scores in the pink areas, which holds its ground on the name "unbiased". 4) In the 4th video, however, RTFM's prediction in the pink area is more consistent than ours. By inspecting the frames on the left, we realize that the two peaks in the pink area of U4 correspond to the burning fire and the running suspect caught on fire. Hence UMIL's prediction is reasonable and sufficient for triggering the alarm on the first peak.

Computational Efficiency. Lastly, we investigated the speed of the proposed model. For inference, our method processes a 5-frame clip in 0.003 seconds on a Nvidia 2080Ti GPU. Notably, this is almost $80\times$ faster than the SOTA RTFM [31], which spends 0.76 seconds to process a 16-frame clip on Nvidia 2080Ti. Thanks to our unbiased training scheme, we can fine-tune the backbone to learn a WSVAD-tailored representation, which achieves even better performance than existing SOTA. This also shows the promising future of UMIL in real-time applications.

5. Conclusion

In this work, we presented an Unbiased Multiple Instance Learning (UMIL) scheme that learns an unbiased anomaly classifier and a tailored representation for Weakly Supervised Video Anomaly Detection (WSVAD). Specifically, the existing MIL training scheme suffers from the context bias by only training on the confident set containing apparent normal/abnormal video snippets. We replace it with an unbiased one—seeking the invariant predictor that simultaneously distinguishes the normal/abnormal snippets in the confident set, and separates the two unsupervised clusters in the rest ambiguous snippets. Hence the context bias that fails among the ambiguous ones is removed. Our approach is empirically validated by the state-of-the-art performance and extensive ablations on standard WSVAD benchmarks. In future, we will seek additional prior beyond unsupervised clustering to discover the intrinsic differences between the ambiguous normal and abnormal snippets and adopt principled representation learning paradigm (*e.g.*, disentanglement) to highlight the anomaly features.

6. Acknowledgments

The author gratefully acknowledges the support of the A*STAR under its AME YIRG Grant (Project No.A20E6c0101), the Lee Kong Chian (LKC) Fellowship fund awarded by Singapore Management University, AI Singapore AISG2-RP-2021-022, the Postgraduate Research & Practice Innovation Program of Jiangsu Province, the National Natural Science Foundation of China (Grants No.62072244), the Natural Science Foundation of Shandong Province (Grant No.ZR2020LZH008). This work was also supported in part by State Key Laboratory of High-end Server & Storage Technology.

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 *CVPR*, 2022. 1

[2] Amit Adam, Ehud Rivlin, Ilan Shimshoni, and Daviv Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *TPAMI*, 2008. 3

[3] Borislav Antić and Björn Ommer. Video parsing for abnormality detection. In *ICCV*, 2011. 1, 3

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 3, 5, 7

[5] Allison Del Giorno, J Andrew Bagnell, and Martial Hebert. A discriminative framework for anomaly detection in large videos. In *ECCV*, 2016. 2

[6] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *CVPR*, 2016. 3, 6

[7] Chengkun He, Jie Shao, and Jiayu Sun. An anomaly-introduced learning method for abnormal event detection. *Multimedia Tools and Applications*, 2018. 3

[8] Zijie Huang and Yulei Wu. A survey on explainable anomaly detection for industrial internet of things. In *DSC*, 2022. 1

[9] Shunsuke Kamijo, Yasuyuki Matsushita, Katsushi Ikeuchi, and Masao Sakauchi. Traffic monitoring and accident detection at intersections. *IEEE transactions on Intelligent Transportation Systems*, 2000. 1

[10] Moshe Koppel, Jonathan Schler, and Elisheva Bonchek-Dokow. Measuring differentiability: Unmasking pseudonymous authors. *Journal of Machine Learning Research*, 8(6), 2007. 3

[11] Louis Kratz and Ko Nishino. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In *CVPR*, 2009. 1

[12] Shuo Li, Fang Liu, and Licheng Jiao. Self-training multi-sequence learning with transformer for weakly supervised video anomaly detection. *AAAI*, 2022. 1, 3

[13] Weixin Li, Vijay Mahadevan, and Nuno Vasconcelos. Anomaly detection and localization in crowded scenes. *TPAMI*, 2013. 1

[14] Hong Liu, Jianmin Wang, and Mingsheng Long. Cycle self-training for domain adaptation. In *NeurIPS*, 2021. 6

[15] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection–a new baseline. In *CVPR*, 2018. 3, 6

[16] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, 2018. 6

[17] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *ICLR*, 2019. 5

[18] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *ICCV*, 2013. 3, 6

[19] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. In *ICCV*, 2017. 6

[20] Hui Lv, Chen Chen, Zhen Cui, Chunyan Xu, Yong Li, and Jian Yang. Learning normal dynamics in videos with meta prototype network. In *CVPR*, 2021. 3

[21] Hui Lv, Chunyan Xu, and Zhen Cui. Global information guided video anomaly detection. In *ACM MM*, 2020. 6

[22] Hui Lv, Chuanwei Zhou, Zhen Cui, Chunyan Xu, Yong Li, and Jian Yang. Localizing anomalies from weakly-labeled videos. *TIP*, 2021. 1, 2, 3, 5, 6

[23] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. Anomaly detection in crowded scenes. In *CVPR*, 2010. 3

[24] Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *CVPR*, 2009. 3

[25] Sadegh Mohammadi, Alessandro Perina, Hamed Kiani, and Vittorio Murino. Angry crowds: Detecting violent events in videos. In *ECCV*, 2016. 1

[26] Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition. In *ECCV*, 2022. 5

[27] Mahdyar Ravanbakhsh, Moin Nabi, Hossein Mousavi, Enver Sangineto, and Nicu Sebe. Plug-and-play cnn for crowd motion analysis: An application in abnormal event detection. In *WACV*, 2018. 3

[28] Mohammad Sabokrou, Mahmood Fathy, Mojtaba Hoseini, and Reinhard Klette. Real-time anomaly detection and localization in crowded scenes. In *CVPR workshops*, 2015. 3

[29] Fahad Sohrab, Jenni Raitoharju, Moncef Gabbouj, and Alexandros Iosifidis. Subspace support vector data description. In *ICPR*, 2018. 6

[30] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *CVPR*, 2018. 1, 2, 3, 5, 6

[31] Yu Tian, Guansong Pang, Yuanhong Chen, Rajvinder Singh, Johan W Verjans, and Gustavo Carneiro. Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. In *ICCV*, 2021. 3, 6, 7, 8

[32] Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, and Marius Popescu. Unmasking the abnormal events in video. In *ICCV*, pages 2895–2903, 2017. 2

[33] Jue Wang and Anoop Cherian. Gods: Generalized one-class discriminative subspaces for anomaly detection. In *ICCV*, 2019. 6

[34] Peng Wu, Jing Liu, Yujia Shi, Yujia Sun, Fangtao Shao, Zhaoyang Wu, and Zhiwei Yang. Not only look, but also listen: Learning multimodal violence detection under weak supervision. In *ECCV*, 2020. 6

[35] Shandong Wu, Brian E Moore, and Mubarak Shah. Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes. In *CVPR*, 2010. 1

[36] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *NeuIPs*, 2015. 3

[37] Dan Xu, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe. Learning deep representations of appearance and motion for anomalous event detection. *BMVC*, 2015. 3

[38] Zhongqi Yue, Qianru Sun, Xian-Sheng Hua, and Hanwang Zhang. Transporting causal mechanisms for unsupervised domain adaptation. In *ICCV*, 2021. 3

[39] Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Interventional few-shot learning. *NeurIPS*, 2020. 3

[40] M Zaigham Zaheer, Arif Mahmood, M Haris Khan, Mattia Segu, Fisher Yu, and Seung-Ik Lee. Generative cooperative learning for unsupervised video anomaly detection. In *CVPR*, 2022. 3

[41] Jiangong Zhang, Laiyun Qing, and Jun Miao. Temporal convolutional network with complementary inner bag loss for weakly supervised anomaly detection. In *ICIP*, 2019. 6

[42] Bin Zhao, Li Fei-Fei, and Eric P Xing. Online detection of unusual events in videos via dynamic sparse coding. In *CVPR*, 2011. 1

[43] Jia-Xing Zhong, Nannan Li, Weijie Kong, Shan Liu, Thomas H Li, and Ge Li. Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection. In *CVPR*, 2019. 1, 3, 5, 6

[44] Yi Zhu and Shawn Newsam. Motion-aware feature for improved video anomaly detection. *BMVC*, 2019. 3, 6