A Coarse-to-Fine Pseudo-Labeling (C2FPL) Framework for Unsupervised Video Anomaly Detection

Anas Al-lahham   Nurbek Tastan   Zaigham Zaheer   Karthik Nandakumar

Mohamed bin Zayed University of Artificial Intelligence (MBZUAI)

Abu Dhabi, UAE

{anas.al-lahham, nurbek.tastan, zaigham.zaheer, karthik.nandakumar}@mbzuai.ac.ae

Abstract

Detection of anomalous events in videos is an important problem in applications such as surveillance. Video anomaly detection (VAD) is well-studied in the one-class classification (OCC) and weakly supervised (WS) settings. However, fully unsupervised (US) video anomaly detection methods, which learn a complete system without any annotation or human supervision, have not been explored in depth. This is because the lack of any ground truth annotations significantly increases the magnitude of the VAD challenge. To address this challenge, we propose a simple-but-effective two-stage pseudo-label generation framework that produces segment-level (normal/anomaly) pseudo-labels, which can be further used to train a segment-level anomaly detector in a supervised manner. The proposed coarse-to-fine pseudo-label (C2FPL) generator employs carefully-designed hierarchical divisive clustering and statistical hypothesis testing to identify anomalous video segments from a set of completely unlabeled videos. The trained anomaly detector can be directly applied on segments of an unseen test video to obtain segment-level, and subsequently, frame-level anomaly predictions. Extensive studies on two large-scale public-domain datasets, UCF-Crime and XD-Violence, demonstrate that the proposed unsupervised approach achieves superior performance compared to all existing OCC and US methods, while yielding comparable performance to the state-of-the-art WS methods. Code is available at: https://github.com/AnasEmad11/C2FPL

1. Introduction

Applications such as video surveillance continuously generate large amounts of video data. While a vast majority of these videos only contain normal behavior, it is essential to detect anomalous events (e.g., shooting, road accidents, fighting, etc.) that deviate from normal behavior and may occur occasionally in such videos. Hence, video anomaly
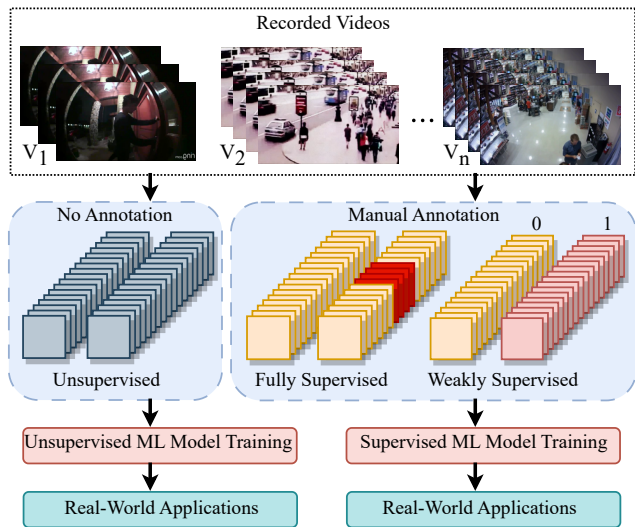

Figure 1. Supervised (right) vs. unsupervised (left) video anomaly detection pipeline. In a supervised setting, some sort of manual annotation of the recorded videos is required for training an anomaly detection model. We aim to eliminate this annotation step by proposing a fully unsupervised approach.

detection (VAD) is a critical problem, especially in surveillance applications [9, 13, 14, 19].

Conventional VAD methods rely heavily on manually annotated anomaly examples (Figure 1(right)) [2]. However, given the rare occurrence and short temporal nature of anomalies in real-world scenarios, obtaining accurate fine-grained annotations is a laborious task. Recently, several VAD methods have been proposed to leverage video-level labels and perform weakly supervised (WS) training [5, 15, 19, 25, 35, 39] to reduce the annotation costs. However, since surveillance datasets are usually a large-scale collection of videos, it is still cumbersome to obtain any kind of labels. For example, to obtain even a video-level binary label, an annotator may still have to watch the

whole video, which can take a considerable amount of time. For example, a well-known WS-VAD dataset called XD-Violence [31] contains videos spanning 217 hours. An alternative paradigm for VAD is one-class classification (OCC), which assumes that only normal videos are available for training [16, 19, 25, 30, 35]. However, the OCC setting does not completely alleviate the annotation problem because an annotator still has to watch all the training videos to ensure that no anomaly is present within them.

A label-free fully unsupervised approach is a more practical and useful setting, especially in real-world scenarios where recording video data is easier than annotating it [36]. An unsupervised video anomaly detection (US-VAD) method can address the aforementioned disadvantages of supervised methods by completely eradicating the need for manual annotations (Figure 1). However, US-VAD methods are yet to gain much traction within the computer vision community. Recently, Zaheer *et al*. [36] introduced an US-VAD approach in which the model is trained on unlabeled normal and anomalous videos. Their idea is to utilize several properties of the training data to obtain pseudo-labels via cooperation between a generator and a classifier. While this method is elegant, its performance is significantly lower than the state-of-the-art WS and OCC methods [23, 36].

In this work, we attempt to bridge this gap between unsupervised and supervised methods by taking unlabelled set of training videos as input and producing segment-level pseudo-labels without relying on any human supervision. Towards this end, we make the following key contributions:

- We propose a two-stage coarse-to-fine pseudo-label (C2FPL) generator that utilizes hierarchical divisive (top-down) clustering and statistical hypothesis testing to obtain segment-level (fine-grained) pseudo-labels.

- Based on the C2FPL framework, we propose an US-VAD system that is trainable without any annotations. To the best of our knowledge, this is among the first few works to explore the US-VAD setting in detail.

- We evaluate the proposed approach on two large-scale VAD datasets, UCF-Crime [19] and XD-Violence [31], and achieve state-of-the-art performance in the unsupervised category, while also outperforming all existing OCC and several WS-VAD methods.

2. Related Work

Early VAD methods mostly relied on supervised learning, where anomalous frames in a video are explicitly labeled in the training data [6, 27]. Since supervised approaches require large amounts of annotated data and annotation of anomalies is a laborious task, WS, OCC, and US VAD methods are gaining more attention.

2.1. One-Class Classification for VAD

To avoid the capturing of anomalous examples, researchers have widely explored one-class classification (OCC) methods [7, 12, 29, 32]. In OCC-VAD, only normal videos are used to train an outlier detector. At the time of inference, data instances that do not conform to the learned normal representations are predicted as anomalous. Since OCC methods are known to fail if normal data contains some anomaly examples [36], they require careful verification of all the videos in the dataset, which does not reduce the annotation load. Furthermore, video data is often too diverse to be modeled successfully and new normal scenes differing from the learned representations may be classified as anomalous. Therefore, OCC approach has limited applicability in the context of VAD.

2.2. Weakly Supervised VAD

Taking advantage of weakly labeled (i.e., video-level labels) anomalous samples has led to significant improvements over OCC training [19, 25]. Multiple Instance Learning (MIL) is one of the most commonly used methods for WS-VAD [16, 19, 25, 30], where segments of a video are grouped into a bag and bag-level labels are assigned. Sultani *et al*. [19] first introduced the MIL framework with a ranking loss function, which is computed between the top-scoring segments of normal and anomaly bags.

One of the key challenges in WS-VAD is that the positive (anomaly) bags are noisy. Since anomalies are localized temporally, most of the segments in an anomaly bag are also normal. Therefore, Zhong *et al*. [39] reformulated the problem as binary classification in the presence of noisy labels and used a graph convolution network (GCN) to remove label noise. The training of GCN was computationally expensive due to the presence of an action classifier. Furthermore, MIL-based methods require complete video inputs at each training iteration. Consequently, the correlation of the input data significantly affects the training of an anomaly detection network. To minimize this correlation, CLAWS Net [35] proposed a random batch selection approach in which temporally consistent batches are arbitrarily selected for training a binary classifier.

2.3. Unsupervised VAD

Unsupervised video anomaly detection (US-VAD) methods are learned using unlabeled training data. This problem is extremely challenging due to the lack of ground truth supervision and the rarity of anomalies. However, it is highly rewarding because it can completely eradicate the costs associated with obtaining manual annotations and allow such systems to be deployed without human intervention. Due to the difficulty of the problem, it has received little attention in the literature. Generative Cooperative Learning [36]
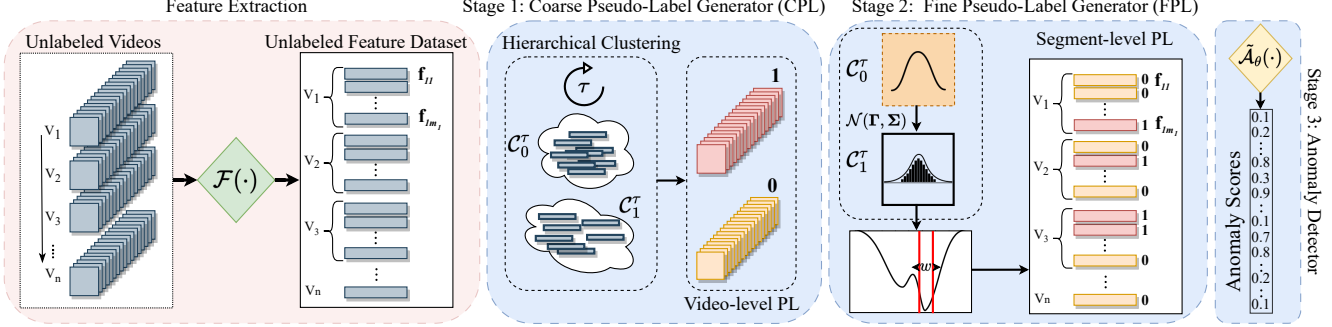
Figure 2. Overall training pipeline of the proposed C2FPL framework for unsupervised video anomaly detection (US-VAD). All training videos are first partitioned into segments and each segment is represented by a feature vector obtained using a pre-trained feature extractor. Then, our two-stage coarse-to-fine pseudo-label (C2FPL) generator produces segment-level pseudo-labels, which are used to train a segment-level anomaly detector. Pseudo-label generation consists of two stages: hierarchical divisive clustering that generates coarse (video-level) pseudo-labels (CPL) and statistical hypothesis testing that creates fine-grained (segment-level) pseudo-labels (FPL).

is a recent work that presents an US-VAD system to detect anomalies in a given video by first training a generative model to reconstruct normal video frames and then using the discrepancy between the reconstructed frames and the actual frames as a measure of anomaly. It involves training two models simultaneously: one to reconstruct the normal frames and the other to generate classification scores.

3. Proposed Methodology

Problem Definition: Let $\mathcal{D} = \{V_1, V_2, \cdots, V_n\}$ be a training dataset containing n videos without any labels. The goal of US-VAD is to use $\mathcal{D}$ and learn an anomaly detector $\mathcal{A}(\cdot)$ that classifies each frame in a given test video V_* as either *normal* (0) or *anomalous* (1).

Notations: We split each video V_i into a sequence of m_i non-overlapping segments S_{ij}, where each segment is in turn composed of r frames. Note that $i \in [1, n]$ refers to the video index, and $j \in [1, m_i]$ is the segment index within a video. While many WS-VAD methods [16, 19, 25, 30] compress each video into a fixed number of segments (i.e., $m_i = m, \forall i \in [1, n]$) along the temporal axis, we avoid any compression and make use of all available non-overlapping segments. For each segment S_{ij}, a feature vector $\mathbf{f}_{ij} \in \mathbb{R}^d$ is obtained using a pre-trained feature extractor $\mathcal{F}(\cdot)$.

High-level Overview of the Proposed Solution: Our coarse-to-fine pseudo-labeling (C2FPL) framework for US-VAD consists of three main stages during training (see Figure 2). In the first coarse pseudo-labeling (CPL) stage, we generate a video-level pseudo-label $\hat{y}_i \in \{0, 1\}$, $i \in [1, n]$ for each video in the training set using a hierarchical divisive clustering approach. In the second fine pseudo-labeling (FPL) stage, we generate segment-level pseudo-labels $\tilde{y}_{ij} \in \{0, 1\}$, $i \in [1, n]$, $j \in [1, m_i]$ for all the segments in the training set through statistical hypothesis testing. In the third anomaly detection (AD) stage, we train

a segment-level anomaly detector $\tilde{\mathcal{A}}_\theta(\cdot) : \mathbb{R}^d \to [0, 1]$ that assigns an anomaly score between 0 and 1 (higher values indicate higher confidence of being an anomaly) to the given video segment based on its feature representation $\mathbf{f}_{ij}$.

3.1. Coarse (Video-Level) Pseudo-Label Generator

Since the training dataset does not contain any labels, we first generate pseudo-labels for the videos in the training set by recursively clustering them into two groups: normal and anomalous (see Alg. 1). The idea of using iterative clustering to generate pseudo-labels has been considered earlier in other application domains [1, 4, 38]. However, direct application of these methods to the US-VAD problem fails to provide satisfactory solutions due to two reasons. Firstly, directly clustering multivariate features $\mathbf{f}_{ij}$ leads to a curse of dimensionality (features are high-dimensional but the sample size is small). Secondly, the clusters in our context are not permutation-invariant (normal and anomalous cluster labels cannot be interchanged). To overcome these problems, we propose a method that relies on a low-dimensional feature summary and divisive hierarchical clustering.

Previous works in WS-VAD have shown that normal video segments have lower temporal feature magnitude compared to anomalous segments [26]. Furthermore, we also observed that the variations in feature magnitude across different segments are lower for normal videos. Based on this intuition, we represent each video V_i using a statistical summary $\mathbf{x}_i = [\mu_i, \sigma_i]$ of its features as follows:

$$\mu_i = \frac{1}{m_i} \sum_{j=1}^{m_i} \|\mathbf{f}_{ij}\|_2, \tag{1}$$

$$\sigma_i = \sqrt{\frac{1}{(m_i - 1)} \sum_{j=1}^{m_i} (\|\mathbf{f}_{ij}\|_2 - \mu_i)^2}, \tag{2}$$

where $\|\cdot\|_2$ represents the ℓ_2 norm of a vector. Thus, each video V_i is represented using a 2D vector $\mathbf{x}_i$, corresponding to the mean and standard deviation of the feature magnitude of its segments. This ensures a uniform representation of all videos despite their varying temporal length.

Videos in the training set are iteratively divided into two clusters ($\mathcal{C}_0^t$ and $\mathcal{C}_1^t$) based on the above representation $\mathbf{x}_i$. Here, t denotes the step index and $\mathcal{C}_0$ and $\mathcal{C}_1$ represent the normal and anomaly clusters, respectively. Since no data labels are available, assigning normal and anomaly labels to the clusters is not trivial. Intuitively, easy anomalies (considered as easy outliers) may be separated into a smaller cluster. On the other hand, the larger cluster is likely to contain more normal videos as well as some hard anomalies that need further refinement. Therefore, initially, all the videos in the training set are assigned to the normal cluster and the anomaly cluster is initialized to an empty set, i.e., $\mathcal{C}_0^0 = \{\mathbf{x}_i\}_{i \in [1,n]}$ and $\mathcal{C}_1^0 = \varnothing$. At each step t ($t \geq 1$), the cluster $\mathcal{C}_0^{t-1}$ is re-clustered to obtain two new child clusters, say $\mathcal{C}_l$ and $\mathcal{C}_s$ with $|\mathcal{C}_l|$ and $|\mathcal{C}_s|$ samples, respectively. Without loss of generality, let $|\mathcal{C}_s| < |\mathcal{C}_l|$. The smaller cluster $\mathcal{C}_s$ is merged with the previous anomaly cluster, i.e., $\mathcal{C}_1^t = (\mathcal{C}_1^{t-1} \cup \mathcal{C}_s)$, while the larger cluster is labeled as normal, i.e., $\mathcal{C}_0^t = \mathcal{C}_l$. This process is repeated until the ratio of the number of videos in the anomaly cluster ($|\mathcal{C}_1^t|$) to the number of videos in the normal cluster ($|\mathcal{C}_0^t|$) is larger than a threshold, i.e., $\frac{|\mathcal{C}_1^t|}{|\mathcal{C}_0^t|} > \eta$. At the end of the CPL stage, all the videos in the training set are assigned a pseudo-label based on their corresponding cluster index, i.e., $\hat{y}_i = k$, if $\mathbf{x}_i \in \mathcal{C}_k^\tau$, where $k \in \{0, 1\}$ and τ denotes the final clustering iteration.

3.2. Fine (Segment-Level) Pseudo-Label Generator

All the segments from videos that are "pseudo-labeled" as normal ($\hat{y}_i = 0$) by the previous stage can be considered as normal. However, most of the segments in an anomalous video are also normal due to temporal localization of anomalies. Hence, further refinement of the coarse (video-level) labels is required to generate segment-level labels for anomalous videos. To achieve this goal, we treat the detection of anomalous segments as a statistical hypothesis testing problem. Specifically, the null hypothesis is that a given video segment is normal. By modeling the distribution of features under the null hypothesis as a Gaussian distribution, we identify the anomalous segments by estimating their p-value and rejecting the null hypothesis if the p-value is less than the significance level α.

To model the distribution of features under the null hypothesis, we consider only the segments from videos that are pseudo-labeled as normal by the CPL stage. Let $\mathbf{z}_{ij} \in \mathbb{R}^{\tilde{d}}$ be a low-dimensional representation of a segment S_{ij}. We assume that $\mathbf{z}_{ij}$ follows a Gaussian distribution $\mathcal{N}(\mathbf{\Gamma}, \mathbf{\Sigma})$ under the null hypothesis and estimate the parameters $\mathbf{\Gamma}$ and $\mathbf{\Sigma}$ as follows:

Algorithm 1 Coarse-to-Fine Pseudo-Label Generation

Input: Training dataset $\mathcal{D} = \{V_1, \cdots, V_n\}$, pre-trained feature extractor $\mathcal{F}(\cdot)$, parameters η, β
Output: Segment-level pseudo-labels $\{\tilde{y}_{ij}\}$, where $i \in [1, n]$ and $j \in [1, m_i]$

1: **for** $i = 1$ to n **do**
2:      Partition V_i into m_i segments $[S_{i1}, \cdots, S_{im_i}]$
3:      Extract segment features $[\mathbf{f}_{i1}, \cdots, \mathbf{f}_{im_i}]$ using $\mathcal{F}(\cdot)$
4:      Compute $\mathbf{x}_i = [\mu_i, \sigma_i]$ using Eqs. 1 & 2
5: **end for**
6: **CPL**: $t = 0$, $\mathcal{C}_0^t = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$, $\mathcal{C}_1^t = \varnothing$
7: **while** $|\mathcal{C}_1^t| / |\mathcal{C}_0^t| \leq \eta$ **do**
8:      $(\mathcal{C}_s, \mathcal{C}_l) \leftarrow$ Clustering($\mathcal{C}_0^t$), where $|\mathcal{C}_s| < |\mathcal{C}_l|$
9:      $\mathcal{C}_1^{t+1} \leftarrow \mathcal{C}_1^t \cup \mathcal{C}_s$, $\mathcal{C}_0^t \leftarrow \mathcal{C}_l$
10:      $t \leftarrow t + 1$
11: **end while**
12: $\forall i \in [1, n], \hat{y}_i \leftarrow 0$ if $\mathbf{x}_i \in \mathcal{C}_0^t$, **else** $\hat{y}_i \leftarrow 1$
13: **FPL**: $\forall i \in [1, n], j \in [1, m_i], \tilde{y}_{ij} \leftarrow 0$, Compute $\mathbf{z}_{ij}$
14: Compute $(\mathbf{\Gamma}, \mathbf{\Sigma})$ using Eqs. 3 & 4
15: **for** $i = 1$ to n **do**
16:      **if** $\hat{y}_i = 1$ **then**
17:          Compute p_{ij} using Eq. 5, $\forall j \in [1, m_i]$
18:          $w_i \leftarrow \lceil \beta m_i \rceil$
19:          $l_i = \arg\min_l \left\{ \frac{1}{w_i} \sum_{j=(l+1)}^{(l+w_i)} p_{ij}, \ \forall l \in [0, m_i - w_i] \right\}$
20:          $\tilde{y}_{ij} \leftarrow 1, \ \forall j \in [l_i + 1, l_i + w]$
21:      **end if**
22: **end for**

$$\mathbf{\Gamma} = \frac{1}{M_0} \sum_{i=1, \hat{y}_i=0}^{n} \sum_{j=1}^{m_i} \mathbf{z}_{ij}, \tag{3}$$

$$\mathbf{\Sigma} = \frac{1}{(M_0 - 1)} \sum_{i=1, \hat{y}_i=0}^{n} \sum_{j=1}^{m_i} (\mathbf{z}_{ij} - \mathbf{\Gamma})(\mathbf{z}_{ij} - \mathbf{\Gamma})^T, \tag{4}$$

where $M_0 = \sum_{i=1, \hat{y}_i=0}^{n} m_i$. Subsequently, for all the segments in videos that are pseudo-labeled as anomalous, the p-value is computed as:

$$p_{ij} = \frac{1}{(2\pi)^{(\tilde{d}/2)} \sqrt{|\Sigma|}} \exp\left(-\frac{1}{2} (\mathbf{z}_{ij} - \mathbf{\Gamma})^T \Sigma^{-1} (\mathbf{z}_{ij} - \mathbf{\Gamma}) \right), \tag{5}$$

$\forall j \in [1, m_i], i \in [1, n]$ such that $\hat{y}_i = 1$. If $p_{ij} < \alpha$, the segment can be potentially assigned a pseudo-label of 1. Figure 3 shows an illustration of this approach, which clearly indicates strong agreement between the estimated p-values and the ground truth anomaly labels of the validation set.

One unresolved question in the above formulation is how to obtain the low-dimensional representation $\mathbf{z}_{ij}$ for a segment S_{ij}. In this work, we simply set $\mathbf{z}_{ij} = \|\mathbf{f}_{ij}\|_2$ and hence
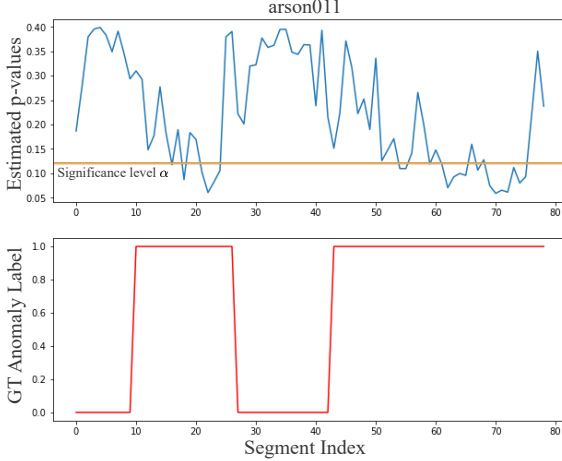
Figure 3. Statistical hypothesis testing approach applied to pseudo-label an anomalous video **arson011** from the *validation set*. The top row shows the segment-level p-values for all the segments in the video, where a lower p-value means less likelihood of being normal. A possible pseudo-labeling strategy is to mark all segments with a p-value lower than the significance level (denoted by the horizontal orange line) as anomalous. Strong agreement can be observed between the estimated p-values and the ground-truth (GT) anomaly label for the given video shown in the bottom row.

$\tilde{d} = 1$. Note that other statistics could also be employed in addition to (or in lieu of) the ℓ_2 feature magnitude.

Directly assigning a pseudo-label to a segment based on its p-value ignores the reality that anomalous segments in a video tend to be temporally contiguous. One way to overcome this limitation is to mark a contiguous sequence of $w_i = \lceil \beta m_i \rceil$ segments, $0 < \beta < 1$ and $\lceil \cdot \rceil$ represents the ceil function, as the anomalous region within each video that is pseudo-labeled as an anomaly. The anomalous region is determined by sliding a window of size w_i across the video and selecting the window that has the lowest average p-values (i.e., $\min_l \left\{ \frac{1}{w_i} \sum_{j=(l+1)}^{(l+w_i)} p_{ij}, \ \forall \ l \in [0, m_i - w_i] \right\}$). Each segment present in this anomalous region is assigned a pseudo-label of 1, while all the remaining segments are pseudo-labeled as normal (value of 0). At the end of this FPL stage, a pseudo-label $\tilde{y}_{ij} \in \{0, 1\}$ is assigned to all the segments in the training set.

3.3. Anomaly Detector

The coarse and fine pseudo-label generators together provide a pseudo-label for every video segment in the training dataset. This results in a pseudo-labeled training set $\tilde{\mathcal{D}} = \{(\mathbf{f}_{ij}, \tilde{y}_{ij})\}$ containing M samples, where $i \in [1, n]$, $j \in [1, m_i]$, and $M = \sum_{i=1}^{n} m_i$. This labeled training set $\tilde{\mathcal{D}}$ can be used to train the anomaly detector $\tilde{\mathcal{A}}_\theta(\cdot)$ in a supervised fashion by minimizing the following objective:

$$\min_\theta \sum_{i=1}^{n} \sum_{j=1}^{m_i} \mathcal{L}(\mathcal{A}_\theta(\mathbf{f}_{ij}), \tilde{y}_{ij}), \qquad (6)$$

where $\mathcal{L}$ is an appropriate loss function and θ denotes the parameters of the anomaly detector $\tilde{\mathcal{A}}(\cdot)$.

Following recent state-of-the-art methods [19,25,35,36], two basic neural network architectures are considered for our anomaly detector. In particular, we employ a shallow neural network (Figure 2) with two fully connected (FC) hidden layers and one output layer mapped to a binary class. A dropout layer and a ReLU activation function are applied after each FC layer. Additionally, following Zaheer *et al*. [35], we add two self-attention layers (detailed architecture is provided in the Supplementary material). A softmax activation function follows each of the self-attention layers, each of which has the same dimensions as the corresponding FC layer in the backbone network. Final anomaly score prediction is produced by the output sigmoid function.

Unlike many existing methods (e.g., [19]) that require having a complete video in one training batch, our approach allows random segment selection for training. Recently, Zaheer *et al*. [35] demonstrated the benefits of feature vector randomization for training. However, based on its design, their method was limited to randomizing consecutive batches while maintaining the temporal order of segments within a batch. In our case, since we have obtained pseudo-labels for each segment, we can apply training with complete randomization to reap maximum benefits. Therefore, feature vectors are obtained across the dataset to form the training batches. Formally, each training batch $\mathcal{B}$ contains B randomly selected samples from the set $\tilde{\mathcal{D}}$ without any order constraints between the samples.

3.4. Inference

During inference, a given test video V_* is partitioned into m_* non-overlapping segments S_{*j}, $j \in [1, m_*]$. Feature vectors $\mathbf{f}_{*j}$ are extracted from each segment using $\mathcal{F}(\cdot)$, which are directly passed to the trained detector $\tilde{\mathcal{A}}_\theta(\cdot)$ to obtain segment-level anomaly score predictions. Since the eventual goal is frame-level anomaly prediction, all the frames within a segment of the test video are marked as anomalous if the predicted anomaly score for that corresponding segment exceeds a threshold.

4. Experimental Results

4.1. Experimental Setup

Datasets: Two large-scale VAD datasets are used to evaluate our approach: UCF-Crime [19] and XD-Violence [31]. **UCF-Crime** consists of 1610 (290) training (test) videos collected from real-world surveillance camera feeds, totaling 128 hours in length. **XD-Violence** is a multi-modal VAD dataset that is collected from sports streaming videos,

Supervision	Method	Features	FNS	AUC(%)
OCC	SVM [19]	I3D	-	50
	Hasan et al. [7]	-	-	50.60
	SSV [18]	-	-	58.50
	BODS [29]	I3D	-	68.26
	GODS [29]	I3D	-	70.46
	SACR [20]	-	-	72.70
	Zaheer et al. [36]	ResNext	✗	74.20
WS	Sultani et al.† [19]	I3D	✓	77.92
	Zaheer et al. [36]	ResNext	✗	79.84
	RTFM [25]	I3D	✓	84.30
	MSL [11]	I3D	✓	85.30
	S3R [30]	I3D	✓	85.99
	C2FPL* (Ours)	I3D	✗	85.5
US	Kim et al. [10]	ResNext	-	52.00
	Zaheer et al. [36]	ResNext	✗	71.04
	DyAnNet [24]	I3D	✓	79.76
	C2FPL (Ours)	I3D	✗	80.65

Table 1. Frame-level AUC performance comparison on UCF-Crime dataset. Wherever available, RGB results are reported. Our unsupervised C2FPL method is compared against both unsupervised and supervised (WS and OCC) methods. The column FNS indicates whether the method uses a fixed number of segments m (m = 32 when FNS is true) and '-' indicates this information is not available. The top two results under each supervision setting are shown in blue and red in that order. † indicates that results are reported from [25], where the method in [19] was retrained using I3D features.

Supervision	Method	Features	FNS	AUC(%)
OCC	Hasan et al. [7]	AE	-	50.32
	Lu et al. [12]	I3D	-	53.56
	BODS [29]	I3D	-	57.32
	GODS [29]	I3D	-	61.56
WS	S3R [30]	I3D	✓	53.52
	RTFM† [25]	I3D	✓	89.34
	C2FPL* (Ours)	I3D	✗	90.4
US	RareAnom [23]	I3D	✓	68.33
	C2FPL (Ours)	I3D	✗	80.09

Table 2. Frame-level AUC performance comparison on XD-Violence dataset. The column FNS indicates whether the method uses a fixed number of segments m (m = 32 when FNS is true) and "-" indicates this information is not available. The top two results under each supervision setting are shown in blue and red in that order. † indicates that we re-compute the AUC of method in [25] using I3D features.

movies, web videos, and surveillance cameras. It consists of 3954 (800) training (test) videos that span around 217 hours. We utilize only the visual modality of the XD-Violence dataset for our experiments. Both these datasets originally contain video-level ground-truth labels for the training set and frame-level labels for the test set. Hence, they are primarily meant for the WS-VAD task. In this work, we ignore the training labels and only use test labels to evaluate our US-VAD model.

Evaluation Metric: We adopt the commonly used frame-level area under the receiver operating characteristic curve (AUC) as the evaluation metric for all our experiments [19, 23, 25, 33–35]. Note that the ROC curve is obtained by varying the threshold on the anomaly score during inference and higher AUC values indicate better results.

Implementation Details: Each video is partitioned into multiple segments, with each segment containing r = 16 frames. The well-known I3D [3] method is used as the pre-trained feature extractor $\mathcal{F}(\cdot)$ to extract RGB features with dimensionality d = 2048. Following [22], we also apply 10-crop augmentation to the I3D features. The CPL generator uses Gaussian Mixture Model (GMM)-based clustering [17] and the threshold η is set to 1.0. The parameter β used in the FPL generator is set to 0.2. The anomaly detec-

tor $\tilde{\mathcal{A}}_\theta(\cdot)$ is trained using a binary cross-entropy loss function along with ℓ_2 regularization. The detector is trained for 100 epochs using a stochastic gradient descent optimizer with a learning rate of 0.01. The batch size B is set to 128.

4.2. Comparison with state-of-the-art

In this section, we provide performance comparison of our proposed unsupervised C2FPL method with recent state-of-the-art (SOTA) supervised and unsupervised VAD methods [7, 18, 19, 21, 29, 36, 37].

UCF-Crime. The AUC results on the UCF-Crime dataset are shown in Table 1. Wherever possible, results based on I3D RGB features are reported to ensure a fair comparison. The proposed C2FPL method achieves an AUC performance of 80.65%, outperforming the existing US and OCC methods while performing comparably to existing SOTA WS methods. Note that OCC methods assume that the training data contains only normal videos, while we do not make any such assumption. Furthermore, our unsupervised C2FPL framework even outperforms some methods in the WS setting [19, 36, 37], thus bridging the gap between unsupervised and supervised approaches. However, compared to the top performing WS method S3R [30] using the same I3D features, our approach yields 5.34% lower AUC. While this is impressive considering that our method does not require any supervision, it highlights the need for further improvement in the accuracy of the CPL stage.

XD-Violence. Our C2FPL framework is also evaluated on XD-Violence dataset and the results are reported in Table 2. The proposed method has an AUC of 80.09%, which is significantly better than the unsupervised RareAnom [23] method. Additionally, our framework achieves good results even in comparison to other OCC and WS methods.
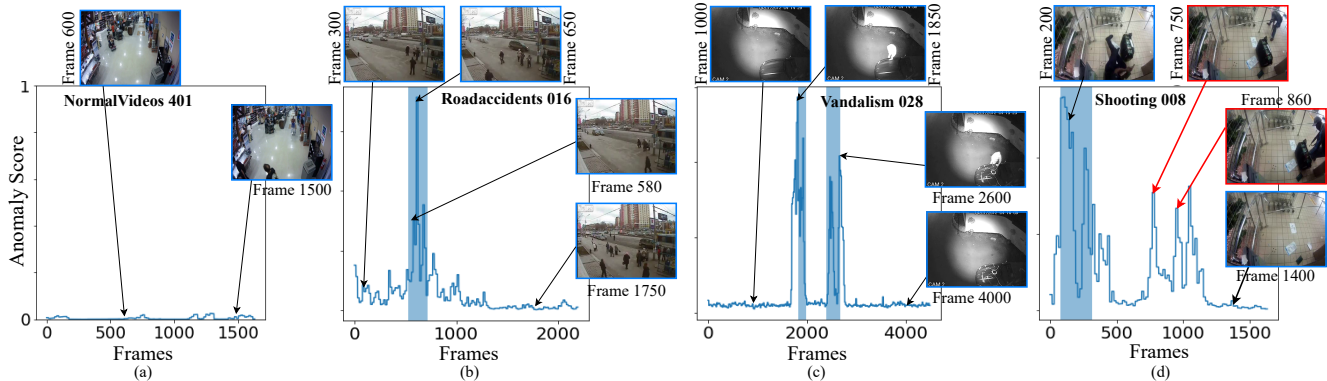
Figure 4. Qualitative results of our method on different test videos of the UCF-Crime dataset. The blue color shadow shows the ground truth anomalous frames.

Stage 1 (CPL)	Stage 2 (FPL)	Stage 3 (AD)	Scenario	AUC (%)
✓	✓	✓	US C2FPL framework	80.6
✗	✓	✓	Ground-truth video-level labels (WS)	85.5
✗	✓	✓	Random video-level labels	69.4
✓	✗	✓	CPL pseudo-labels assigned to segments	64.1
✗	✗	✓	Ground-truth video-level labels assigned to segments	72.7
✗	✗	✓	Random segment-level labels	38.7
✓	✓	✗	(1 – p-value) as anomaly score	57.0

Table 3. Ablation studies analyzing the impact of each component of the proposed approach on the UCF-Crime dataset.

Qualitative Results: We also provide some qualitative results in Figure 4, where anomaly scores predicted by our C2FPL approach are visualized for several videos from the UCF-Crime dataset. It can be observed that the predicted anomaly scores generally correlate well to the anomaly ground truth in many cases, demonstrating the good anomaly detection capability of our approach despite being trained without any supervision. A **failure case**, shooting008 video (UCF-Crime), is also visualized in Figure 4(d). Our detector predicts several frames after the actual shooting event as anomalous. Careful inspection of this video shows a person with a gun entering the scene after the actual event, which our method marks as anomalous, but the ground-truth frame label is normal. Such discrepancies affect the frame-level AUC.

4.3. Ablation Study

Next, we conduct a detailed ablation study to analyze the impact of each component of the proposed C2FPL framework for US-VAD using the UCF-Crime dataset.

Impact of CPL: The objective of CPL is to generate coarse video-level labels for all videos in the training dataset. To evaluate the impact of this component, we carry out two experiments and report the results in Table 3. In the first experiment, the CPL stage is removed and the video-level pseudo-labels are assigned randomly. In this case, the performance drops significantly to 69.4% indicating that the coarse pseudo-labels generated by CPL are indeed very useful in guiding the subsequent stages of the proposed system. On the other extreme, we also experimented with using the ground-truth video-level labels instead of the generated coarse pseudo-labels. Note that this setting is equivalent to WS training used widely in the literature. As expected, the performance improves to 85.5%, which is almost on par with the best WS method S3R [30] using the same I3D features (see Table 1). On the XD-Violence dataset, the C2FPL method adapted for the WS setting achieves an AUC of 90.4%, which is better than existing WS methods on the same dataset. These results highlight the potential improvement that can be achieved by improving the accuracy of the CPL stage. It also demonstrates the ability of our proposed approach to learn without labels, but at the same time exploit the ground-truth WS labels when they are available.

Impact of FPL: Since the goal of FPL is to obtain segment-level labels, we consider the following three scenarios. Firstly, when C2FPL framework is completely ignored and the segment-level pseudo-labels are assigned randomly, the performance of the trained anomaly detector collapses to a very low AUC of 38.72%. This experiment proves that the generated segment-level pseudo-labels are indeed very informative and aid the training of an accurate anomaly detector. Secondly, we ignore only the FPL stage and assign the coarse video-level labels obtained from CPL to all the segments in the corresponding video. There is still a substantial performance drop to 64.1% (from 80.65% when FPL is used). Finally, we again consider the WS setting and assign the ground-truth video-level labels to all the segments

in a video. Even in this case, the performance improves only to 72.7% (compared to 85.5% when FPL is used in the WS setting). The last two results clearly prove that the use of FPL reduces segment-level label noise to a large extent, thereby facilitating better training of the anomaly detector.

Impact of Anomaly Detector: To understand the impact of the segment-level anomaly detector, we excluded the detector and directly used (1–p-value) obtained during the FPL stage as the anomaly score. This results in a significant drop in AUC to 57.0%, which indicates that while the C2FPL framework can generate informative pseudo-labels, these labels are still quite noisy and cannot be directly used for frame-level anomaly prediction. The anomaly detector is critical to learn from these noisy pseudo-labels and make more accurate fine-grained predictions.

Method	Sultani [19]	Zaheer [36]	RTFM [25]	S3R [30]	Ours
Params	1.07M	6.5M	24.72M	73.5M	2.13M

Table 4. Number of trainable parameters of the proposed approach in comparison with some existing methods. Our approach achieves good performance with significantly fewer parameters.

4.4. Parameter Sensitivity Analysis

Sensitivity to η: The sensitivity of the proposed method to the value of η is studied on the UCF-Crime dataset. The best results are achieved when $\eta = 1$, which corresponds to having a roughly equal number of videos in the normal and anomaly clusters. When $\eta = 0.5$ or $\eta = 1.5$, the AUC drops to 75.64% and 71.22%, respectively. It is important to emphasize that though the number of normal segments in a dataset is usually much larger than the number of anomalous segments, the number of normal and anomalous videos in the available datasets are roughly equal. For example, UCF-Crime has 800 normal videos and 810 anomalous videos, while XD-Violence has 2049 normal videos and 1905 anomalous videos. Therefore, the choice of $\eta = 1$ is appropriate for these two datasets. In real-world unsupervised settings, the ratio of anomalous to normal videos in a given dataset may not be known in advance because there are no labels. When η is mis-specified, there is some performance degradation, which is a limitation of the proposed C2FPL approach.

Sensitivity to β: In the FPL stage, a window of size $\lceil \beta m_i \rceil$ is used to *loosely* incorporate the temporal contiguity constraint. In the earlier experiments, β was set to 0.2 (20% of the video length). However, in practice, the number of anomalous segments in a video may vary widely and would not be known in advance. The sensitivity of the proposed
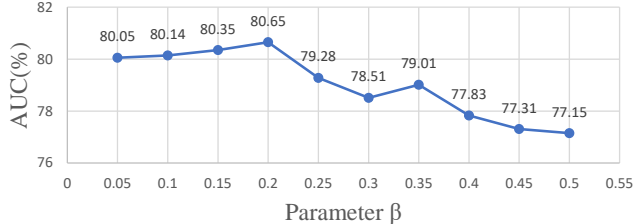

Figure 5. Sensitivity of C2FPL framework to parameter β.

method to the value of β is shown in Figure 5. These results indicate that our method is quite robust to changes in β.

Fixed number of segments: We hypothesize that not compressing the videos at test/train time, as commonly done in the existing literature [19, 26], is beneficial for the overall anomaly detection performance. To validate this hypothesis, we experiment with compressing each video into to a fixed number of segments $m = 32$ before applying the proposed method. With such compression, the performance of our method drops to 77.70% and 78.08% for UCF-Crime and XD-Violence datasets, respectively. This justifies our choice of not using any compression.

4.5. Computational Complexity Analysis

Apart from feature extraction, the proposed C2FPL training method requires a few invocations of the GMM clustering subroutine, a single round of Gaussian distribution fitting, and training of the segment-level anomaly detector $\tilde{\mathcal{A}}_\theta$. Since GMM clustering is performed at the video level on 2D data, the computational cost of the two-stage pseudo-label generator is insignificant (0.6 seconds) compared to that of the anomaly detector training (60 seconds per epoch). As seen in Figure 2, the architecture of $\tilde{\mathcal{A}}_\theta$ is fairly simple with only 2.13M parameters, which is significantly lower than all SOTA methods except Sultani *et al*. [19], as shown in Table 4. It may be noted that, despite having fewer parameters, the WS variant of our approach outperforms almost all the other methods on both datasets (Table 1 & 2). The only exception is S3R, which has 0.5% higher AUC compared to our approach, while having over 71M extra parameters than our method. During inference, our method achieves 70 frames per second (fps) on NVIDIA RTX A6000 which is almost double the rate of real-time applications. This indicates that our system can achieve good real-time detection in real-world scenarios.

5. Conclusion

Unsupervised video anomaly detection (US-VAD) methods are highly useful in real-world applications as a complete system can be trained without any annotation or human intervention. In this work, we propose a US-VAD approach based on a two-stage pseudo-label generator that

facilitates the training of a segment-level anomaly detector. Extensive experiments conducted on two large-scale datasets, XD-Violence and UCF-Crime, demonstrate that the proposed approach can successfully reduce the gap between unsupervised and supervised approaches.

References

[1] Waqar Ahmed, Pietro Morerio, and Vittorio Murino. Adaptive pseudo-label refinement by negative ensemble learning for source-free unsupervised domain adaptation. *arXiv preprint arXiv:2103.15973*, 2021.

[2] Borislav Antić and Björn Ommer. Video parsing for abnormality detection. In *2011 International conference on computer vision*, pages 2415–2422. IEEE, 2011. 1

[3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 6

[4] Yoonki Cho, Woo Jae Kim, Seunghoon Hong, and Sung-Eui Yoon. Part-based pseudo label refinement for unsupervised person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7308–7318, 2022. 3

[5] Jia-Chang Feng, Fa-Ting Hong, and Wei-Shi Zheng. Mist: Multiple instance self-training framework for video anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14009–14018, 2021. 1

[6] Nico Görnitz, Marius Kloft, Konrad Rieck, and Ulf Brefeld. Toward supervised anomaly detection. *Journal of Artificial Intelligence Research*, 46:235–262, 2013. 2

[7] 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, 6

[8] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 11

[9] Shunsuke Kamijo, Yasuyuki Matsushita, Katsushi Ikeuchi, and Masao Sakauchi. Traffic monitoring and accident detection at intersections. *IEEE transactions on Intelligent transportation systems*, 1(2):108–118, 2000. 1

[10] Jin-Hwa Kim, Do-Hyeong Kim, Saehoon Yi, and Taehoon Lee. Semi-orthogonal embedding for efficient unsupervised anomaly segmentation. *arXiv preprint arXiv:2105.14737*, 2021. 6

[11] Shuo Li, Fang Liu, and Licheng Jiao. Self-training multisequence learning with transformer for weakly supervised video anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1395–1403, 2022. 6

[12] 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. 2, 6

[13] 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. 1

[14] Sadegh Mohammadi, Alessandro Perina, Hamed Kiani, and Vittorio Murino. Angry crowds: Detecting violent events in videos. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 3–18. Springer, 2016. 1

[15] Romero Morais, Vuong Le, Truyen Tran, Budhaditya Saha, Moussa Mansour, and Svetha Venkatesh. Learning regularity in skeleton trajectories for anomaly detection in videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11996–12004, 2019. 1

[16] Didik Purwanto, Yie-Tarng Chen, and Wen-Hsien Fang. Dance with self-attention: A new look of conditional random fields on anomaly detection in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 173–183, October 2021. 2, 3

[17] Douglas A Reynolds et al. Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663), 2009. 6

[18] Fahad Sohrab, Jenni Raitoharju, Moncef Gabbouj, and Alexandros Iosifidis. Subspace support vector data description. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 722–727. IEEE, 2018. 6

[19] 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. 1, 2, 3, 5, 6, 8

[20] 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*, pages 184–192, 2020. 6

[21] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999. 6

[22] Ryo Takahashi, Takashi Matsubara, and Kuniaki Uehara. Data augmentation using random image cropping and patching for deep cnns. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(9):2917–2931, 2019. 6

[23] Kamalakar Vijay Thakare, Debi Prosad Dogra, Heeseung Choi, Haksub Kim, and Ig-Jae Kim. Rareanom: A benchmark video dataset for rare type anomalies. *Pattern Recognition*, 140:109567, 2023. 2, 6

[24] Kamalakar Vijay Thakare, Yash Raghuwanshi, Debi Prosad Dogra, Heeseung Choi, and Ig-Jae Kim. Dyannet: A scene dynamicity guided self-trained video anomaly detection network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5541–5550, 2023. 6

[25] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4975–4986, 2021. 1, 2, 3, 5, 6, 8

[26] 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. *arXiv preprint arXiv:2101.10030*, 2021. 3, 8

[27] Maximilian E Tschuchnig and Michael Gadermayr. Anomaly detection in medical imaging-a mini review. In *Data Science–Analytics and Applications: Proceedings of the 4th International Data Science Conference–iDSC2021*, pages 33–38. Springer, 2022. 2

[28] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2017. 11

[29] Jue Wang and Anoop Cherian. Gods: Generalized one-class discriminative subspaces for anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8201–8211, 2019. 2, 6

[30] Jhih-Ciang Wu, He-Yen Hsieh, Ding-Jie Chen, Chiou-Shann Fuh, and Tyng-Luh Liu. Self-supervised sparse representation for video anomaly detection. In *Computer Vision– ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIII*, pages 729– 745. Springer, 2022. 2, 3, 6, 7, 8

[31] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pages 322–339. Springer, 2020. 2, 5

[32] Muhammad Zaigham Zaheer, Jin-ha Lee, Marcella Astrid, and Seung-Ik Lee. Old is gold: Redefining the adversarially learned one-class classifier training paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14183–14193, 2020. 2

[33] Muhammad Zaigham Zaheer, Jin-ha Lee, Marcella Astrid, Arif Mahmood, and Seung-Ik Lee. Cleaning label noise with clusters for minimally supervised anomaly detection. In *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020. 6

[34] Muhammad Zaigham Zaheer, Jin Ha Lee, Arif Mahmood, Marcella Astrid, and Seung-Ik Lee. Stabilizing adversarially learned one-class novelty detection using pseudo anomalies, 2022. 6

[35] Muhammad Zaigham Zaheer, Arif Mahmood, Marcella Astrid, and Seung-Ik Lee. Claws: Clustering assisted weakly supervised learning with normalcy suppression for anomalous event detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 358–376. Springer, 2020. 1, 2, 5, 6, 11, 12

[36] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision*

and Pattern Recognition*, pages 14744–14754, 2022. 2, 5, 6, 8

[37] Jiangong Zhang, Laiyun Qing, and Jun Miao. Temporal convolutional network with complementary inner bag loss for weakly supervised anomaly detection. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 4030–4034. IEEE, 2019. 6

[38] Xiao Zhang, Yixiao Ge, Yu Qiao, and Hongsheng Li. Refining pseudo labels with clustering consensus over generations for unsupervised object re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3436–3445, 2021. 3

[39] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1237–1246, 2019. 1, 2

Appendices

A. Self Attention

Figure 6 shows the detailed architecture of our proposed C2FPL network. The FC layers described in manuscript: Section 3.3 have 512 and 32 neurons where each is followed by a ReLU activation function and a dropout layer with a dropout rate of 0.6. In addition, we add two self-attention layers. In this section, we will discuss the choice design as well as the aim of using this layer.

The aim of self-attention (SA) in our proposed C2FPL framework is to highlight parts of feature vectors critical in detecting anomalies. Our configuration applies self-attention over each feature vector (feature dimension) independently without requiring temporal order. This is unlike a compareable existing architecture by Zaheer *et al.* [35] where the Normalcy Suppression Module (NSM) aims to learn attention based on the temporally consistent feature vectors in the input batch (Figure 8(a)) and the attention is calculated along the batch dimension (temporal axis).

To study this in details, we define several possible configurations of the self-attention used in our C2FPL and report their performances in this section. Through thorough analysis, we verify the effectiveness of our design choices within the framework.

A.1. Residual vs Multiplicative Self-Attention (SA)

Zaheer *et al.* [35], in CLAWS Net, formulate the problem of self-attention in terms of suppressing certain features which are achieved by multiplicative attention. To provide a comparison, we discuss two different SA configurations as shown in Figure 7. First, following Zaheer *et al.* [35], given an input batch b we calculate the output $H(b)$ by performing an element-wise multiplication $\otimes$ between SA output $S(b)$ and backbone output $FC(b)$ as:

$$H(b) = S(b) \otimes FC(B)$$

Although such multiplication has been helpful in CLAWS Net, generally it has been shown to have the unfavorable result of dissipating model representations [8, 28]. It's because attention generates probabilities that, when

Framework	SA configuration	AUC (%)
CPL → FPL → AD	Multiplicative	63.5
	Residual (Ours)	80.6

Table 5. Area under the curve (AUC) comparison of two SA configurations configurations on the UCF-Crime dataset. (The framework configuration is the same as shown in manuscript: Table 3).

Framework	SA Dimension	AUC (%)
CPL → FPL → AD	Batch Dimension	76.5
	Feature Dimension (Ours)	80.6

Table 6. Area under the curve (AUC) comparison of two SA types on UCF-Crime dataset. (The framework is the same as shown in manuscript: Table 3).
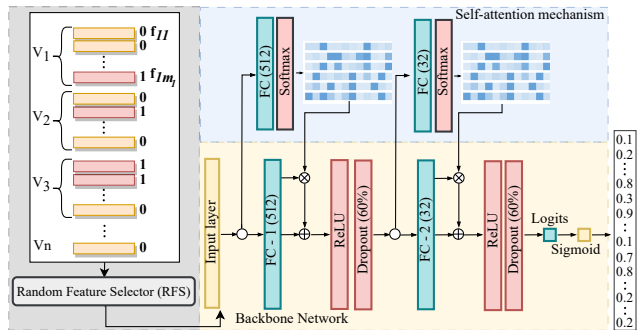

Figure 6. Detailed architecture of our proposed learning network: The training batch containing pseudo-labeled feature vectors is the input to the FC backbone network (lower). In addition to the backbone network, we add two self-attention layers (upper).

multiplied by the features directly, can drastically lower the values.

In our framework, we utilize residual SA in which attention-applied features are added back to the original features. Therefore, The output $H(b)$ is calculated as:

$$H(b) = (FC(b) \otimes S(b)) \oplus FC(b)$$

where $\oplus$ is an addition operation.

Table 5 shows the performance difference between multiplication and residual attention approaches. We can observe that the use of multiplication negatively affects our model's AUC performance (63.5%). We attribute this to the suppression nature of multiplication [8,28]. The specifically designed NSM of CLAWS Net [35] aims to dissipate normal portions of the temporally consistent input batches that help the backbone network produce low anomaly scores. However, the nature of our training is not suitable for this formulation. Therefore, using residual attention, which only highlights individual parts of each feature vector in a given batch, the performance of our model increases to 80.6% on the UCF-crime dataset.

A.2. Types of self-attention

In conjunction with Zaheer *et al.* [35], We discuss two different types of self-attentions depending on the dimensions along which Softmax probabilities are computed in an element-wise fashion.
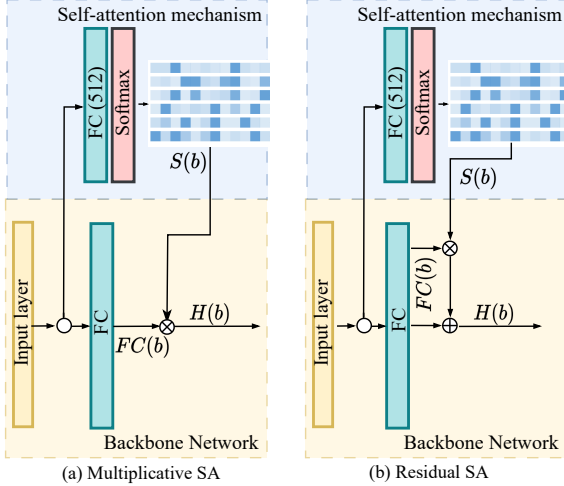
Figure 7. Visualization of the two self-attention configurations including (a) Multiplicative SA and our proposed (b) Residual SA.
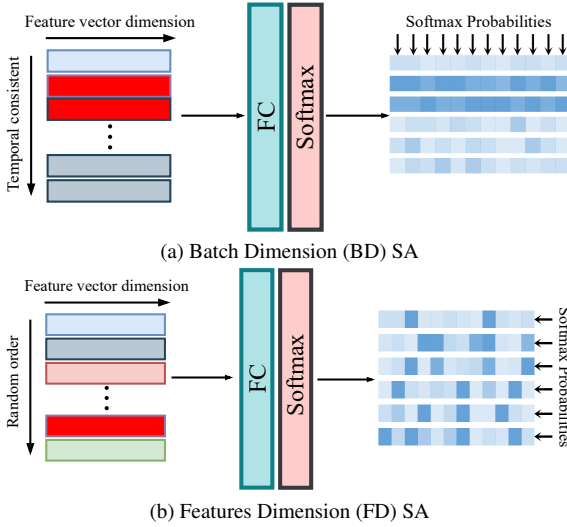

Figure 8. Visualization of the two types of self-attention: (a) Batch Dimension (BD): Softmax probabilities are calculated along the Batch dimension (temporal axis). (b) Features Dimension (FD): Softmax probabilities are calculated along the feature vector dimension.

Softmax probabilities over the batch dimension (BD). As mentioned, Zaheer *et al.* [35] calculates the probabilities temporally to make use of the temporal information preserved within a batch (Figure 8 (a)). However, we have argued and demonstrated in our presented C2FPL framework that preserving temporal information is not necessary for improved anomaly detection performance. Therefore, using temporal attention along the batch dimension may not be as effective in our framework as it has been proven in CLAWS Net by Zaheer *et al.* [35]. Nevertheless, we utilize

their proposed self-attention and compare it with our design of self-attention.

Softmax probabilities over the feature dimension (FD). This self-attention over feature dimension (FD) is the configuration used in our C2FPL framework, as explained in manuscript: Section 3.3 (lines 494-500). Since we assume no temporal consistency among batches, the probabilities are computed over the feature dimension (Figure 8 (b)).

Table 6 summarizes the frame-level AUC performance of the two types. It can be seen that the FD type (ours) outperforms the BD type attention by a margin of 4.1%. This verifies the importance of using self-attention along the feature vector dimension, achieving significant performance gains.

B. Qualitative Results

We also provide additional qualitative results in Figure 9, where anomaly scores predicted by our C2FPL approach are visualized for other classes of anomalous videos from the UCF-Crime dataset. In some cases, the anomalous frames in certain videos might exceed the annotated ones because the annotations only cover a portion of the event. For instance, the abnormal event in the RoadAccidents004 video begins at about frame 145 and lasts significantly longer than the annotated window, which only shows the accident impact event.

An additional **failure case**, shooting034 video (UCF-Crime), is also visualized in Figure 9(h). Our proposed model correctly predicts the ground-truth anomalous window. However, later frames (1200) of the video show one of the occupants involved in the shooting quickly entering his car before speeding off, which our detector marks as an anomalous event while that event is annotated as a normal event.

C. Convergence Analysis

As our approach is an unsupervised anomaly detection method, we empirically analyze its convergence using 10 random seed runs as shown in Figure 10. For all experiments, our C2FPL model attains an average AUC of 80.14% ± 0.31%. This demonstrate that our proposed framework not only achieves excellent anomaly detection but also demonstrates good convergence.
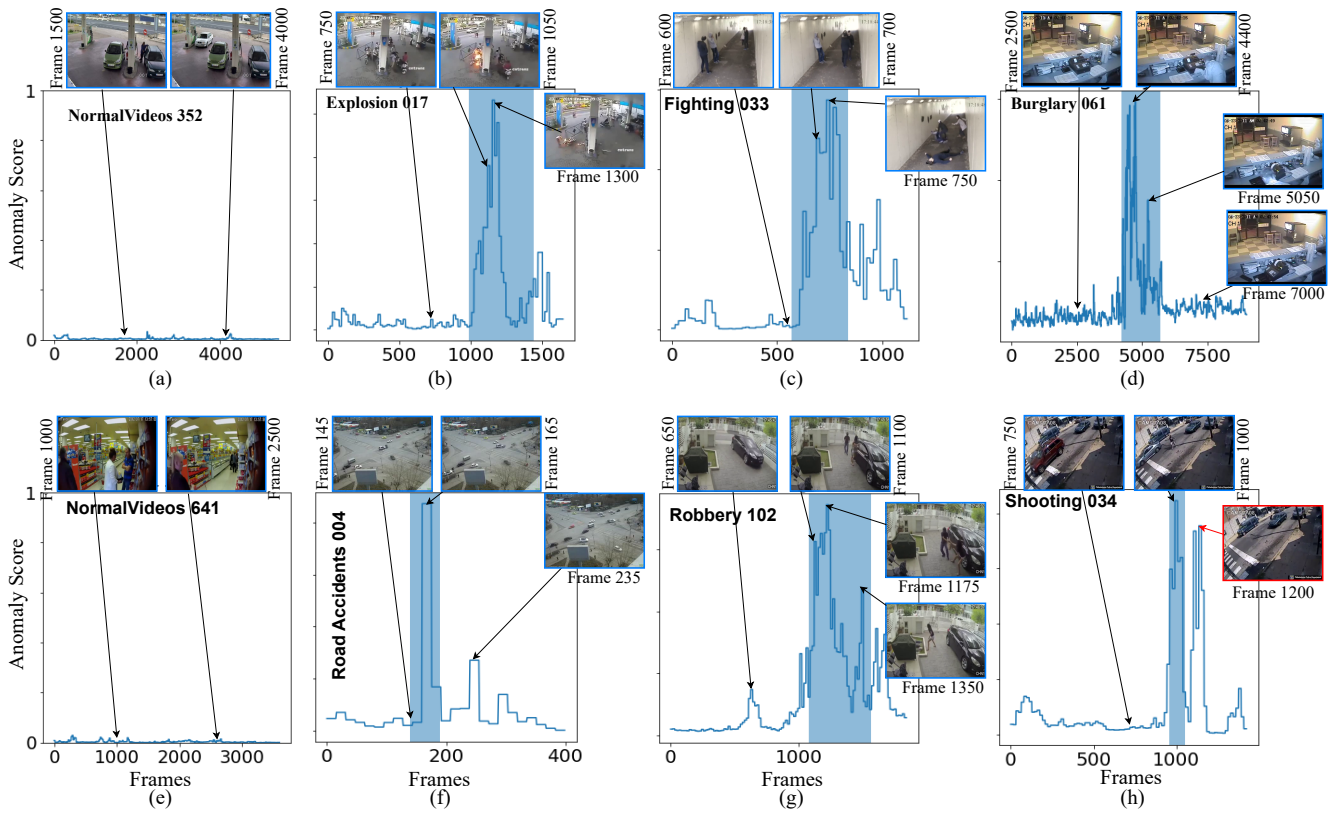
Figure 9. Anomaly scores of the proposed C2FPL framework on different videos from the UCF-Crime Dataset.
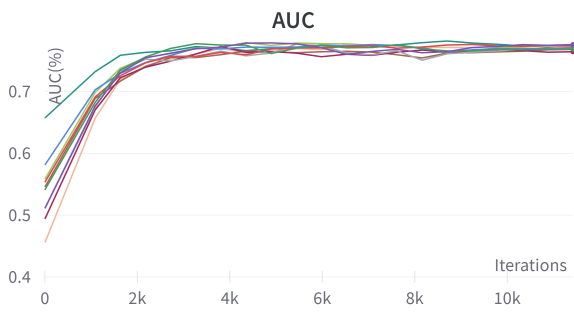

Figure 10. Convergence of our proposed model using multiple random seed experiments.