ExploitingCompletenessandUncertaintyofPseudoLabels
forWeaklySupervisedVideoAnomalyDetection

ChenZhang1,2GuorongLi3∗YuankaiQi4ShuhuiWang5,6
LaiyunQing3QingmingHuang1,2,3,5Ming-HsuanYang7,8
1StateKeyLaboratoryofInformationSecurity,InstituteofInformationEngineering,CAS
2SchoolofCyberSecurity,UniversityofChineseAcademyofSciences
3SchoolofComputerScienceandTechnology,UniversityofChineseAcademyofSciences
4AustralianInstituteforMachineLearning,TheUniversityofAdelaide
5KeyLaboratoryofIntelligentInformationProcessing,InstituteofComputingTechnology,CAS
6PengChengLaboratory,7UniversityofCalifornia,Merced,8YonseiUniversity

zhangchen@iie.ac.cn,liguorong@ucas.ac.cn,qykshr@gmail.com,
wangshuhui@ict.ac.cn,{lyqing,qmhuang}@ucas.ac.cn,mhyang@ucmerced.edu

Abstract

1. Introduction
Automaticallydetectingabnormaleventsinvideoshasattractedincreasingattentionforitsbroadapplicationsinintelligentsurveillance systems. Since abnormal events are sparse in videos, recent studies are mainly developed within the weakly supervised learning framework [5, 12, 19, 25, 27, 29, 32, 34, 37–41], where only video-level annotations are available. However, the goal of anomaly detection is to predict frame-level anomaly scores during test. This results in great challenges for weakly supervised video anomaly detection.

Existingmethodsbroadlyfallintotwocategories:onestage methods based on Multiple Instance Learning (MIL) and two-stage self-training methods. One-stage MIL-based
methods [19, 27, 29, 39, 41] treat each normal and abnormal video as a negative and positive bag respectively, and clips of a video are the instances of a bag. Formulating anomaly detection as a regression problem, these methods adopt ranking loss to encourage the highest anomaly score in a positive bag to be higher than that in a negative bag. Due to the lack of clip-level annotations, the anomaly scores generated by MIL-based methods are usually less accurate. To alleviate this problem, two-stage self-training methods are proposed [5, 12]. In the first stage, pseudo labels for clips are generated by MIL-based methods. In the second stage, MIST [5] utilizes these pseudo labels to refine discriminative representations. In contrast, MSL [12] refines the pseudo labels via a transformer-based network. Despite progress, existing methods still suffer two limitations. First, the ranking loss used in the pseudo label generator ignores the completeness of abnormal events. The reason is that a positive bag may contain multiple abnormal clips as shown in Figure 1, but MIL is designed to detect only the most likely one. The second limitation is that the uncertainty of generated pseudo labels is not taken into account in the second stage. As the pseudo labels are usually noisy, directly using them to train the final classifier may hamper its performance.

To address these problems, we propose to enhance pseudo labels via exploiting completeness and uncertainty properties. Specifically, to encourage the complete detection of abnormal events, we propose a multi-head module to generate pseudo labels (each head serves as a classifier) and introduce a diversity loss to ensure the distribution difference of pseudo labels generated by the multiple classification heads. In this way, each head tends to discover a different abnormal event, and thus the pseudo label generator covers as many abnormal events as possible. Then, instead of directly training a final classifier with all pseudo labels, we design an iterative uncertainty-based training strategy. We measure the uncertainty using Monte Carlo (MC) Dropout [6] and only clips with lower uncertainty are used to train the final classifier. At the first iteration, we use such uncertainty to refine pseudo labels obtained in the first stage, and in the remaining iterations, we use it to refine the output of the desired final classifier.

The main contributions of this paper are as follows:

- We design a multi-head classifier scheme together with a diversity loss to encourage the pseudo labels to cover as many abnormal clips as possible.
- We design an iterative uncertainty aware self-training strategy to gradually improve the quality of pseudo labels.
- Experiments on UCF-Crime, TAD, and XD-Violence datasets demonstrate the favorable performance compared to several state-of-the-art methods.

2. Related Work

Semi-Supervised Methods. In the semi-supervised setting, only normal videos are required in the training set. Semi-supervised video anomaly detection methods can be divided into one-class classifier-based methods [24, 33, 36] and reconstruction-based methods [1, 3, 4, 9, 16, 21, 22]. In one-class classifier-based methods, the model constructs a boundary that distinguishes normal events from abnormal events by learning information about normal videos. Xu et al. [36] adopt an autoencoder to learn appearance and motion features as well as their associations, and then use multiple one-classifiers to predict anomaly scores based on these three learned feature representations. Sabokrou et al. [24] propose to train an end-to-end one-classification model in an adversarial manner. For the problem of anomaly detection in complex scenarios, Wu et al. [33] propose to jointly optimize representation learning and one-classification using convolutional neural networks. Reconstruction-based methods aim to minimize the reconstruction error of the training data, and take the minimum error as the threshold for discriminating abnormality. Some [4, 22] learn the dictionary of normal videos, and video clips that cannot be represented by the dictionary are determined to be abnormal. The others [3, 9] learn the rules of normal video sequences through autoencoders, and events that produce higher reconstruction errors are judged as anomalies. To prevent the model from over-fitting normal videos, later works [1, 16, 21] introduce a memory module for recording normal patterns.

Weakly Supervised Methods. Different from the semi-supervised setting, there are both normal and abnormal videos in the training set for weakly supervised methods, but frame-level annotations are not available. Most of the weakly supervised anomaly detection methods are one-stage MIL-based methods. In [27], the first MIL-based method with ranking loss is proposed, along with a large-scale video anomaly detection dataset. Later, Zhang et al. [39] propose an inner bag loss, which is complementary to the outer bag ranking loss. To learn a motion-aware feature that can better detect anomalies, Zhu et al. [41] use an attention module to take temporal context into the multi-instance ranking model. Tian et al. [29] develop a robust top-k MIL method by training a temporal feature magnitude learning function. To effectively utilize the temporal context, Lv et al. [19] propose a high-order context encoding model. Zhang et al. [38] aim to address the open event issue existing in anomaly detection tasks by using meta-learning. Sapkota et al. [25] construct a submodularity diversified MIL loss in a Bayesian non-parametric way, which can satisfy anomaly detection in more realistic settings with outliers and multimodal scenarios.

Recently, two-stage self-training methods are proposed to generate more accurate and fine-grained anomaly scores.
which adopt a two-stage pipeline, i.e., pseudo labels are generated first and then fed into a classification module. Feng et al. [5] propose to use the information provided by the multi-instance pseudo label generator to fine-tune the feature encoder to generate more discriminative features specifically for the task of video anomaly detection. Li et al. [40] reweight the weakly supervised video anomaly detection. Zhong et al. [34] propose an audio-visual dataset and design a holistic and localized framework to explicitly model relations of video snippets to learn discriminative feature representations. In [32], Wu et al. further explore the importance of temporal relation and discriminative features for weakly supervised anomaly detection.

Self-Training. Self-training is one of the mainstream techniques in semi-supervised learning [11, 23, 30] and has recently shown important progress for tasks like classification [7, 20, 35] and domain adaptation [14, 42]. For self-training, the training data consists of a small amount of labeled data and a large amount of unlabeled data. The core idea is to use the model trained with labeled data to generate pseudo labels of unlabeled data, and then train the model jointly with labeled data and pseudo labels. The training process is repeated until the model converges. In weakly supervised video anomaly detection, Feng et al. [5] propose a self-training framework in which clip-level pseudo labels generated by a multiple instance pseudo label generator are assigned to all clips of abnormal videos to refine feature encoder. Most similar to our work is the Multi-Sequence Learning method proposed by Li et al. [12], which refines anomaly scores by gradually reducing the length of selected sequences by self-training. However, the self-training mechanisms used by these methods do not consider the uncertainty of pseudo labels, leading to a gradually deviating self-training process guided by noisy pseudo labels. In contrast, we develop an uncertainty aware self-training strategy that can reduce the effect of unreliable pseudo labels. We also consider the importance of temporal context for video understanding, in an effort to better refine anomaly scores.

3. Method

The proposed pseudo label enhancement framework based on completeness and uncertainty is shown in Figure 2. We first use a multi-head classifier trained with a diversity loss and MIL ranking loss to generate initial clip-level pseudo labels. Then, we utilize an iterative uncertainty aware pseudo label refinement strategy to gradually improve the quality of pseudo labels to train the final desired classifier. In the following, we first formulate the task of weakly supervised video anomaly detection and then elaborate each component of our method.

3.1. Notations and Preliminaries

Assume that we are given a set of $N$ videos $\mathcal{V} = \{V_i\}_{i=1}^N$ and the ground-truth labels $\mathcal{Y} = \{Y_i\}_{i=1}^N \in \{0, 1\}$. $\mathcal{Y}_i = 1$
if an abnormal clip is present in the video and \( Y_i = 0 \) otherwise. During training, only video-level labels are available. However, in the testing stage, the goal of this task is to generate frame-level anomaly scores to indicate the temporal location of abnormal events. Following previous MIL-based methods [19, 27, 32, 39, 41], for each input video \( V \), we first divide it into 16-frame non-overlapping \( T_i \) clips and use a pre-trained 3D convolutional network to extract features, forming a clip feature sequence \( C_i = \{ c_{i,1}, c_{i,2}, \ldots, c_{i,T_i} \} \in \mathbb{R}^{T_i \times D_V} \), where \( T_i \) is the number of extracted video clip features and \( D_V \) is the feature dimension. Since long untrimmed videos may contain different numbers of clips, which is inconvenient for training. Therefore, consistent with [27, 29], the video clip features are combined into \( S \) temporal video segments denoted as \( X_i = \{ x_{i,1}, x_{i,2}, \ldots, x_{i,S} \} \in \mathbb{R}^{S \times D_V} \) by averaging multiple consecutive clip features. We treat an abnormal video \( V \) as a positive bag and a normal video \( V \) as a negative bag, and treat each segment \( x_{i,s}^a \) or \( x_{i,s}^n \) as an instance in the bag.

3.2. Completeness of Pseudo Labels

Inspired by [13], which uses a diversity loss to model action completeness, we design a completeness enhanced pseudo label generator composed of parallel multi-head classifier, together with a diversity loss to detect as many abnormal events as possible in a video. Each head \( f_g (\cdot; \phi^k) \) is composed of three fully connected layers parameterized by \( \phi^k \). Taking the video segment features \( X_i = \{ x_{i,s} \}_{s=1}^S \) as input, each head outputs the anomaly scores of each segment, which are further passed through a softmax to generate a score distribution:

\[
\hat{Y}_i^k = \text{softmax} \left( f_g \left( X_i; \phi^k \right) \right)
\]  

where \( \hat{Y}_i^k \in \mathbb{R}^{S \times 1} \) denotes the score distribution of the \( i \)-th video from the \( k \)-th head. The predicted score distributions of \( K \) heads are then enforced to be distinct from each other by a diversity loss, which minimizes the cosine similarity of the distribution between any two heads:

\[
\mathcal{L}_{\text{diver}} = \frac{1}{Z} \sum_{k=1}^{K-1} \sum_{q=k+1}^{K} \left\| \hat{Y}_i^k \cdot \hat{Y}_i^q \right\|_2 \tag{2}
\]

where \( Z = \frac{1}{2} K (K - 1) \). A regularization term on the norm of the segment score sequences is used to balance multiple heads and to avoid performance degradation due to dominance by one head:

\[
\mathcal{L}_{\text{norm}} = \frac{1}{K} \sum_{k=1}^{K} \left\| A^k \right\| - \left\| A^{avg} \right\| \tag{3}
\]

where \( A^k = f_g \left( X_i; \phi^k \right) \) denotes the anomaly scores generated by the \( k \)-th head and \( A^{avg} \) is the average of the anomaly scores produced by each head: \( A^{avg} = \frac{1}{K} \sum_{k=1}^{K} f_g \left( X_i; \phi^k \right) \).

Under the action of the diversity loss and norm regularization, the anomaly scores generated by multiple heads can achieve maximum differentiation and detect different anomalous segments. Finally, \( A^{avg} \) is followed by a sigmoid function to obtain the predicted segment-level labels ranging from 0 to 1:

\[
\hat{Y}_i = \text{sigmoid} \left( A^{avg} \right). \tag{4}
\]

where \( \hat{Y}_i = \{ \hat{y}_{i,1}, \hat{y}_{i,2}, \ldots, \hat{y}_{i,S} \} \) represents the predicted segment-level labels of the \( i \)-th video. For the abnormal video \( V_i^a \), the predicted labels are denoted as \( \hat{Y}_i^a = \{ \hat{y}_{i,1}^a, \hat{y}_{i,2}^a, \ldots, \hat{y}_{i,S}^a \} \). For the normal video \( V_i^n \), the predicted labels are denoted as \( \hat{Y}_i^n = \{ \hat{y}_{i,1}^n, \hat{y}_{i,2}^n, \ldots, \hat{y}_{i,S}^n \} \).

Like [27], the ranking loss is used to constrain the highest anomaly score of abnormal videos to be higher than that of normal videos:

\[
\max_{\hat{y}_{i,s}^a \in \hat{Y}_i^a} \hat{y}_{i,s}^a > \max_{\hat{y}_{i,s}^n \in \hat{Y}_i^n} \hat{y}_{i,s}^n \tag{5}
\]

To maximize the separability between positive and negative instances, a hinge-based ranking loss is used:

\[
\mathcal{L}_{\text{MIL}} = \max \left( 0, 1 - \max_{\hat{y}_{i,s}^a \in \hat{Y}_i^a} \hat{y}_{i,s}^a + \max_{\hat{y}_{i,s}^n \in \hat{Y}_i^n} \hat{y}_{i,s}^n \right) \tag{6}
\]

Finally, the completeness enhanced pseudo label generator is trained with loss as follows:

\[
\mathcal{L}_f = \mathcal{L}_{\text{MIL}} + \alpha \mathcal{L}_{\text{diver}} + \alpha \mathcal{L}_{\text{norm}} \tag{7}
\]

where \( \alpha \) is the hyper-parameter to balance the losses.

3.3. Uncertainty of Pseudo Labels

Instead of directly using the clip-level pseudo labels obtained in the first stage (Sec. 3.2) to train the final desired clip classifier \( f_c \), we propose an uncertainty aware self-training strategy to mine clips with reliable pseudo labels. Specifically, we introduce the uncertainty estimation leveraging Monte Carlo (MC) Dropout [6] so that clips with low uncertainty (i.e., reliable) pseudo labels are selected for training \( f_c \). This process is conducted several iterations. Throughout these iterations, the pseudo labels are continuously refined, eventually generating high-quality fine-grained pseudo labels to train the final desired clip classifier \( f_c \). Note that the pseudo labels are initially obtained in the first stage and are then updated by \( f_c \).

Uncertainty Estimation. We use MC-Dropout [6] to estimate the uncertainty of clip-level pseudo labels. For training video clips \( C_i = \{ c_{i,t} \}_{t=1}^{T_i} \), we perform \( M \) stochastic forward passes through the model \( f \) trained with dropout. In the first iteration, we use the multi-head classifier in the
first stage as the trained model (i.e., \( f = f_g \)). In the remaining iterations, \( f = f_c \). Each pass generates clip-level pseudo labels as follows:

\[
\hat{Y}_i = \text{sigmoid} \left( f(c_i; \tilde{W}^m) \right) \quad (8)
\]

where \( \tilde{W}^m \) denotes the \( m^{th} \) sampled masked model parameters and \( \hat{Y}_i = \{ \hat{y}_{i,t} \}_{t=1}^{T_i} \). The clip-level pseudo labels \( \hat{Y}_i = \{ \hat{y}_{i,t} \}_{t=1}^{T_i} \) used as the supervision for training the clip classifier are given by the predictive mean:

\[
E(\hat{Y}_i) = \frac{1}{M} \sum_{m=1}^{M} (\hat{Y}_i^m) \quad (9)
\]

The prediction uncertainties \( U_i = \{ u_{i,t} \}_{t=1}^{T_i} \) of \( \hat{Y}_i \) are given by the diagonal entries (i.e., the variance of elements in \( \hat{Y}_i \)) in the covariance matrix \( \text{Cov}(\hat{Y}_i) \):

\[
\text{Cov}(\hat{Y}_i) = \frac{1}{M} \sum_{m=1}^{M} \hat{Y}_i^m \hat{Y}_i^\top - E(\hat{Y}_i) E(\hat{Y}_i) \quad (10)
\]

### Iterative Reliable Clip Mining

As the goal is to train a reliable model with low-uncertainty pseudo labels, we mine reliable clips after uncertainty estimation. For each video, we rank the uncertainties \( U_i \) of its pseudo labels from small to large and remain the clips with the least \( R\% \) uncertainty scores, where \( R\% \) represents the sample ratio. In this way, reliable video clips and corresponding clip-level pseudo labels can be mined.

Since participating in training with only clip-level features ignores the contextual relationship of the video, we use a long-term feature memory [31] to model the temporal relationship between video clips. The clip features of each video are stored in a memory pool. After obtaining the selected reliable clip, we retrieve the window size \( w \) clip features \( \mathcal{H}_{i,t} = [c_{i,t-w}, \ldots, c_{i,t-1}] \) before the current clip \( c_{i,t} \) from the memory pool. We obtain \( \mathcal{H}_{i,t} \) by performing mean pooling on \( \mathcal{H}_{i,t} \), and then concatenate the current clip feature \( c_{i,t} \) with \( \mathcal{H}_{i,t} \) into a new temporal clip feature \( \mathcal{T}_{i,t} \). Thus we can use all reliable temporal features set \( \Omega_R(\mathcal{C}) \) and reliable clip-level pseudo labels \( \Omega_R(\hat{Y}) \) to train the clip classifier \( f_c \) based on the binary cross entropy loss:

\[
\mathcal{L}_{XE} = \sum_{\mathcal{T}_{i,t} \in \Omega_R(\mathcal{C})} - (\hat{y}_{i,t} \log (f_c(\mathcal{T}_{i,t}))) + (1 - \hat{y}_{i,t}) \log (1 - f_c(\mathcal{T}_{i,t})) \quad (11)
\]

where \( \hat{y}_{i,t} \in \Omega_R(\hat{Y}) \) represents the pseudo label of the \( t_{ih} \) clip in the \( i_{th} \) video. Then we can obtain the clip classifier trained in the current iteration, and perform uncertainty estimation and reliable sample selection in the next iteration to further train the desired clip classifier.

### Algorithm 1 Completeness-and-Uncertainty Aware Pseudo Label Enhancement

**Input:** A set of \( N \) videos \( \mathcal{V} = \{ V_i \}_{i=1}^{N} \) and video-level labels \( \mathcal{Y} = \{ Y_i \}_{i=1}^{N} \).

**Output:** Clip classifier \( f_c \).

1: // Completeness Enhanced Pseudo Label Generator.
2: Extract \( T_i \) clip features for each video \( V_i \) as \( \{ c_{i,t} \}_{t=1}^{T_i} \).
3: Combine \( \{ c_{i,t} \}_{t=1}^{T_i} \) into \( S \) segment features \( \{ x_{i,s} \}_{s=1}^{S} \).
4: Training multi-head classifier \( f_g \) with \( \{ x_{i,s} \}_{s=1}^{S} \) and \( Y_i \) via Eq. 7.
5: Obtain the trained model \( f = f_g \).
6: // Uncertainty Aware Pseudo Label Refinement.
7: while not converged do
8: Leverage MC-Dropout for the trained model \( f \).
9: Get pseudo labels via Eq. 9, \( \hat{Y}_i = E(\hat{Y}_i) \).
10: Get uncertainties via Eq. 10, \( U_i = \text{diag} (\text{Cov}(\hat{Y}_i)) \).
11: \( (U_i)_{\text{sorted}} \leftarrow \text{SORT}(U_i) \) // asc sorting
12: Remain clips with the least \( R\% \) uncertainty scores.
13: Model temporal relationship for the remained clips.
14: All reliable clip set \( \Omega_R(\mathcal{C}) = \Omega_R(\{ c_{i,t} \}_{i=1}^{N}) \).
15: All reliable label set \( \Omega_R(\hat{Y}) = \Omega_R(\{ \hat{y}_{i,t} \}_{i=1}^{N}) \).
16: Training \( f_c \) with \( \Omega_R(\mathcal{C}) \) and \( \Omega_R(\hat{Y}) \) via Eq. 11.
17: Obtain the trained model \( f = f_c \).
18: end while
19: Return clip classifier \( f_c \) for inference.

### 3.4. Model Training and Inference

**Training.** We increase completeness of pseudo labels with Eq. 7 to cover as many abnormal clips as possible in the first stage. In the second stage, we mine reliable video clips with uncertainty estimation to train the clip classifier using Eq. 11 and gradually refine clip-level pseudo labels through multiple iterations. Algorithm 1 shows the main steps of the training process.

**Inference.** Given a test video, we directly utilize the clip classifier \( f_c \) to predict anomaly scores.

### 4. Experimental Results

We perform experiments on three publicly available datasets including UCF-Crime [27], TAD [19] and XD-Violence [34].

#### 4.1. Datasets and Evaluation Metrics

**Datasets.** UCF-Crime is a large-scale benchmark dataset for video anomaly detection with 13 anomaly categories. The videos are captured from diverse scenes, such as streets, family rooms, and shopping malls. The dataset contains 1610 training videos annotated with video-level labels and 290 test videos with frame-level annotation. TAD is a re-
cently released dataset in traffic scenarios. It covers 7 real-world anomalies in 400 training videos and 100 test videos. In line with the weak supervision setting in [27], the training set is annotated with video-level labels, and the test set provides frame-level labels. XD-Violence is the largest dataset currently used for weakly supervised anomaly detection. Its videos are collected through multiple channels, such as movies, games, and car cameras. It contains 3954 training videos with video-level labels and 800 test videos with frame-level labels, covering 6 anomaly categories. Moreover, it provides audio-visual signals, enabling anomaly detection by leveraging multimodal cues.

**Evaluation Metrics.** Similar to the previous works [5, 27, 39, 40], we choose the area under the curve (AUC) of the frame-level receiver operating characteristic curve to evaluate the performance of our proposed method on UCF-Crime and TAD datasets. For XD-Violence, following [12, 29, 32, 34], we use average precision (AP) as the metric.

### 4.2. Implementation Details

**Feature Extractor.** Consistent with existing methods [29], we use pre-trained I3D [2] model to extract clip features from 16 consecutive frames. Then we divide each video into 32 segments as the input to the multi-head classifier, i.e., $S = 32$. For the XD-violence dataset, following the setting in [32, 34], we extracted audio features by leveraging the VGGish network [8].

**Training Details.** The head number $K$ of the multi-head classifier is 2 for all datasets. Each head consists of three fully connected (FC) layers with 512, 128, and 1 units, respectively. The first and third fully connected layers are followed by ReLU activation and sigmoid activation, respectively. We use a dropout function between FC layers with a dropout rate of 0.6. The classifier is trained using the Adadelta optimizer with a learning rate of 0.1. The parameter $\alpha$ is set to 10 for UCF-Crime, and 0.1 for TAD and XD-Violence. The number of stochastic forward passes $M$ is set to 50 for all datasets when generating the initial pseudo labels and uncertainty scores. During pseudo label refinement, the clip classifier consists of three FC layers and 80% of the dropout regulation is used between FC layers. It is trained using the Adam optimizer with a learning rate of $1e^{-4}$ and a weight decay of $5e^{-4}$. At the end of each iteration, we set $M = 5$ to obtain pseudo labels and uncertainty scores.

### 4.3. Comparisons with Prior Work

**Results on UCF-Crime.** Table 1 summarizes the performance comparison between the proposed method and other methods on the UCF-Crime test set. From the results, we can see that our proposed method outperforms all the previous semi-supervised methods [9, 10, 17, 28] and weakly supervised methods [5, 12, 19, 25, 27, 29, 32, 34, 37, 39–41]. Using the same I3D-RGB features, an absolute gain of 0.92% is achieved in terms of the AUC when compared to the best previous two-stage method [12].

**Results on TAD.** The performance comparisons on TAD are shown in Table 2. Compared with the previous semi-supervised approaches [15, 18] and weakly supervised methods [5, 19, 27, 29, 41], our method achieves the superior performance on AUC. Remarkably, with the same 3D features, our method is 2.40% and 2.02% better than two-stage MIST [5] and one-stage RTFM [29].

**Results on XD-Violence.** As shown in Table 3, our method also achieves favorable performance when compared with previous semi-supervised approaches [9, 26] and weakly su-
Effectiveness of Completeness of Pseudo Labels. To investigate the effect of exploiting the completeness property, we conduct experiments (the 3rd row) with only iterative uncertainty aware pseudo label refinement strategy added to the baseline. From the results, we can see that uncertainty property can bring a performance gain of 1.83% in AUC, which indicates that using an uncertainty aware self-training strategy to gradually improve the quality of pseudo labels is important.

Analysis on the Uncertainty Aware Self-Training. To look deeper into the proposed uncertainty aware self-training strategy, we also make a comparison with standard self-training mechanism that randomly selects samples from the new labeled set. As shown in Figure 3, with the number of iterations increasing, the performance of the uncertainty aware self-training strategy increases rapidly and then stabilizes gradually, whereas the performance improvement of the standard self-training strategy is much smaller. This shows that adopting the uncertainty aware self-training strategy to select reliable samples in each iteration is more beneficial for training the desired classifier.

4.5. Hyperparameter Analysis

Effect of the head number $K$. In Table 5, we show the comparative result of varying the head number $K$ of the multi-head classifier on the UCF-Crime dataset. When $K = 1$, the pseudo label generator is a single-head classifier, which limits the ability of pseudo labels to cover complete anomalies. As the head number increases from 2 to 5, the performance of multi-head classifier models all exceed that of single-head classifier model and our method yields the best performance when $K = 2$. Since the performance difference between models with head numbers from 2 to 5 is not so significant, we set $K = 2$ for all datasets.

Table 5. Performance comparison of different head numbers.

<table>
<thead>
<tr>
<th>K</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC (%)</td>
<td>84.69</td>
<td>86.22</td>
<td>85.05</td>
<td>85.90</td>
<td>85.10</td>
</tr>
</tbody>
</table>

Table 3. Comparison with other methods on XD-Violence.

<table>
<thead>
<tr>
<th>Supervised</th>
<th>Methods</th>
<th>Feature</th>
<th>AP(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semi</td>
<td>OCSVM [26]</td>
<td>I3D+VGGish</td>
<td>72.25</td>
</tr>
<tr>
<td></td>
<td>Hasan et al. [9]</td>
<td>I3D+VGGish</td>
<td>30.77</td>
</tr>
<tr>
<td></td>
<td>SVM baseline</td>
<td>I3D+VGGish</td>
<td>50.78</td>
</tr>
<tr>
<td></td>
<td>Sultani et al. [27]</td>
<td>I3D+VGGish</td>
<td>73.20</td>
</tr>
<tr>
<td></td>
<td>Wu et al. [34]</td>
<td>I3D</td>
<td>75.41</td>
</tr>
<tr>
<td></td>
<td>Wu et al. [34]</td>
<td>I3D+VGGish</td>
<td>78.64</td>
</tr>
<tr>
<td>Weakly</td>
<td>Wu et al. [32]</td>
<td>I3D+VGGish</td>
<td>75.90</td>
</tr>
<tr>
<td></td>
<td>RTFM [29]</td>
<td>I3D</td>
<td>77.81</td>
</tr>
<tr>
<td></td>
<td>MSL [12]</td>
<td>I3D</td>
<td>78.28</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>I3D</td>
<td>78.74</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>I3D+VGGish</td>
<td>81.43</td>
</tr>
</tbody>
</table>

Table 4. Ablation results on UCF-Crime.

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Completeness</th>
<th>Uncertainty</th>
<th>AUC(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>√</td>
<td></td>
<td></td>
<td>82.86</td>
</tr>
<tr>
<td>√</td>
<td>√</td>
<td></td>
<td>84.89</td>
</tr>
<tr>
<td>√</td>
<td></td>
<td>√</td>
<td>84.69</td>
</tr>
<tr>
<td>√</td>
<td>√</td>
<td>√</td>
<td>86.22</td>
</tr>
</tbody>
</table>

Effect of the head number $K$. In Table 5, we show the comparative result of varying the head number $K$ of the multi-head classifier on the UCF-Crime dataset. When $K = 1$, the pseudo label generator is a single-head classifier, which limits the ability of pseudo labels to cover complete anomalies. As the head number increases from 2 to 5, the performance of multi-head classifier models all exceed that of single-head classifier model and our method yields the best performance when $K = 2$. Since the performance difference between models with head numbers from 2 to 5 is not so significant, we set $K = 2$ for all datasets.

Table 5. Performance comparison of different head numbers.

<table>
<thead>
<tr>
<th>K</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC (%)</td>
<td>84.69</td>
<td>86.22</td>
<td>85.05</td>
<td>85.90</td>
<td>85.10</td>
</tr>
</tbody>
</table>

Effectiveness of Uncertainty of Pseudo Labels. To investigate the effect of exploiting the uncertainty property, we conduct experiments (the 3rd row) with only iterative uncertainty aware pseudo label refinement strategy added to the baseline. From the results, we can see that uncertainty property can bring a performance gain of 1.83% in AUC, which indicates that using an uncertainty aware self-training strategy to gradually improve the quality of pseudo labels is important.
Effect of the diversity loss weight $\alpha$. To explore the influence of the diversity loss weight, we conduct experiments on the UCF-Crime dataset and report the AUC with different diversity loss weights. As shown in Table 6, we report the results as $\alpha$ increases from 0.01 to 100. The AUC can be consistently improved as $\alpha$ grows from 0.01 to 10 and then decreases when $\alpha$ increases to 100, which means 10 is sufficient to model the completeness of pseudo labels on the UCF-Crime dataset. For the TAD and XD-Violence datasets, the weight of the diversity loss $\alpha$ is set to 0.1.

Effect of self-training iterations $j$. In Figure 3, we report the AUC of our proposed method with different iterations on the UCF-Crime dataset. In general, we observe the performance to improve rapidly initially as $j$ increases, and gradually converge in $3 - 5$ iterations. Considering efficiency and performance comprehensively, we set $j = 3$ for UCF-Crime and XD-Violence datasets. Due to the relatively small scale of the TAD dataset, we set $j = 5$ for TAD.

Effect of the sample ration $R$. Table 7 reports the experimental results evaluated with different sample ratios on the UCF-Crime dataset. We observe that setting $R$ to 0.5 is an optimum choice. As $R$ increases, the AUC improves rapidly and then declines gradually. According to our analysis, the sampling ratio of reliable samples should depend on the scale of the training set. With a large-scale training set, it is more likely to generate high-quality pseudo labels, so a larger sampling ratio can be set. On the contrary, a smaller sampling ratio should be set. Therefore, for the large-scale XD-Violence dataset, we set $R = 0.7$. For the small-scale TAD dataset, $R$ is set to 0.3.

### Table 6. Performance comparison of different diversity loss weights.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>0.01</th>
<th>0.1</th>
<th>1</th>
<th>10</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC(%)</td>
<td>84.56</td>
<td>84.43</td>
<td>84.82</td>
<td><strong>86.22</strong></td>
<td>75.06</td>
</tr>
</tbody>
</table>

### Table 7. Performance comparison of different sample ratios.

<table>
<thead>
<tr>
<th>$R$</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC(%)</td>
<td>85.13</td>
<td>85.25</td>
<td><strong>86.22</strong></td>
<td>85.58</td>
<td>85.71</td>
</tr>
</tbody>
</table>

### Figure 4. Qualitative results on UCF-Crime (Explosion), TAD (PedestrianOnRoad) and XD-Violence (Fighting, Normal) test videos. The pink square area is the interval where abnormal events occur in the video, and the blue curve indicates the predicted anomaly scores of each video frame.

5. Conclusions

In this paper, we focus on enhancing the quality of pseudo labels and propose a two-stage self-training method that exploits completeness and uncertainty properties. First, to enhance the completeness of pseudo labels, a multi-head classification module constrained by a diversity loss is designed to generate pseudo labels that can cover as many anomalous events as possible. Then, an iterative uncertainty aware self-training strategy is employed to select reliable samples to train the clip classifier. The output of the clip classifier is gradually refined through multiple iterations, resulting in high-quality pseudo labels to train the final desired clip classifier. Extensive experiments show the effectiveness of our method.

### Acknowledgements.

This work was supported in part by the Italy–China Collaboration Project TALENT under Grant 2018YFE0118400; in part by the National Natural Science Foundation of China under Grant 62236008, 61836002, 62272438, 61976069, U21B2038, 61931008, 62022083, and 61872333; and in part by the Fundamental Research Funds for Central Universities.
References


[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*, pages 322–339, 2020. 1, 3, 5, 6, 7


[38] Chen Zhang, Guorong Li, Qianqian Xu, Xinfeng Zhang, Li Su, and Qingming Huang. Weakly supervised anomaly detection in videos considering the openness of events. *IEEE TITS*, 23(11):21687–21699, 2022. 1, 2


