Federated Weakly Supervised Video Anomaly Detection with Multimodal Prompt Supplementary Materials

Benfeng Wang[1], Chao Huang[1*], Jie Wen[2], Wei Wang[1], Yabo Liu[2], Yong Xu[2]

[1]School of Cyber Science and Technology, Shenzhen Campus of Sun Yat-sen University
[2]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen
wangbf23@mail2.sysu.edu.cn, huangch253@mail.sysu.edu.cn, wenjie@hit.edu.cn, wangwei29@mail.sysu.edu.cn, yaboliu.ug@gmail.com, yongxu@ymail.com

Datasets Splits

As mentioned before, in order to simulate real-life application scenarios, we re-organize the two large scale datasets UCF-Crime and XD-Violence following CLAP (Al-Lahham et al. 2024). In this section, we introduce the dataset split strategies in detail. Figure 1 and Figure 2 show the distribution of the split dataset with event and scene based split.

Random Split In this setting, the dataset is split into several equal parts. Each client holds the same number of both normal and abnormal videos.

Event Based Split In this setting, we assume that each client only considers a certain anomaly, which means that different clients may hold different numbers of videos due to the scarcity of abnormal videos. Note that the UCF-Crime and XD-Violence datasets are both equipped with anomaly class labels, so we distribute videos of the same abnormal category to the same client.

In order to ensure that each client has relatively balanced normal and abnormal videos, we distribute normal videos to each client proportionally based on the number of abnormal videos they have. Additionally, given that some videos of XD-Violence dataset may contain multiple anomalies, we choose the dominant anomaly for those videos. Figure 3 shows some example videos of UCF-Crime split based on event.

Scene Based Split In this setting, we consider the most complex situation, where the videos held by the clients are recorded in the same environment such as living room, park and street. This setting is the most challenging because there may be imbalance between normal and abnormal videos. Figure 4 shows some example videos on UCF-Crime split based on scene.

Training and Inference Process

In this section, we describe the training and inference process of the proposed method in detail, as shown in Algorithm 1 and Algorithm 2. For each round r, each client will train the prompt generator and the temporal modeling block, which are represented as θ_i^r.

*Corresponding Author

Algorithm	UCF		XD	
	Event	Scene	Event	Scene
FedAvg	**85.07**	84.03	**74.23**	75.99
FedProx	84.74	**84.86**	73.17	**78.52**
SCAFFOLD	82.69	84.06	71.00	73.36

Table 1: The results on using different federated algorithms.

Experiments on Different Federated Algorithms

In the main body of this paper, we choose FedAVG (McMahan et al. 2017) as federated algorithms. In this section, we conduct experiments on different federated algorithms FedProx (Li et al. 2020) and SCAFFOLD (Karimireddy et al. 2020). The results are shown in Table 1.

On the one hand, the core idea of FedProx is to reduce the degree of deviation between local models and global models when local training by adding an additional normalization term to loss, which can be presented as follows:

$$\mathcal{L} = L + \frac{\mu}{2} \left\| w_i - w_{global} \right\|_2 \tag{1}$$

where L is the original loss, w_i and w_{global} are local and global parameters, respectively, and μ is hyper-parameter. We set $\mu = 0.01$ in our experiments.

On the other hand, SCAFFOLD solves the problem of client-drift. Specifically, SCAFFOLD utilizes a Control Variable c to guide the training direction of local model. The local updating process can be presented as follows:

$$\theta_i' \leftarrow \theta_i - \eta(\nabla \mathcal{L} - c_i + c) \tag{2}$$

where θ_i is the local parameters of client i, η is the learning rate, c_i and c are local and global control variable, respectively.

Visualization

In this section, we visualize more anomaly confidence of the proposed method and FedCoOp on Xd-Violence. As shown

Algorithm 1: Training Process of the Proposed Method

Input: Global rounds R, local training epochs E, client numbers N, local dataset $\mathcal{D}_i = \{(v_j, Y_j)\}_{j=1}^{N_i}$, batch size B, anomaly class label texts l, frozen pre-trained CLIP models $E_{image}(\cdot)$ and $E_{text}(\cdot)$.

Output: The final global parameters θ^R.

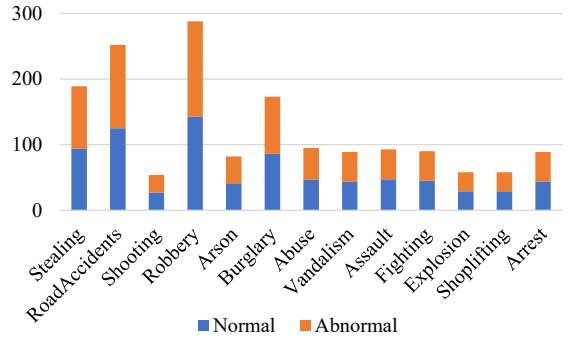
for each round $r \in \{1, ..., R\}$ **do**
  for each client $i \in \{1, ..., N\}$ **do**
    Receives and loads the global parameters θ^r
    for each local epoch $e \in \{1, ..., E\}$ **do**
      for each batch $\{(v_j, Y_j)\}_{j=0}^{B} \in \mathcal{D}_i$ **do**
        Encode video frames: $I \leftarrow E_{image}(v)$;
        Obtain temporal dependencies: $V \leftarrow \varphi(I)$;
        Encode class labels into embedding: $\mathcal{T} \leftarrow E_{text}(l)$;
        Compute average in batch dimension: $\bar{V} \leftarrow Average(V)$;
        Compute local contexts: $Q_{\bar{V}} \leftarrow \bar{V} \times W_Q$;
        Compute global contexts: $K_{\mathcal{T}} \leftarrow \mathcal{T} \times W_K, V_{\mathcal{T}} \leftarrow \mathcal{T} \times W_V$;
        Generate text prompt with cross-attention $\mathcal{P} \leftarrow CrossAttention(Q_{\bar{V}}, K_{\mathcal{T}}, V_{\mathcal{T}})$;
        Tokenize class labels: $tokens \leftarrow Tokenizer(l)$;
        Concatenate the generated prompt and the class label tokens: $t \leftarrow Concat(tokens, \mathcal{P})$;
        Obtain the text features: $T \leftarrow E_{text}(t)$;
        Compute the alignment map: $M \leftarrow V \cdot (T)^T$;
        Compute $S = \{s_1, ...s_K\}$ with top k values of M's each column;
        Compute video level multi-class prediction: $p_i = \frac{exp(s_i/\tau)}{\sum_j exp(s_j/\tau)}$;
        Compute loss with cross entropy: $\mathcal{L}_i = -YlogP$;
        Update local parameters with optimizer: $\theta_i^{r+1} \leftarrow optimizer(\theta_i^r, \mathcal{L}_i)$;
      end for
    end for
  end for
  Aggregate the local parameters: $\theta^{r+1} = \sum_i \frac{|\mathcal{D}_i|}{\sum_j |\mathcal{D}_j|} \theta_i^{r+1}$;
end for
return the final global parameters θ^R

in Figure 5 and Figure 6, the red curves are the results produced by the proposed method while the blue ones are produced by FedCoOp. The gray areas represent the temporal ground-truth of the anomalies in the videos. As we can see, FedCoOp always predicts higher anomaly confidences for normal areas, which is evidently inferior to the proposed method. This gap can be attributed to the text prompts. FedCoOp learns a set of vectors for each client. However, the learned vectors are highly related to local data, and the aggregated vectors may cause poor performance in global test set. In contrast, the proposed method utilize a prompt generator driven by global and local contexts, which is more robust and ensures moderate personalization as well as generalization.

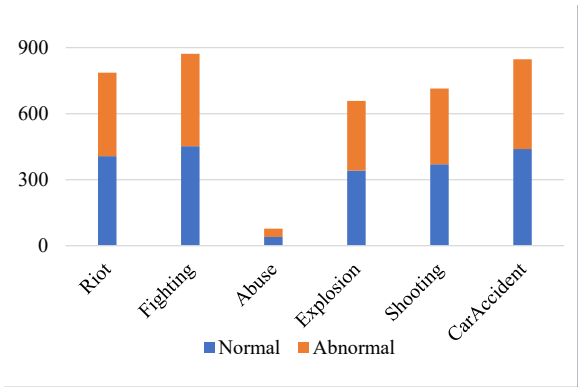
Algorithm 2: Inference Process of the Proposed Method

Input: local test set with frame level labels $\mathcal{D} = \{(v_j, Y_j)\}_{j=1}^{N}$, anomaly class label texts l, frozen pre-trained CLIP models $E_{image}(\cdot)$ and $E_{text}(\cdot)$, pre-trained global parameters θ.
Output: The AUC and AP.

Load pre-trained global parameters θ;
for each data $(v_j, Y_j) \in \mathcal{D}$ **do**
  Encode video frames: $I \leftarrow E_{image}(v)$;
  Obtain temporal dependencies: $V \leftarrow \varphi(I)$;
  Encode class labels into embedding: $\mathcal{T} \leftarrow E_{text}(l)$;
  Compute average in batch dimension: $\bar{V} \leftarrow Average(V)$;
  Compute local contexts: $Q_{\bar{V}} \leftarrow \bar{V} \times W_Q$;
  Compute global contexts: $K_{\mathcal{T}} \leftarrow \mathcal{T} \times W_K$, $V_{\mathcal{T}} \leftarrow \mathcal{T} \times W_V$;
  Generate text prompt with cross-attention $\mathcal{P} \leftarrow CrossAttention(Q_{\bar{V}}, K_{\mathcal{T}}, V_{\mathcal{T}})$;
  Tokenize class labels: $tokens \leftarrow Tokenizer(l)$;
  Concatenate the generated prompt and the class label tokens: $t \leftarrow Concat(tokens, \mathcal{P})$;
  Obtain the text features: $T \leftarrow E_{text}(t)$;
  Compute the alignment map: $M \leftarrow V \cdot (T)^T$;
  Compute binary frame level anomaly confidence: $C \leftarrow 1 - M[:, 0]$;
  Compute AUC and AP: $AUC_j \leftarrow auc_{score}(C, Y_j)$, $AP_j \leftarrow ap_{score}(C, Y_j)$;
end for
Compute average of AUC and AP: $AUC \leftarrow \frac{1}{N}\sum_{j=0}^{N} AUC_j$, $AP \leftarrow \frac{1}{N}\sum_{j=0}^{N} AP_j$;
return the final AUC and AP.

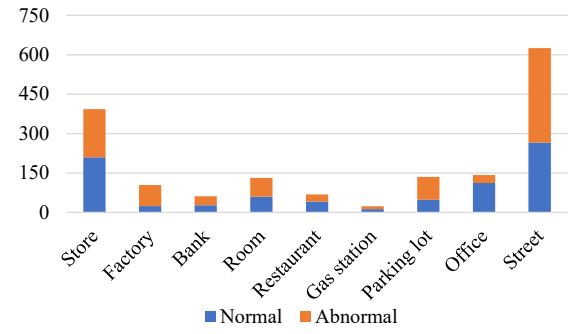


(a) Event based spilt on UCF-Crime

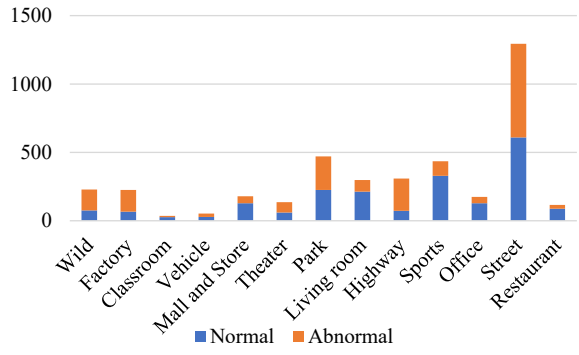


(b) Event based spilt on XD-Violence

Figure 1: The distribution of datasets split based on event.



(a) Scene based spilt on UCF-Crime



(b) Scene based spilt on XD-Violence

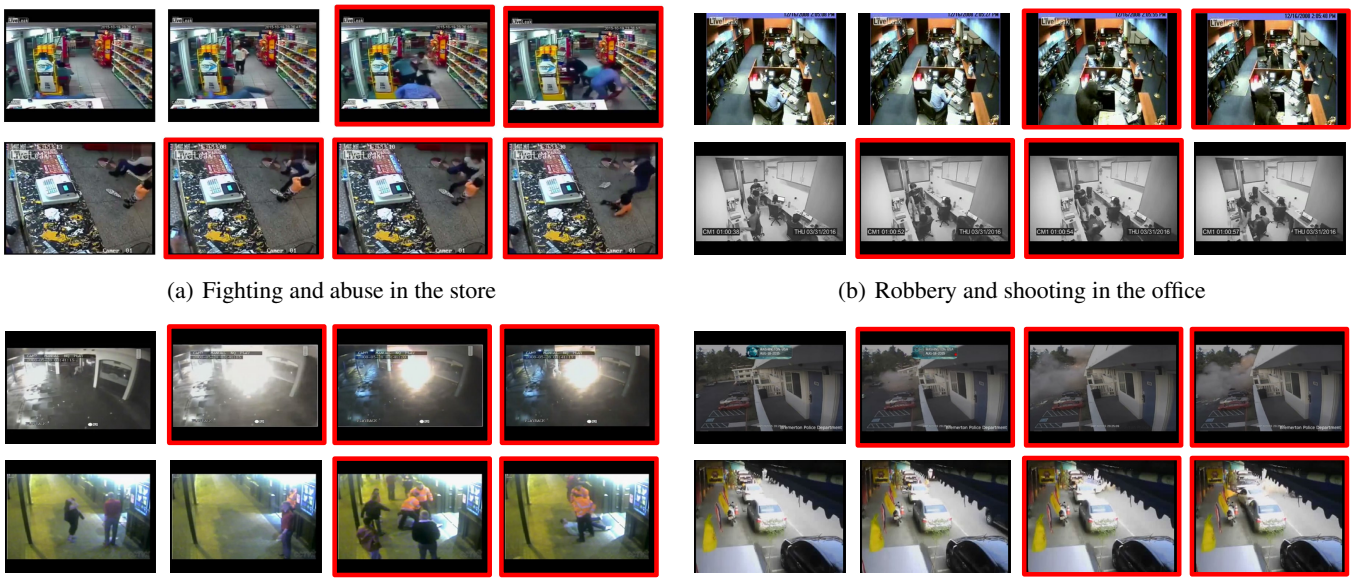
Figure 2: The distribution of datasets split based on scene.

(a) Arson

(b) Fighting

(c) Robbery

Figure 3: Example videos of UCF-Crime split based on event.



(a) Fighting and abuse in the store

(b) Robbery and shooting in the office

(c) Arson and arrest on the street

(d) Explision and accident in the parking lot

Figure 4: Example videos of UCF-Crime split based on scene.

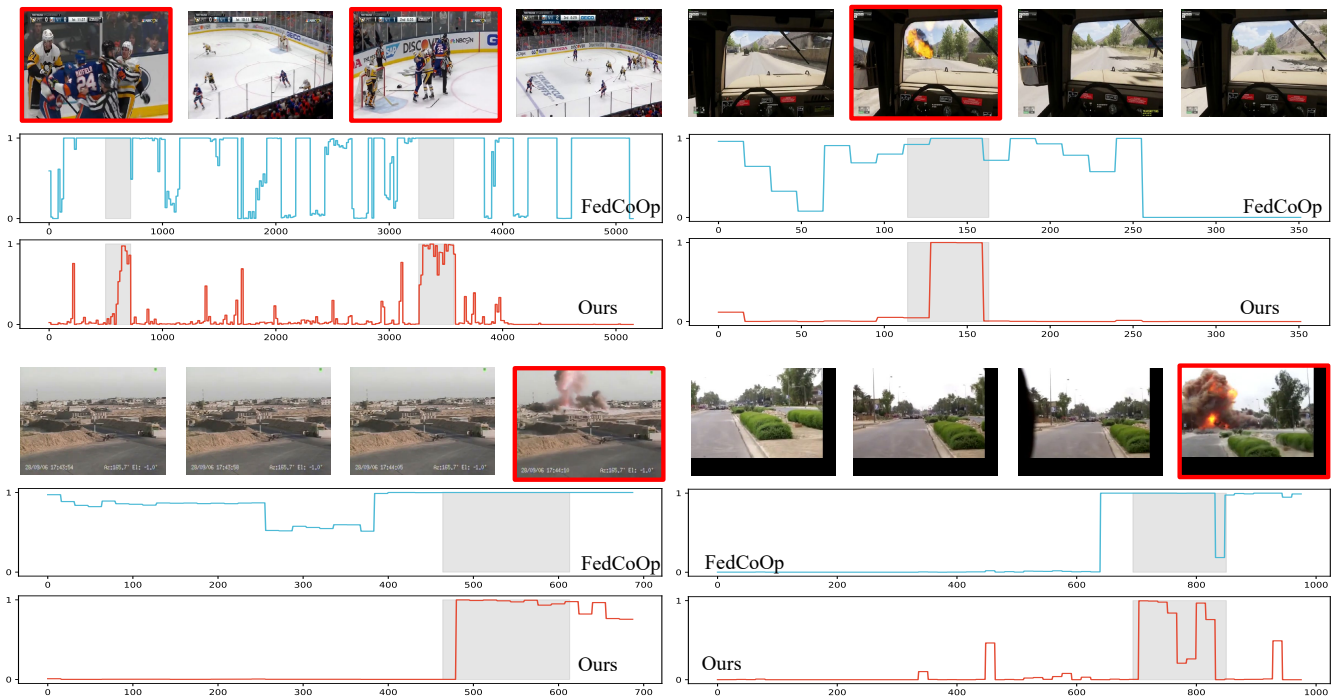


Figure 5: The visualization of comparison with the proposed method and FedCoOp on XD-Violence. The red lines represent the proposed method while the blue ones represent FedCoOp. The gray areas represent the temporal ground-truth of the anomalies in the videos.

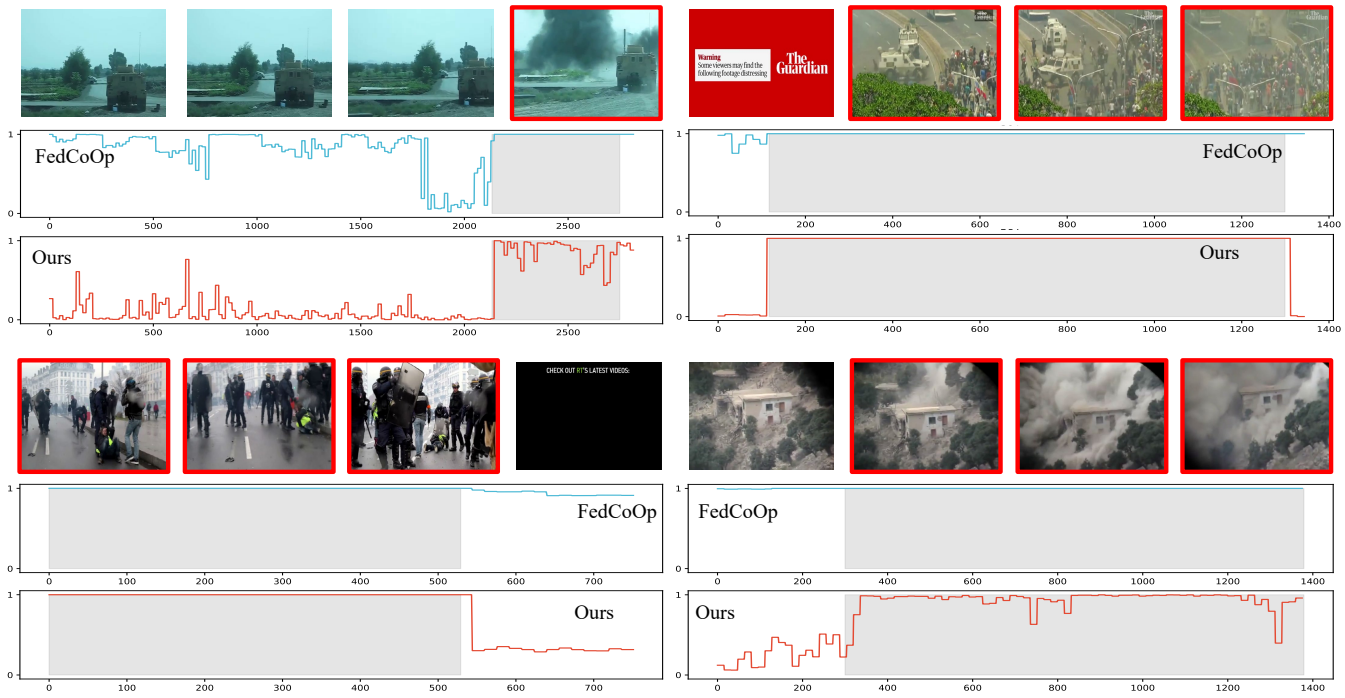


Figure 6: The visualization of comparison with the proposed method and FedCoOp on XD-Violence. The red lines represent the proposed method while the blue ones represent FedCoOp. The gray areas represent the temporal ground-truth of the anomalies in the videos.

References

Al-Lahham, A.; Zaheer, M. Z.; Tastan, N.; and Nandakumar, K. 2024. Collaborative Learning of Anomalies with Privacy (CLAP) for Unsupervised Video Anomaly Detection: A New Baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12416–12425.

Karimireddy, S. P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; and Suresh, A. T. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, 5132–5143. PMLR.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2: 429–450.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282. PMLR.