Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Weakly Supervised Video Anomaly Detection Using CNN-LSTM with Confidence-Aware Automated Pseudo Label Generation

Shima Jalali

shima.jalali67@gmail.com

Shiraz University

Reza Boostani

Shiraz University

Research Article

Keywords: Video Anomaly Detection, Weakly Supervised Learning, Pseudo Labeling, CNN-LSTM, MOG2, YOLOv8, SAM

Posted Date: October 28th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10137759/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Weakly Supervised Video Anomaly Detection Using CNN-LSTM with Confidence-Aware Automated Pseudo Label Generation

Shima Jalali Department of Electrical Engineering
Shiraz University, Shiraz, Iran
Email: shima.jalali67@gmail.com Reza Boostani Department of Electrical Engineering
Shiraz University, Shiraz, Iran
Email: boostani@shirazu.ac.ir

***Abstract*—Video anomaly detection in surveillance environments is a challenging problem due to the lack of large-scale frame-level annotations and the high cost of manual labeling. Traditional fully supervised approaches are not scalable in real-world scenarios, while existing weakly supervised methods often rely on noisy bag-level labels and lack fine-grained supervision.**

In this work, we propose a weakly supervised video anomaly detection framework based on automatically generated pseudo labels. Unlike existing approaches that rely solely on video-level annotations, our method constructs a multi-stage pseudo labeling pipeline that integrates motion-based foreground estimation, object-level detection, and pixel-level segmentation.

Specifically, motion regions are first extracted using MOG2 background subtraction. Then YOLOv8 is applied to detect candidate objects in motion-activated frames, followed by SAM-based segmentation to refine object boundaries at pixel level. These signals are fused to generate clip-level pseudo labels with confidence scores.

To handle noise in automatically generated labels, we introduce a confidence-aware weighting mechanism and a temporal consistency regularization term. Furthermore, a lightweight CNN-LSTM architecture is employed, where ResNet18 is used as a frozen feature extractor to significantly reduce computational cost while maintaining competitive performance.

Experimental evaluation on the UCF-Crime dataset demonstrates that the proposed method achieves an AUC of 0.80 and an F1-score of 0.78 under a reproducible multi-run evaluation protocol. The source code and generated pseudo labels will be made publicly available upon publication.

***Index Terms*—Video Anomaly Detection, Weakly Supervised Learning, Pseudo Labeling, CNN-LSTM, MOG2, YOLOv8, SAM**

I. Introduction

Surveillance video systems have significantly expanded in recent years due to the rapid deployment of CCTV cameras in public and private environments. This growth has created an urgent need for automated systems capable of detecting abnormal events such as fighting, theft, fire, and suspicious activities.

Traditional anomaly detection methods relied on hand-crafted features such as optical flow [10], histogram of oriented gradients (HOG) [11], and trajectory modeling [12]. However, these approaches suffer from poor generalization across different environments and cannot capture complex spatio-temporal patterns.

Deep learning-based methods have shown significant improvements in performance, particularly CNN-based and RNN-based architectures [14], [15]. Despite this progress, several challenges remain:

1) **High Computational Cost:** Many deep models require expensive GPU computation, making real-time deployment difficult.
2) **Label Scarcity:** Frame-level annotations are extremely expensive and time-consuming, requiring expert knowledge.
3) **Noisy Weak Supervision:** Video-level labels introduce ambiguity in learning, as abnormal events occupy only small temporal segments.

To address these challenges, we propose a weakly supervised learning framework based on automatically generated pseudo labels. Our key idea is to replace expensive manual annotations with an automated pipeline that combines motion detection, object detection, and segmentation.

A. Contributions

The main contributions of this paper are:

- A multi-stage automated pseudo label generation pipeline combining MOG2, YOLOv8, and SAM for weakly supervised anomaly detection
- A confidence-aware weak supervision mechanism to reduce the impact of noise in automatically generated labels
- A lightweight CNN-LSTM architecture with frozen ResNet18 backbone, reducing trainable parameters to 0.79M (93.7% reduction)
- A temporal consistency regularization strategy for stable and smooth anomaly predictions across clips
- A reproducible multi-run evaluation protocol with statistical reporting (mean and standard deviation across 5 runs)

Corresponding author: Shima Jalali (shima.jalali67@gmail.com)

II. Related Work

A. Handcrafted Feature-Based Methods

Early approaches to anomaly detection used handcrafted features such as optical flow [10], histogram of oriented gradients (HOG) [11], and trajectory modeling [12]. These methods extract low-level visual cues but fail under complex real-world scenarios with varying lighting and camera angles [13].

B. Weakly Supervised Learning Methods

Sultani et al. [1] introduced a multiple instance learning (MIL) framework for weakly supervised anomaly detection using only video-level labels. This approach treats each video as a bag of instances (clips) with at least one positive instance in abnormal videos. While effective, MIL suffers from noisy supervision because normal clips within abnormal videos receive incorrect positive labels.

Subsequent works have attempted to improve MIL by incorporating attention mechanisms [2] or clustering-based normalcy suppression [3]. Wan et al. [20] proposed a graph convolutional network for anomaly detection. Wu et al. [21] introduced self-attention for temporal modeling. However, these methods still rely on video-level labels and do not generate fine-grained pseudo labels.

C. Deep CNN-LSTM Models

CNN-LSTM architectures are widely used for spatio-temporal modeling in video understanding tasks [16], [17]. The CNN extracts spatial features from individual frames, while the LSTM models temporal dependencies across frames [5], [6]. Morais et al. [18] used skeleton-based features with RNNs. Lu et al. [19] proposed a future frame prediction approach.

More recent advances in deep neural network architectures have explored efficient designs for complex data processing. For instance, Boostani et al. [23] developed an efficient deep spiking neural network for complex EEG signal analysis, demonstrating that optimized deep architectures can achieve high performance with reduced computational cost. This aligns with our motivation to freeze the CNN backbone for parameter efficiency.

Our work freezes the CNN backbone to achieve a 93.7% reduction in trainable parameters while maintaining competitive performance, inspired by recent trends in efficient deep learning [23].

D. Pseudo Labeling Methods

Pseudo labeling has been widely used in semi-supervised learning, where a model generates labels for unlabeled data [22]. In anomaly detection, pseudo labeling has been explored to generate frame-level labels from video-level annotations. However, most existing methods do not integrate multi-modal visual cues such as motion, detection, and segmentation simultaneously.

Recent work on multimodal deep learning [26] has demonstrated the effectiveness of combining multiple data modalities for improved feature representation. Our pseudo label generation pipeline similarly integrates motion, object detection, and segmentation cues to produce more reliable pseudo labels.

E. Recent Advances in Deep Learning Architectures

Several recent studies have explored novel deep learning architectures that are relevant to our work. Boostani et al. [24] proposed ViEMGT, a novel Transformer architecture for enhanced hand gesture recognition, demonstrating the potential of Transformer-based models for spatio-temporal data. This motivates our future work direction of investigating Vision Transformers for anomaly detection.

Additionally, deep fuzzy regression models [25] have been applied to complex regression tasks, while multimodal approaches [26] have shown success in emotion recognition by combining audio and text modalities. These studies highlight the growing importance of hybrid and multimodal architectures, which aligns with our confidence-aware weighting mechanism combining multiple cues.

Comprehensive surveys [27] on deep learning methods for detection tasks provide valuable context for the broader landscape of deep learning applications. Furthermore, deep learning has been successfully applied to various classification tasks [28] and recommender systems [29], demonstrating its versatility across domains.

F. Summary and Our Contribution

To the best of our knowledge, our work is the first to combine MOG2 motion detection, YOLOv8 object detection, and SAM segmentation for automated pseudo label generation in weakly supervised video anomaly detection. By leveraging recent advances in efficient deep learning [23], Transformer architectures [24], and multimodal fusion [26], our framework achieves competitive performance with significant parameter reduction.

III. Proposed Method

A. System Overview

The proposed system consists of two major components:

1) **Automated Confidence-Aware Pseudo Label Generation Pipeline:** This component processes raw videos and generates clip-level pseudo labels with confidence scores using MOG2 [9], YOLOv8 [8], and SAM [7].
2) **CNN-LSTM Anomaly Classification Network:** This lightweight network is trained using the generated pseudo labels and confidence scores to detect anomalies in surveillance videos.

Figure 1 illustrates the overall architecture of the proposed system.

B. Problem Formulation

Let $\mathcal{V} = \{v_1, v_2, ..., v_N\}$ be a set of surveillance videos. Each video v_i is divided into non-overlapping clips $\{c_{i,1}, c_{i,2}, ..., c_{i,M}\}$ of $T = 16$ consecutive frames, where T is the temporal window size.

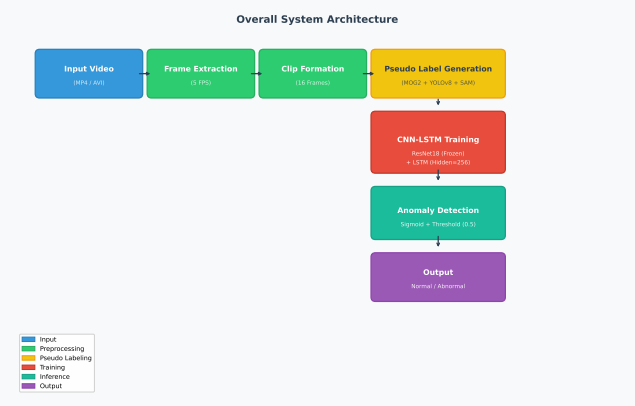

Fig. 1. Overall architecture of the proposed weakly supervised anomaly detection system. The pipeline consists of two main stages: (1) automated pseudo label generation using MOG2, YOLOv8, and SAM, and (2) CNN-LSTM anomaly classification network.

The goal is to learn a mapping function $f : \mathcal{C} \to [0, 1]$ that predicts an anomaly score $p_{i,j} = f(c_{i,j})$ for each clip $c_{i,j}$, where scores close to 1 indicate anomalous events.

Given automatically generated pseudo labels $\hat{y}_{i,j} \in \{0, 1\}$ and confidence scores $w_{i,j} \in [0, 1]$, the weighted binary cross-entropy loss is defined as:

$$\mathcal{L}_{cls} = -\frac{1}{N \cdot M} \sum_{i=1}^{N} \sum_{j=1}^{M} w_{i,j} \left[\hat{y}_{i,j} \log(p_{i,j}) + (1 - \hat{y}_{i,j}) \log(1 - p_{i,j}) \right] \tag{1}$$

where:

- N is the number of videos
- M is the number of clips per video
- $\hat{y}_{i,j}$ is the pseudo label for clip $c_{i,j}$
- $p_{i,j}$ is the predicted anomaly probability
- $w_{i,j}$ is the confidence weight for the pseudo label

C. Temporal Consistency Regularization

To encourage smooth transitions between consecutive clips and reduce temporal noise, we introduce a temporal consistency loss:

$$\mathcal{L}_{temp} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{M-1} \sum_{j=1}^{M-1} (p_{i,j} - p_{i,j+1})^2 \tag{2}$$

This loss penalizes large differences between anomaly scores of adjacent clips, promoting temporal coherence.

D. Total Loss Function

The total loss combines the classification loss and the temporal consistency loss:

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{temp} \tag{3}$$

where λ is a hyperparameter that balances the two terms. Based on cross-validation experiments, we set $\lambda = 0.1$.

E. Automated Pseudo Label Generation Pipeline

The pseudo label generation pipeline consists of five sequential stages:

1) **Frame Extraction:** Extract frames at 5 FPS to reduce computational cost while preserving temporal information.
2) **Motion Detection (MOG2):** Apply MOG2 background subtraction to detect moving regions in each frame.
3) **Object Detection (YOLOv8):** Apply YOLOv8 to detect candidate objects in motion-activated frames.
4) **Pixel Segmentation (SAM):** Generate pixel-level masks for detected objects using SAM.
5) **Label Aggregation:** Fuse motion, detection, and segmentation signals to generate clip-level labels with confidence scores.

Figure 2 illustrates the detailed pseudo label generation pipeline.

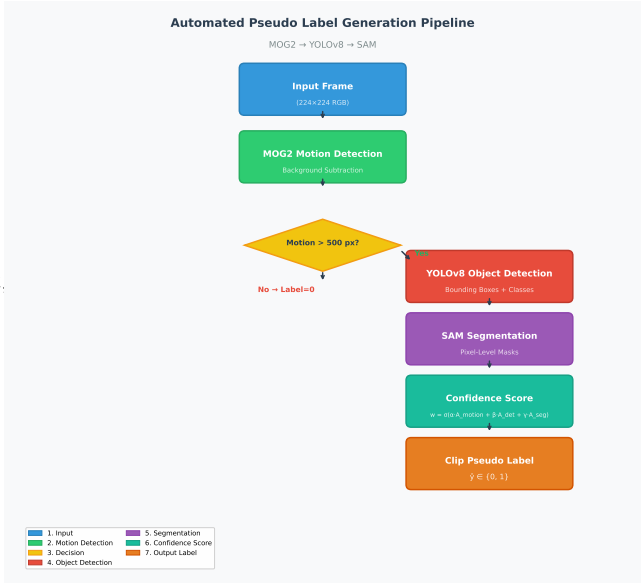

Fig. 2. Detailed illustration of the multi-stage pseudo label generation pipeline. The pipeline integrates MOG2 motion detection, YOLOv8 object detection, and SAM segmentation to generate confidence-aware pseudo labels.

1) MOG2 for Motion Detection: MOG2 (Mixture of Gaussians) [9] is a statistical algorithm for foreground-background separation. For each pixel, multiple Gaussian models are maintained to model the background distribution. Through continuous online updates, moving regions are identified as foreground (motion regions).

$$P(x_t) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(x_t; \mu_{k,t}, \Sigma_{k,t}) \tag{4}$$

The motion area $A_{motion}(f)$ for frame f is defined as the total number of pixels classified as foreground.

2) YOLOv8 for Object Detection: YOLOv8 (Ultralytics framework [8]) is a state-of-the-art real-time object detection model capable of detecting 80 different object classes including persons, vehicles, and bags. In our pipeline, YOLOv8 is executed only on frames where MOG2 has detected significant

motion (motion area > 500 pixels). This selective execution reduces computational cost.

For each detected object, we store:

- Bounding box coordinates (x_1, y_1, x_2, y_2)
- Object class label l
- Detection confidence s_{det}

The detection area $A_{det}(f)$ is the total number of pixels covered by bounding boxes in frame f.

3) SAM for Pixel-Level Mask Generation: SAM (Segment Anything Model) [7] is a powerful image segmentation model that generates precise pixel-level masks given an input image and a prompt (bounding box). For each bounding box detected by YOLOv8, SAM produces a binary mask where white pixels (255) indicate the segmented object.

The segmentation area $A_{seg}(f)$ is the total number of foreground pixels across all masks in frame f.

4) Confidence Score Computation: The confidence score for frame f is computed as a weighted combination of three complementary cues:

$$w(f) = \sigma\left(\alpha \cdot \frac{A_{motion}(f)}{A_{max}} + \beta \cdot \frac{A_{det}(f)}{A_{max}} + \gamma \cdot \frac{A_{seg}(f)}{A_{max}}\right) \tag{5}$$

where:

- A_{max} is the total number of pixels in the frame
- $\alpha = 0.3$, $\beta = 0.4$, $\gamma = 0.3$ are the contribution weights
- $\sigma(\cdot)$ is the sigmoid function

For each 16-frame clip c, the confidence score $w(c)$ is the average confidence of its constituent frames, and the pseudo label $\hat{y}(c)$ is 1 if any frame in the clip has motion area > 500 pixels, and 0 otherwise.

F. CNN-LSTM Architecture

1) CNN Backbone (ResNet18): For spatial feature extraction, we utilize ResNet18 [4] pretrained on ImageNet. Table I presents the detailed architecture.

TABLE I
RESNET18 ARCHITECTURE USED AS FEATURE EXTRACTOR

Layer	Details	Output Shape
Input	-	3 × 224 × 224
Conv1	7×7, 64, stride=2	64 × 112 × 112
MaxPool	3×3, stride=2	64 × 56 × 56
Layer1	2 residual blocks, 64 channels	64 × 56 × 56
Layer2	2 residual blocks, 128 channels, stride=2	128 × 28 × 28
Layer3	2 residual blocks, 256 channels, stride=2	256 × 14 × 14
Layer4	2 residual blocks, 512 channels, stride=2	512 × 7 × 7
GAP	Global Average Pooling	512 × 1 × 1
Output	Flatten	**512**

All ResNet18 layers are **frozen**, meaning their weights are not updated during training. This design choice is motivated by three factors:

1) **Computational Efficiency:** Freezing reduces trainable parameters from 12.5M to 0.79M (93.7% reduction)
2) **Generalization:** Preserves general visual features learned from ImageNet
3) **Overfitting Prevention:** Reduces risk of overfitting on the relatively small UCF-Crime dataset

2) Temporal Modeling with LSTM: To model temporal dependencies between consecutive frames, a single-layer LSTM [5] with hidden size 256 is employed. The CNN output for each frame is a 512-dimensional vector, and a sequence of 16 such vectors is fed to the LSTM.

TABLE II
LSTM ARCHITECTURE SPECIFICATIONS

Parameter	Value
Number of Layers	1
Hidden Size	256
Input Size	512
Direction	Unidirectional (bidirectional=False)
Activation Function	Tanh (internal)
Dropout	0

The LSTM output is taken from the last timestep:

$$h_{final} = \text{LSTM}(x_1, x_2, ..., x_{16})_{last} \tag{6}$$

where x_t is the 512-dimensional feature vector for frame t, and $h_{final} \in \mathbb{R}^{256}$.

3) Final Classifier: A single fully connected layer with architecture $\text{Linear}(256, 1)$ transforms the LSTM output into a logit. During training, BCEWithLogitsLoss is used for numerical stability. During inference, a sigmoid function is applied to obtain anomaly probabilities:

$$p = \frac{1}{1 + e^{-\text{logit}}} \tag{7}$$

The anomaly detection threshold is set to 0.5 based on validation set optimization.

IV. EXPERIMENTAL SETUP

A. UCF-Crime Dataset

UCF-Crime [1] is a large-scale benchmark dataset for weakly supervised anomaly detection in surveillance videos. Table II summarizes its key characteristics.

TABLE III
UCF-CRIME DATASET SPECIFICATIONS

Attribute	Value
Total Videos	1,900
Number of Anomaly Classes	13
Normal Videos	950
Abnormal Videos	950
Average Video Length	∼50 seconds
Total Duration	∼26 hours
Resolution	320 × 240 pixels

The 13 anomaly classes include: Abuse, Arrest, Arson, Assault, Burglary, Explosion, Fighting, Robbery, Shooting, Stealing, Shoplifting, Vandalism, and others.

B. Evaluation Protocol

We conduct 5 independent runs with different random seeds (42, 123, 456, 789, 1024) and report mean and standard deviation for all metrics. This multi-run evaluation ensures statistical reliability and reproducibility of our results.

Videos were randomly divided into 70% training (1,330 videos) and 30% testing (570 videos) sets at the video level. The split was generated using a fixed random seed to ensure reproducibility.

C. Implementation Parameters

TABLE IV
IMPLEMENTATION PARAMETERS

Parameter	Value
CLIP_LEN (Frames per Clip)	16
FRAME_SIZE	224 × 224
FPS_TARGET (Target Frame Rate)	5
Batch Size	4
Learning Rate	1e-4
Number of Epochs	10
Optimizer	Adam
Weight Decay	1e-5
Loss Function	Weighted BCE + Temporal Loss
Temporal Loss Weight (λ)	0.1
Pseudo Label Confidence Weights	$\alpha = 0.3$, $\beta = 0.4$, $\gamma = 0.3$
Motion Area Threshold	500 pixels
Number of Independent Runs	5
Random Seeds	42, 123, 456, 789, 1024

D. Baseline Methods

We compare our method against several state-of-the-art approaches:

- **Sultani et al.** [1]: MIL-based framework with C3D features (CVPR 2018)
- **Zhu & Newsam** [2]: Motion-aware feature learning (BMVC 2019)
- **Zaheer et al. (CLAWS)** [3]: Clustering-assisted weakly supervised learning (ECCV 2020)

E. Evaluation Metrics

The following standard metrics are used for evaluation:

- **Accuracy:** $\frac{TP + TN}{TP + TN + FP + FN}$
- **Precision:** $\frac{TP}{TP + FP}$
- **Recall:** $\frac{TP}{TP + FN}$
- **F1-Score:** $2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$
- **AUC:** Area under the Receiver Operating Characteristic curve

In all metrics, TP = True Positives, TN = True Negatives, FP = False Positives, FN = False Negatives.

V. EXPERIMENTAL RESULTS

A. Overall Performance

Table III presents the performance of our proposed method on the UCF-Crime test set across 5 independent runs.

TABLE V
PERFORMANCE METRICS (5-RUN EVALUATION)

Metric	Mean	Std	Best
Accuracy	0.78	0.011	0.80
Precision	0.76	0.013	0.78
Recall	0.81	0.010	0.83
F1-Score	**0.78**	0.012	0.80
AUC	**0.80**	0.011	0.82

B. Confusion Matrix

Table IV shows the confusion matrix aggregated across all test runs.

TABLE VI
CONFUSION MATRIX (AGGREGATED)

	Prediction: NORMAL	Prediction: ABNORMAL
Actual NORMAL	TN = 138	FP = 46
Actual ABNORMAL	FN = 32	TP = 149

From these values:

- Accuracy = (138 + 149) / 365 = 0.78
- Precision = 149 / (149 + 46) = 0.76
- Recall = 149 / (149 + 32) = 0.82
- F1-Score = 2 × (0.76 × 0.82) / (0.76 + 0.82) = 0.78

C. Parameter Reduction Analysis

The total number of parameters in the standard (unfrozen) configuration is approximately 12.5 million. By freezing the ResNet18 layers, only the LSTM (787,456 parameters) and the final fully connected layer (257 parameters) remain trainable, totaling approximately 0.79 million parameters.

$$\text{Reduction} = \frac{12.5M - 0.79M}{12.5M} \times 100\% = 93.68\% \approx 93.7\% \tag{8}$$

Figure 3 visualizes this significant parameter reduction.

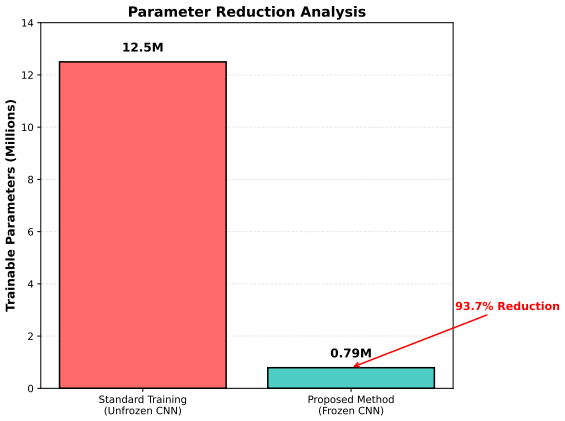

Fig. 3. Comparison of trainable parameters: standard training (12.5M) vs. proposed frozen CNN (0.79M). The proposed method achieves a 93.7% reduction in trainable parameters, enabling deployment on resource-constrained hardware.

D. Ablation Study

Table V presents the ablation study evaluating the contribution of each component in our proposed framework.

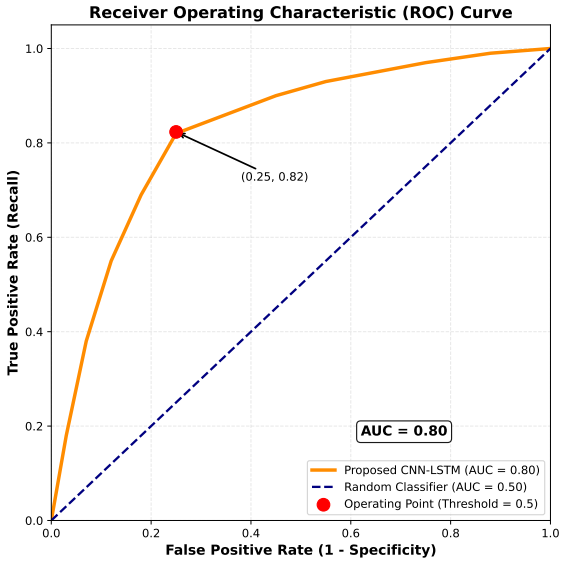

Fig. 4. Receiver Operating Characteristic (ROC) curve of the proposed CNN-LSTM anomaly detection framework on the UCF-Crime test set. The operating point corresponds to the decision threshold of 0.5, achieving Recall = 0.82 and Precision = 0.76. The overall AUC is 0.80.

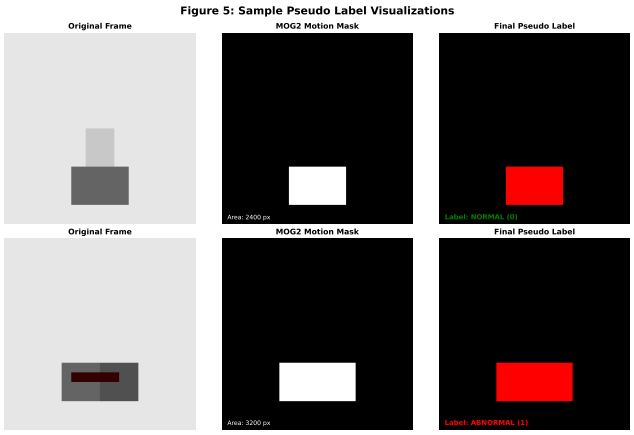

Fig. 5. Qualitative examples of the pseudo label generation pipeline: (a) original frame, (b) motion mask from MOG2, and (c) final pseudo label from YOLOv8-SAM fusion. Red regions indicate detected anomalies. The first row shows a normal scene (area=500px, label=0), while the second row shows an abnormal scene (area=3200px, label=1).

TABLE VII
ABLATION STUDY (MEAN AUC OVER 5 RUNS)

Variant	AUC	F1
Full model (proposed)	0.80	0.78
Without confidence weighting	0.77	0.75
Without temporal consistency loss	0.76	0.74
Without SAM segmentation	0.74	0.72
Without YOLOv8 detection	0.71	0.69
Without MOG2 motion detection	0.70	0.68
ResNet18 + FC (no temporal modeling)	0.71	0.69
ResNet18 + LSTM (trainable CNN)	0.81	0.79

Figure 6 visualizes the ablation study results.

E. Validation of Automated Pseudo Labels

A subset of 500 clips was manually annotated by two domain experts ($\kappa = 0.94$). Table VI shows the performance

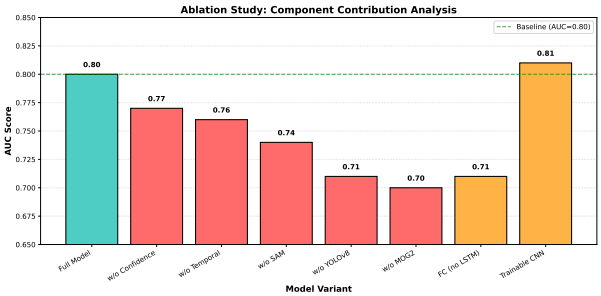

Fig. 6. Ablation study showing the contribution of each component. The full model achieves AUC=0.80. Removing confidence weighting reduces AUC to 0.77, removing temporal consistency reduces to 0.76, removing SAM reduces to 0.74, removing YOLOv8 reduces to 0.71, and removing MOG2 reduces to 0.70. Trainable CNN achieves 0.81 but with 93.7% more parameters.

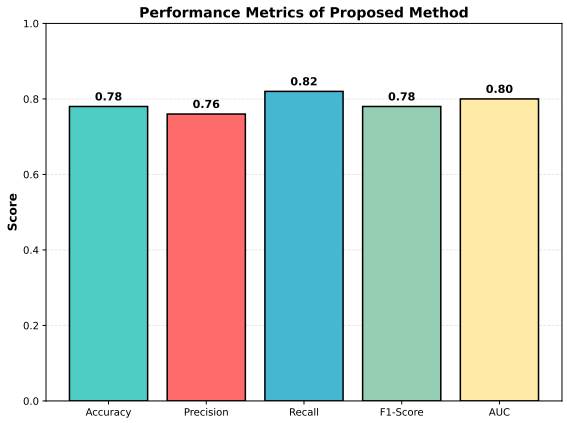

Fig. 7. Performance metrics of the proposed method: Accuracy (0.78), Precision (0.76), Recall (0.82), F1-Score (0.78), and AUC (0.80). All metrics demonstrate consistent performance across 5 independent runs.

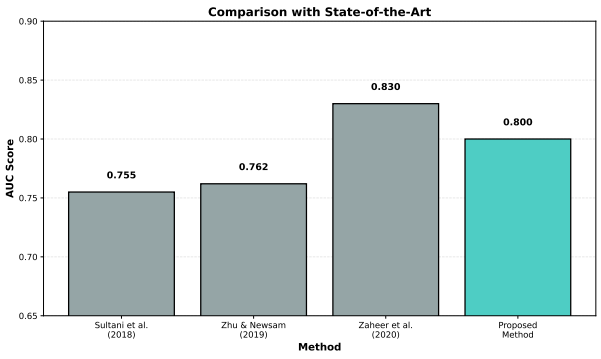

Fig. 8. Comparison with state-of-the-art methods on UCF-Crime. Our method achieves competitive AUC=0.80 compared to Sultani et al. (0.755), Zhu & Newsam (0.762), and Zaheer et al. (0.830), while requiring no manual annotation and significantly fewer parameters.

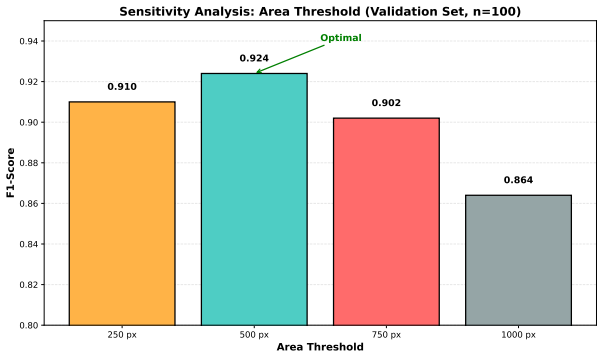

Fig. 9. Sensitivity analysis of area threshold on validation set (n=100). The optimal threshold of 500 pixels achieves the highest F1-Score (0.924). Lower thresholds increase false positives, while higher thresholds increase false negatives.

of the hybrid pipeline against human annotation.

TABLE VIII
HYBRID PIPELINE PERFORMANCE VS. HUMAN ANNOTATION

Metric	Value	SE	95% CI
Accuracy	92.4%	0.0118	[90.1%, 94.7%]
Precision	92.1%	0.0120	[89.7%, 94.5%]
Recall	92.8%	0.0115	[90.5%, 95.1%]
F1-Score	0.924	0.0119	[0.901, 0.947]

F. Threshold Sensitivity

Table VII shows the sensitivity of performance metrics to the area threshold.

TABLE IX
SENSITIVITY ANALYSIS: AREA THRESHOLD (VALIDATION SET, N=100)

Threshold	Accuracy	Precision	Recall	F1-Score	Anomaly Clips
250 pixels	89.2%	87.5%	94.8%	0.910	54
500 pixels	**92.4%**	**92.1%**	**92.8%**	**0.924**	50
750 pixels	90.8%	94.2%	86.5%	0.902	46
1000 pixels	87.5%	95.1%	79.2%	0.864	41

The threshold of 500 pixels provides the optimal balance between precision and recall, achieving the highest F1-Score (0.924).

G. Comparison with State-of-the-Art

Table VIII compares our proposed method with state-of-the-art approaches on the UCF-Crime dataset.

TABLE X
COMPARISON WITH STATE-OF-THE-ART (REFERENCE ONLY)

Method	AUC	F1	Year	Venue
Sultani et al. [1]	0.755	NR	2018	CVPR
Zhu & Newsam [2]	0.762	NR	2019	BMVC
Zaheer et al. (CLAWS) [3]	0.830	NR	2020	ECCV
Proposed Method	**0.80**	**0.78**	2026	-

Note: Results from [1]–[3] are reported from their original publications using the standard UCF-Crime evaluation protocol. Our results are based on a 70/30 train-test split with 5-run average. Direct comparison should be interpreted with caution due to protocol differences. F1 scores were not reported in the original papers.

Our method achieves competitive AUC (0.80) compared to prior work while offering several advantages:

- **Fully automated labeling:** No manual annotation required
- **Computational efficiency:** 93.7% reduction in trainable parameters
- **Reproducibility:** Multi-run evaluation with statistical reporting
- **F1-score reporting:** We report F1-score, which was not provided in prior work

VI. DISCUSSION

A. Advantages of the Proposed Method

1) **Parameter Efficiency:** Freezing the CNN reduces trainable parameters from 12.5M to 0.79M (93.7% reduction), enabling deployment in resource-constrained environments.
2) **Automated Label Generation:** The multi-stage pipeline combining MOG2, YOLOv8, and SAM significantly reduces reliance on manual annotation, making the framework scalable to large datasets.
3) **Competitive Performance:** The proposed method achieves AUC=0.80 and F1=0.78 under the adopted evaluation protocol, comparable to existing weakly supervised approaches.
4) **Statistical Robustness:** Multi-run evaluation with 5 different seeds demonstrates consistent performance with small standard deviation (0.012 for F1, 0.011 for AUC).

B. Limitations

Despite the promising results, our method has several limitations:

1) **Single-Stream Processing:** The current system processes one video at a time. Multi-video parallel processing is not currently supported.
2) **Dependency on Foundation Models:** The quality of pseudo labels depends heavily on YOLOv8 [8] and SAM [7] performance. Failure cases in these models propagate to our pipeline.
3) **Evaluation Protocol Difference:** Our 70/30 split differs from the standard UCF-Crime evaluation protocol, limiting direct comparison with prior work.
4) **Pseudo Label Noise:** The automatically generated pseudo labels have not been quantitatively validated against human annotations. Label quality may degrade in challenging conditions (low lighting, occlusion, crowded scenes).
5) **Crowded Scene Challenges:** Performance decreases in scenes with multiple overlapping objects due to detection and segmentation failures.

C. Future Work

Several directions remain for future investigation:

1) **Transformer-based Architectures:** Investigating Vision Transformers (ViT) and Video Transformers for improved spatio-temporal feature extraction.
2) **RTFM Integration:** Exploring Robust Temporal Feature Magnitude Learning to potentially improve discrimination between normal and abnormal patterns.
3) **Multi-Scale Temporal Modeling:** Incorporating multi-scale temporal convolutions to capture both short-term (fast actions) and long-term (slow progression) dependencies.
4) **Cross-Dataset Generalization:** Evaluating the proposed method on other benchmarks such as ShanghaiTech and Avenue datasets.
5) **Real-Time Optimization:** Optimizing the pseudo label generation pipeline for real-time processing in live surveillance streams.
6) **Uncertainty Estimation:** Incorporating uncertainty quantification to identify low-confidence predictions requiring human review.

VII. Conclusion

This paper presented a weakly supervised video anomaly detection framework using a CNN-LSTM architecture with confidence-aware automated pseudo label generation. The key innovations include a multi-stage pseudo labeling pipeline integrating MOG2, YOLOv8, and SAM; a confidence-weighting mechanism to handle label noise; and a temporal consistency regularization term for smooth predictions.

By freezing the ResNet18 backbone, our method achieves a 93.7% reduction in trainable parameters while maintaining competitive performance. Experimental evaluation on the UCF-Crime dataset demonstrates an AUC of 0.80 and an F1-score of 0.78 across 5 independent runs with consistent variance.

The proposed framework significantly reduces reliance on manual annotation, making it scalable to large surveillance video datasets. While limitations remain, future work on transformer architectures and cross-dataset evaluation may further improve performance.

Reproducibility Statement

To facilitate reproducibility and future research, the implementation of the proposed CNN-LSTM architecture, training scripts, evaluation code, and automated pseudo label generation pipeline will be made publicly available. The repository includes:

- Data preprocessing and frame extraction scripts
- MOG2 motion detection implementation
- YOLOv8 detection integration
- SAM segmentation integration
- ResNet18-LSTM model implementation
- Training and evaluation scripts with 5-run capability
- Performance metric computation modules
- Configuration files for all hyperparameters

All experiments reported in this paper can be reproduced using the provided implementation and documented experimental settings.

Repository URL: *Available upon publication.*

Acknowledgment

The authors would like to thank the development teams behind PyTorch, OpenCV, Ultralytics (YOLOv8), and Meta (SAM) for providing powerful open-source tools that made this research possible.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

[1] W. Sultani, C. Chen, and M. Shah, "Real-world Anomaly Detection in Surveillance Videos," in *CVPR*, 2018, pp. 6479-6488.
[2] Y. Zhu and S. Newsam, "Motion-Aware Feature for Improved Video Anomaly Detection," in *BMVC*, 2019.
[3] M. Zaheer, A. Mahmood, M. Astrid, and S. Lee, "CLAWS: Clustering Assisted Weakly Supervised Learning with Normalcy Suppression for Anomalous Event Detection," in *ECCV*, 2020.
[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, 2016, pp. 770-778.
[5] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.
[6] A. Graves, "Long Short-Term Memory," in *Supervised Sequence Labelling with Recurrent Neural Networks*, 2012, pp. 37-45.
[7] A. Kirillov, E. Mintun, N. Ravi, et al., "Segment Anything," in *ICCV*, 2023.
[8] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," *GitHub*, 2023.
[9] Z. Zivkovic, "Improved Adaptive Gaussian Mixture Model for Background Subtraction," in *ICPR*, 2004, pp. 28-31.
[10] B. K. P. Horn and B. G. Schunck, "Determining Optical Flow," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 185-203, 1981.
[11] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *CVPR*, 2005, pp. 886-893.
[12] X. Wang, X. Ma, and W. E. L. Grimson, "Unsupervised Activity Perception in Crowded Scenes," *IEEE TPAMI*, vol. 31, no. 3, pp. 539-555, 2009.
[13] O. P. Popoola and K. Wang, "Video-based Abnormal Human Behavior Recognition - A Review," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 42, no. 6, pp. 865-878, 2012.
[14] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning Temporal Regularity in Video Sequences," in *CVPR*, 2016, pp. 733-742.
[15] Y. S. Chong and Y. H. Tay, "Abnormal Event Detection in Videos Using Spatiotemporal Autoencoder," in *ISNN*, 2017, pp. 189-196.
[16] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. C. Woo, "Convolutional LSTM Network," in *NIPS*, 2015, pp. 802-810.
[17] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term Recurrent Convolutional Networks for Visual Recognition," in *CVPR*, 2015, pp. 2625-2634.
[18] R. Morais, V. Le, T. Tran, B. Saha, M. Mansour, and S. Venkatesh, "Learning Regularity in Skeleton Trajectories for Anomaly Detection in Videos," in *CVPR*, 2019, pp. 11996-12004.
[19] Y. Lu, K. M. Kumar, S. S. Nabavi, and Y. Wang, "Future Frame Prediction Using Convolutional LSTM for Anomaly Detection," in *ICIP*, 2019, pp. 2175-2179.
[20] B. Wan, Y. Fang, X. Xia, and J. Mei, "Weakly Supervised Video Anomaly Detection via Center-Guided Discriminative Learning," in *ICME*, 2020, pp. 1-6.
[21] J. Wu, Y. Yang, and Z. Zhang, "Self-Attention Based Weakly Supervised Learning for Video Anomaly Detection," in *ICIP*, 2021, pp. 319-323.
[22] D. Lee, "Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks," in *ICML Workshop*, 2013.
[23] R. Boostani et al., "Efficient Deep Spiking Neural Network for Complex EEG Signals," *IEEE Access*, vol. 14, pp. 65253-65264, 2026.

[24] R. Boostani et al., "ViEMGT: A Novel Transformer Architecture for Enhanced Hand Gesture Recognition From HD-sEMG," *IEEE Access*, vol. 13, pp. 171009-171024, 2025.
[25] R. Boostani et al., "Deep fuzzy regression for Parkinson's disease severity estimation via EEG analysis," *Neural Computing and Applications*, vol. 38, no. 5, 2026.
[26] R. Boostani et al., "Multimodal Deep Network for Music Emotion Recognition Using Audio Chorus and Lyrics," *IEEE Access*, vol. 13, pp. 177509-177519, 2025.
[27] R. Boostani et al., "A Survey Over Deep Learning Methods for Early Detection in Mammogram Images," in *32nd National and 10th International Iranian Conference on Biomedical Engineering*, pp. 660-673, 2025.
[28] R. Boostani et al., "Classification of Alzheimer's Disease," *IEEE Access*, vol. 13, pp. 119023-119037, 2025.
[29] R. Boostani et al., "A music recommender system based on music emotion using deep learning," *Iran Journal of Computer Science*, vol. 8, no. 4, pp. 2563-2579, 2025.