applied sciences

MDPI

Article

An Online Modular Framework for Anomaly Detection and Multiclass Classification in Video Surveillance

Jonathan Flores-Monroy [1,*], Gibran Benitez-Garcia [2], Mariko Nakano-Miyatake [1,*] and Hiroki Takahashi [2,3,4]

[1] Instituto Politécnico Nacional, ESIME Culhuacán, Mexico City 04440, Mexico
[2] Graduate School of Informatics and Engineering, The University of Electro-Communications, Chofugaoka 1-5-1, Chofu-shi 182-8585, Tokyo, Japan; gibran@ieee.org (G.B.-G.); rocky@inf.uec.ac.jp (H.T.)
[3] Artificial Intelligence eXploration Research Center (AIX), The University of Electro-Communications, Chofugaoka 1-5-1, Chofu-shi 182-8585, Tokyo, Japan
[4] Meta-Networking Research Center (MEET), The University of Electro-Communications, Chofugaoka 1-5-1, Chofu-shi 182-8585, Tokyo, Japan
* Correspondence: jfloresm1510@alumno.ipn.mx (J.F.-M.); mnakano@ipn.mx (M.N.-M.)

Abstract

Video surveillance systems are a key tool for the identification of anomalous events, but they still rely heavily on human analysis, which limits their efficiency. Current video anomaly detection models aim to automatically detect such events. However, most of them provide only a binary classification (normal or anomalous) and do not identify the specific type of anomaly. Although recent proposals address anomaly classification, they typically require full video analysis, making them unsuitable for online applications. In this work, we propose a modular framework for the joint detection and classification of anomalies, designed to operate on individual clips within continuous video streams. The architecture integrates interchangeable modules (feature extractor, detector, and classifier) and is adaptable to both offline and online scenarios. Specifically, we introduce a multi-category classifier that processes only anomalous clips, enabling efficient clip-level classification. Experiments conducted on the UCF-Crime dataset validate the effectiveness of the framework, achieving 74.77% clip-level accuracy and 58.96% video-level accuracy, surpassing prior approaches and confirming its applicability in real-world surveillance environments.

Keywords: modular framework; video anomaly detection; multiclass classification; clip-level classification; online processing; video surveillance

Academic Editor: Young-Gab Kim

Received: 16 July 2025
Revised: 14 August 2025
Accepted: 18 August 2025
Published: 22 August 2025

Citation: Flores-Monroy, J.; Benitez-Garcia, G.; Nakano-Miyatake, M.; Takahashi, H. An Online Modular Framework for Anomaly Detection and Multiclass Classification in Video Surveillance. *Appl. Sci.* **2025**, *15*, 9249. https://doi.org/10.3390/app15179249

1. Introduction

In recent years, urban safety has faced significant global challenges, especially in Latin America, where rapid urbanization and increasing crime rates have exacerbated the problem [1–3]. In Mexico, for instance, there were 5689 assaults per 100,000 inhabitants according to ENVIPE [1], highlighting the urgent need for innovative solutions. Strategies such as video surveillance systems and increased police presence have shown effectiveness in reducing crime [4–7]. However, video surveillance systems have critical limitations since they rely on human resources for continuous monitoring, leading to logistical and financial challenges and a high propensity for human error [7]. This underscores the need for methodologies that optimize existing processes, enabling rapid and effective responses to critical events by precisely detecting both the moment and the nature of the incident.

Within the field of artificial intelligence, the technique known as video anomaly detection (VAD) has gained prominence. This technique, part of computer vision, aims to

identify events that deviate from normal patterns in video sequences. In particular, the weakly supervised learning (wVAD) paradigm has become relevant due to its ability to train models using video-level labels without requiring precise temporal annotations [8–13]. Depending on the architecture employed, VAD models process information frame by frame, in short clips, or over longer segments.

In the research of VAD, most efforts have focused on anomaly detection, neglecting the classification of event types. Although there are proposals that integrate both tasks [14–16], they typically operate in parallel and in an offline mode, requiring the full video to be processed before producing a prediction. This condition limits their usefulness in real-world scenarios such as urban surveillance, public transportation, or continuous monitoring, where it is essential to identify and interpret events as they occur. Furthermore, these approaches generally require extensive post-processing or global video summaries, which reduces their responsiveness in time-sensitive contexts.

Considering these limitations, we identified that the main challenge is not only detecting the presence of an anomaly but also classifying it precisely and immediately, using minimal units of information. This ability is crucial in real-world scenarios, where quick interpretation of events is key to activating timely response protocols. Therefore, a methodology is required that can process short video clips and integrate detection and classification into a single flow, while operating with low latency and adaptability across different components.

In this work, we propose a modular framework designed to enhance the capabilities of current VAD models, with a focus on their applicability in real-world environments. This framework consists of three main modules: a feature extractor, an anomaly detector, and a multi-category classifier. It is designed to operate continuously: once the system receives a video stream, it is divided into clips of T' consecutive non-overlapping frames, which constitute the basic processing unit. As part of the preprocessing, we apply a cropping-based data augmentation strategy, in which each frame in a clip can generate up to 10 crops: the center crop, the 4 corner crops, and their horizontal flips. Next, each clip is then processed using the feature extractor, which generates an abstract representation to feed into the anomaly detector. The detector produces an anomaly score for each clip; if this score exceeds a predefined threshold, the clip is sent to the classifier, which determines the specific type of anomaly among several predefined classes. Each module in the proposed framework, including preprocessing steps such as cropping, is interchangeable depending on the purpose and application.

After extensive experimentation, we select a suitable algorithm for each module to achieve an online video surveillance system, in which anomalous events are detected and classified at the clip level through a continuous stream. For the feature extractor, the Unified Transformer Small version (UniFormer-S) is selected after meticulous evaluation, compared to several feature extractors used in VAD. Further details of this evaluation can be found in our previous work [17]. As the anomaly detector, we employed Coarse-to-Fine Pseudo-Labeling (C2FPL) [18], which enables clip-level anomaly detection and uses 10 crops per frame. The classifier receives the anomaly clips from the detector and classifies them into one of the different anomalous events, such as fighting, shooting, or shoplifting. In this case, only the center crop of each frame is used. This last module is our proposal and constitutes the main contribution of this work.

To evaluate the effectiveness of our framework, we used the UCF-Crime database [8]. Since the classifier is the main module developed in this proposal, the analysis focused on its performance. In this configuration, we achieved an accuracy of 58.96%, surpassing the values reported by previous methods. In addition, we obtained complementary results for F1-score, precision, and recall, which reinforce the effectiveness of our approach in

identifying different types of events under online conditions. The main contributions of our work are as follows:

- Proposal of a modular framework for VAD, which is composed of three exchangeable modules: a feature extractor, an anomaly detector, and an anomalous event classifier.
- Design of a multi-category classifier capable of operating on anomalous clips in a continuous stream.
- Development of an online video surveillance system using a modular framework, in which an anomaly event is detected and classified at the clip level.
- The performance of the proposed online framework outperforms previous VADs using the publicly available UCF-Crime dataset.

This work is an extension of the paper published at the SOMET2024 conference [10], where the main contributions of that prior work were as follows:

- Comprehensive evaluation of five typical feature extractors in VAD: 3D Convolutional Neural Networks (C3D) [19], Inflated 3D ConvNet (I3D) [20], Temporal Shift Module (TSM) [21], Unified Transformer Small (UniFormer-S) [17] and Unified Transformer Base (UniFormer-B) [17] in a state-of-the-art anomaly detector proposed by Weijun et al. [9].
- Identification of UniFormer-S as the most balanced extractor in terms of accuracy, computational cost, and processing speed.
- Validation of this extractor through tests on edge devices, highlighting its feasibility for real-world environments.

The remainder of this manuscript is organized as follows: Section 2 reviews the most relevant related work, covering existing techniques for anomaly detection and classification in video. Section 3 describes the materials and methods used, including the architecture of the proposed modular framework and the training process of the multiclass classifier. Section 4 presents the experimental results, reporting the performance of the classifier and its integration with different anomaly detectors. Section 5 discusses the results in relation to the research objectives and the broader context. Finally, Section 6 outlines the conclusions and potential directions for future work.

2. Related Work

2.1. Feature Extractors

In VAD, architectures originally developed for video action recognition (VAR) have been adopted, as they effectively capture the spatiotemporal information essential for detecting unusual behaviors.

A pioneering approach is the use of 3D Convolutional Neural Networks (C3D) [19], which, unlike 2D ConvNets that extract only spatial information from each frame independently, integrate temporal and spatial information jointly through three-dimensional convolutions and poolings. Works such as those by Sultani et al. [8] and RTFM [12] employ this architecture to process video clips and extract deep representations (e.g., from the FC6 layer), maintaining the dynamic relationships between frames.

Subsequently, more advanced architectures such as Inflated 3D ConvNets (I3D) [20] emerged, inflating 2D filters to three dimensions to learn actions and patterns over time with greater fidelity. In VAD, I3D has been established as a successor to C3D in various studies [9,11,12,18], leveraging layers like `mixed_5c` to extract representations that combine spatial and temporal information.

The Unified Transformer (UniFormer) [17] represents an advancement by integrating 3D convolution and spatiotemporal attention mechanisms in a single architecture, efficiently capturing both local and global dependencies. It has been employed in VAD scenarios to

process 32-frame clips and prioritize anomalous regions by measuring Euclidean distances between normal and anomalous clips [13].

Alternatively, the Temporal Shift Module (TSM) [21] offers an efficient mechanism to incorporate the temporal dimension without significantly increasing computational load. TSM shifts channels of feature maps forward or backward in time, enabling 2D architectures to simulate lightweight 3D behavior. ADNet [22] combines TSM with I3D to process sliding windows of video and maximize class separation using a specially designed loss function, demonstrating the usefulness of TSM in segmenting anomalous events.

A recent trend in VAD is the adoption of Contrastive Language–Image Pre-training (CLIP) [23] as a feature extractor. CLIP combines visual and textual information learned from large data corpora, generating multimodal embeddings that capture not only spatial and temporal features but also semantic information. Thanks to its contrastive training, where image representations are aligned with corresponding textual descriptions, CLIP offers very rich and generalizable visual representations. This capability has made it a powerful tool to enhance the description of each frame or segment in anomaly detection and classification tasks. For example, Wu et al. [15] integrated it as part of their pipeline to enrich visual characterization and facilitate the identification of anomalous patterns.

2.2. Anomaly Detectors

In VAD, various approaches or paradigms exist, whose choice mainly depends on the dataset characteristics and the type of available labels. Some models are developed under a completely unsupervised scheme, others under a supervised one, but the most widely adopted approach is the weakly supervised paradigm. This paradigm uses global video-level labels (normal or anomalous), significantly reducing the annotation effort compared to methodologies requiring frame-by-frame segmentations.

In the weakly supervised paradigm, the model must learn to detect anomalous patterns from this limited global information, without precise indications of when or where the anomaly occurs in the sequence. This poses the challenge of inferring the temporal location of anomalies and distinguishing them from normal patterns, processing basic units such as clips or fixed video segments.

One of the most representative methods in this context is Multiple Instance Learning (MIL) [8]. In MIL, videos are divided into segments grouped into “bags”: positive (videos labeled as anomalous) and negative (normal videos). It is assumed that negative bags contain only normal instances, while positive ones may include both normal and anomalous instances. The goal is for the model to learn to identify the most relevant instances in positive bags that explain the anomaly. Typically, a ranking loss function is used to maximize the distance between the most abnormal instances in positive bags and the most abnormal ones in negative bags. Although MIL manages to separate relevant instances, it suffers from noisy labels since the exact location of the anomalies is not available. This has motivated the development of complementary models that aim to improve robustness by addressing label noise and refining instance selection.

For example, Graph Convolutional Networks (GCNs) [11] explicitly reformulate video anomaly detection as a supervised learning task under noisy labels, where normal snippets within anomalous videos act as noise. To address this, GCN propagates supervision from high-confidence clips to uncertain ones based on feature similarity and temporal consistency, effectively correcting noisy annotations through a graph-based structure. This allows the use of standard supervised classifiers with improved label quality.

Another relevant strategy is Robust Temporal Feature Magnitude Learning (RTFM) [12], which mitigates the dominance of negative instances in MIL by learning feature magnitudes that emphasize subtle positive patterns. RTFM uses temporal magnitudes and

multilevel temporal networks (MTNs) to highlight relevant instances and reduce the impact of noise in weakly labeled data. Some recent approaches have also explored integrating transformer-based architectures, whose attention mechanisms capture long-range dependencies in video sequences, enriching representations and improving precision in identifying atypical behaviors.

A notable extension of MIL was proposed by Weijun et al. [9], who integrated the Bidirectional Encoder Representations from Transformers (BERT) architecture [24], known for modeling long-range contextual relationships. In this methodology, MIL still functions as the base for local detection: videos are divided into clips and then into fixed segments grouped into positive and negative bags, aiming for the model to identify the most representative anomalous segments. However, the integration of BERT adds a global classification vector at the video level (a summarized representation of the video's overall context), complementing the local MIL scores. This vector is derived from BERT's contextual analysis of the feature sequence, leveraging its bidirectional attention to capture long-range relationships. During inference, anomaly scores generated locally by MIL are combined with this global prediction from BERT, resulting in a more robust estimation of the presence and nature of anomalous events.

Finally, a recent and notable approach is the Coarse-to-Fine Pseudo-Labeling (C2FPL) framework developed by Al-lahham et al. [18]. Although formulated as a completely unsupervised paradigm, C2FPL can be directly integrated into a weakly supervised context by replacing the video-level pseudo-labels with actual global labels. This method has two stages: first, pseudo-labels for videos are generated using a divisive hierarchical clustering that classifies videos as normal or anomalous based on global feature statistics; second, these labels are refined at the clip level through statistical hypothesis testing that identifies the most anomalous clips in videos classified as anomalous. A clip-level anomaly classifier is then trained with these refined pseudo-labels, allowing the system to assign precise anomaly scores to each clip. Unlike MIL-based methods that require the complete video during inference, C2FPL can process each clip independently during inference, making it more suitable for online environments where immediate clip-level predictions are critical.

In general, anomaly detection methods focus solely on identifying the presence of atypical behaviors in videos, without providing detailed information about the nature of these events. This lack of classification limits their usefulness in real-world scenarios that demand not only anomaly detection but also understanding their context and meaning to trigger appropriate responses. Furthermore, many approaches require processing the full video to produce reliable predictions, hindering their integration into continuous or progressive analysis systems that require fast responses based on minimal units of information.

2.3. Multi-Category Classifiers

The classification of anomalous events has emerged as a complementary component in anomaly detection systems, incorporating models that integrate specific modules to identify the exact nature of the event. Sultani et al. [8] propose two different approaches for this task, both focusing on full-video level classification. In the first approach, videos are segmented into 16-frame clips, features are extracted using C3D, and these features are averaged and normalized via L2 norm, producing a single representative vector that is classified using a Nearest Neighbor method. In the second approach, they incorporate the Tube Convolutional Neural Network (T-CNN) architecture [25], which replaces a pooling layer of C3D with a temporal aggregation module (Tube of Interest Pooling), generating a global vector used directly for classification.

Majhi et al. [14] present a unified model for simultaneous detection and classification. Videos are divided into fixed temporal segments, and their features are extracted via

a feature extractor (FE). These features are refined with an LSTM to capture temporal information, while an initial attention layer highlights the most relevant segments. From this refined feature map, two branches are established: one for detection, which assigns anomaly scores using the MIL ranking loss function [8], and another for classification, which uses a second attention layer and a global average of the refined features followed by a Softmax layer to determine the event category.

Wu et al. [15] introduce a more modern methodology that combines detection and classification using frame-level features extracted by CLIP [23]. These features are refined through a temporal adapter module and a semantic injection module to capture both temporal and contextual relationships. The detection stage computes anomaly scores per frame and selects the top-K highest values, whose average feeds a sigmoid function to generate binary predictions. For classification, this top-K average is aligned with textual embeddings generated by CLIP, allowing the assignment of specific categories through joint optimization of detection and classification losses.

Lastly, Ullah et al. [16] propose a model focused on classifying anomalies in surveillance videos by processing spatial features at the frame level with MobileNetV2 [26] and grouping them into sequences of 30 consecutive frames. These sequences are refined temporally through an LSTM with residual attention and classified via a Softmax layer.

Although some recent methods have incorporated anomaly classification as a complement to detection, their implementation remains limited in the following key aspects. First, these models require the full video to be processed in order to determine the anomaly category. This implies that classification cannot be performed as data arrives but only after analyzing the entire sequence. This condition represents a significant obstacle in real-world environments that demand processing and classification as information is generated, maximizing response speed and accuracy. Additionally, many of these methods structure their workflows in separate branches for detection and classification, requiring full data processing in both routes and creating an additional computational burden that hinders their adoption in real-world applications. Consequently, an approach is needed that enables progressive classification from minimal information units, ensuring more agile and effective integration in these environments.

3. Materials and Methods

In this section, we present the inference process of the proposed modular framework, which is illustrated in Figure 1. Additionally, we describe the training procedure of the multi-category classifier module, covering both the construction of the dataset required for training and the architecture of the classifier itself.

3.1. Modular Framework

The proposed framework is built on the principle that each of its components operates independently. The general methodology of our approach is illustrated in Figure 1. In this process, the input data (denoted as V) comes from a video stream. This video is first processed through a stage referred to as raw video preprocessing, which includes dividing the video into consecutive, non-overlapping clips of T' consecutive frames, resizing each frame to a fixed spatial resolution $H \times W$, and optionally generating up to 10 cropped versions of each clip. These are obtained by consistently applying a spatial crop (such as the center, one of the corners, or their horizontal flips) to all frames within a clip. The final output of this preprocessing stage is a set of C_r cropped clips per original clip, each with dimensions $h \times w \times T' \times Ch$, where C_r is the number of generated cropped versions, $h \times w$ is the resolution of each cropped frame, and Ch is the number of channels (e.g., RGB).

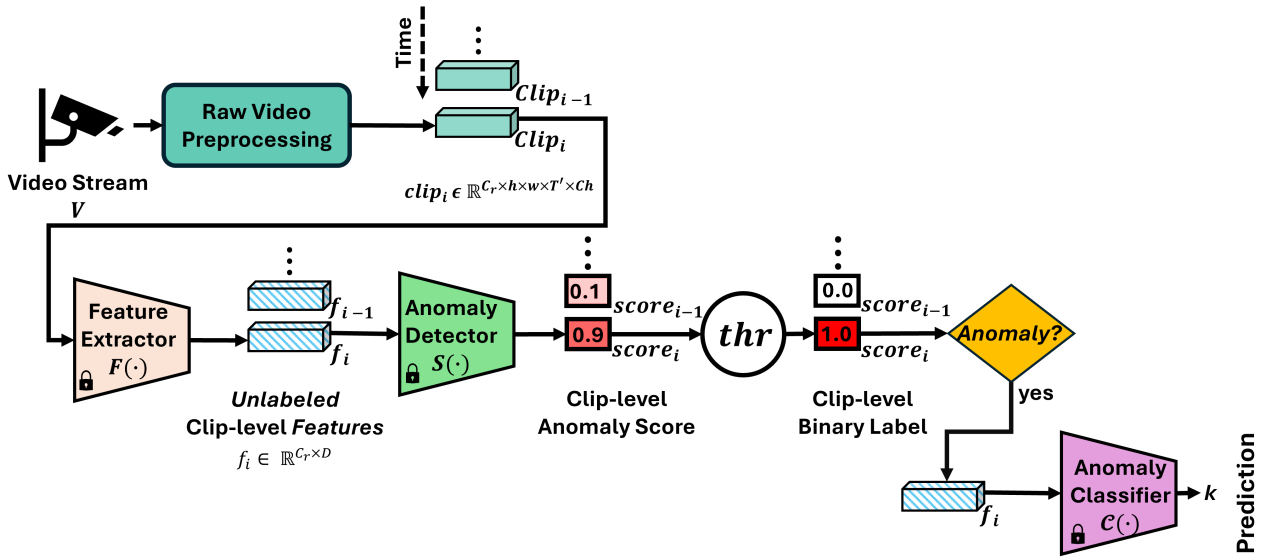

Figure 1. General flowchart of the proposed modular framework for clip-level anomaly detection and classification. The video stream V is divided into consecutive clips, resized and cropped (1 to 10 variants). Each clip is encoded by a pre-trained feature extractor $F(\cdot)$, evaluated by an anomaly detector $S(\cdot)$, and, if deemed anomalous, classified by $\mathcal{C}(\cdot)$ into a specific category. The lock icon indicates that the corresponding model is frozen, i.e., its weights remain unchanged during inference and no additional training is performed. The design enables continuous and progressive inference.

Subsequently, the feature extractor $F(\cdot)$, pre-trained and used without further fine-tuning, processes each $clip_i \in \mathbb{R}^{C_r \times h \times w \times T' \times Ch}$ and generates a feature matrix $f_i \in \mathbb{R}^{C_r \times D}$. D corresponds to the dimensionality of the spatiotemporal features extracted for each segment of the clip.

The extracted features f_i are sent to the anomaly detector $S(\cdot)$ (also pre-trained), which assigns an anomaly score to each f_i. This score is compared against a predefined threshold thr. If the score does not exceed the thr, it is considered that the features do not provide sufficient evidence of an anomaly, and the corresponding clip $clip_i$ is labeled as normal. Otherwise, if the score exceeds the thr, the features f_i are passed to the anomaly classifier $C(\cdot)$, which determines the specific category of the detected event. In both cases, the system proceeds to the next clip, maintaining a continuous, clip-by-clip processing flow.

3.2. Multi-Category Classifier Module

Since the objective of our anomaly classifier $C(\cdot)$ is to identify the specific type of abnormal event occurring in a clip once it has been detected by the anomaly detector $S(\cdot)$, this section presents the training process of the classification module. It is structured in two parts: first, we describe how the training subdataset is constructed using a weakly supervised paradigm. Then, we present the training procedure.

Our method is designed to work with standard anomaly detection datasets that provide only video-level annotations, where each video is labeled with a single anomaly class but lacks temporal localization. In our case, we focus exclusively on the anomalous videos from the training split of the original dataset. Each of these videos is divided into N consecutive clips that span its full duration. Each clip is then processed to extract a spatiotemporal feature representation. Based on these features, we select those with the highest evidence of abnormality, under the assumption that at least one of them reflects the anomaly indicated by the video-level label. The selected features are collected into a subdataset $\mathcal{R}_{\text{anom}}$, which includes samples from videos of all classes in the original training set. This process is illustrated in Figure 2.

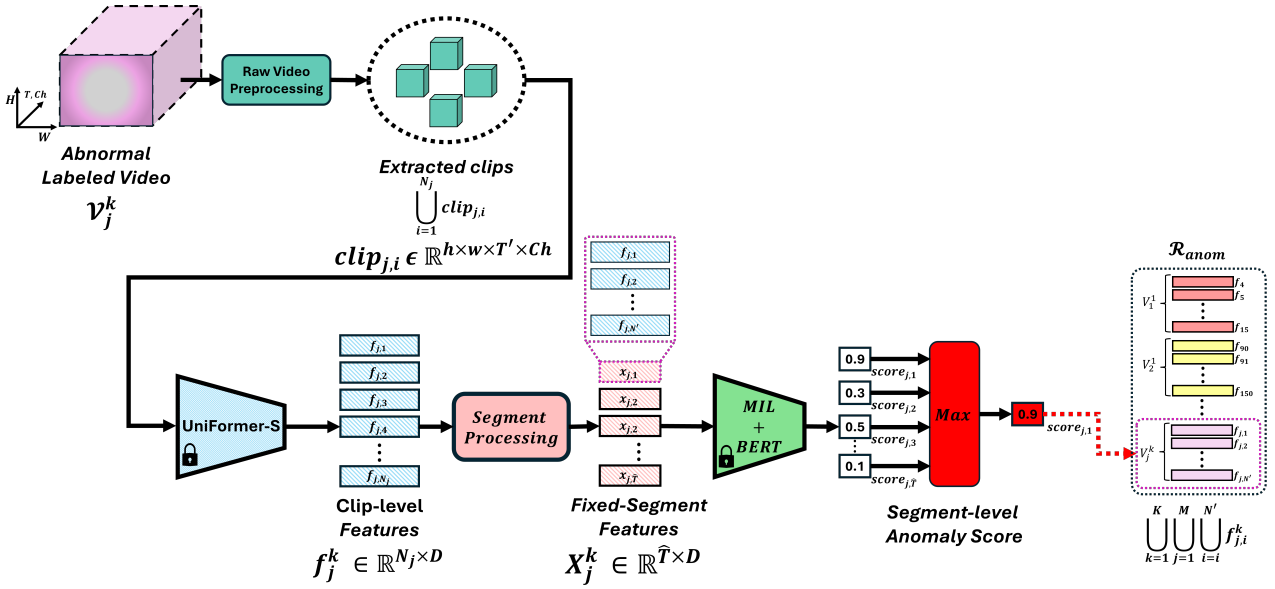

Figure 2. Flowchart for creating the multiclass subdataset $\mathcal{R}_{\text{anom}}$. A video labeled as abnormal from a k class (V_j^k) is first divided into consecutive clips, resized and center cropped, then each clip is processed by the UniFormer-S extractor to obtain clip-level features f_j^k. These features are segmented into fixed-length representations X_j^k, which are then evaluated by the MIL + BERT detector to assign segment-level anomaly scores. The most anomalous segment is identified, and its constituent clips form the set $\mathcal{R}_{\text{anom}}$., which is used to train the anomaly classifier $C(\cdot)$. The lock icon indicates that the corresponding model operates with *frozen* weights, meaning no fine-tuning or further training is performed. Pink dotted boxes enclose the set of feature vectors $\{f_{j,1}, \ldots, f_{j,N'}\}$, which are used to compute the fixed segment $x_{j,t} \in \mathbb{R}^{1\times D}$. Red dashed arrows highlight the segment with the highest anomaly score, which is therefore routed to the anomalous subdataset $\mathcal{R}_{\text{anom}}$ for training the anomaly classifier $C(\cdot)$.

This subdataset $\mathcal{R}_{\text{anom}}$ is then used to train the anomaly classification module (Figure 3), using each feature vector along with its associated class label.

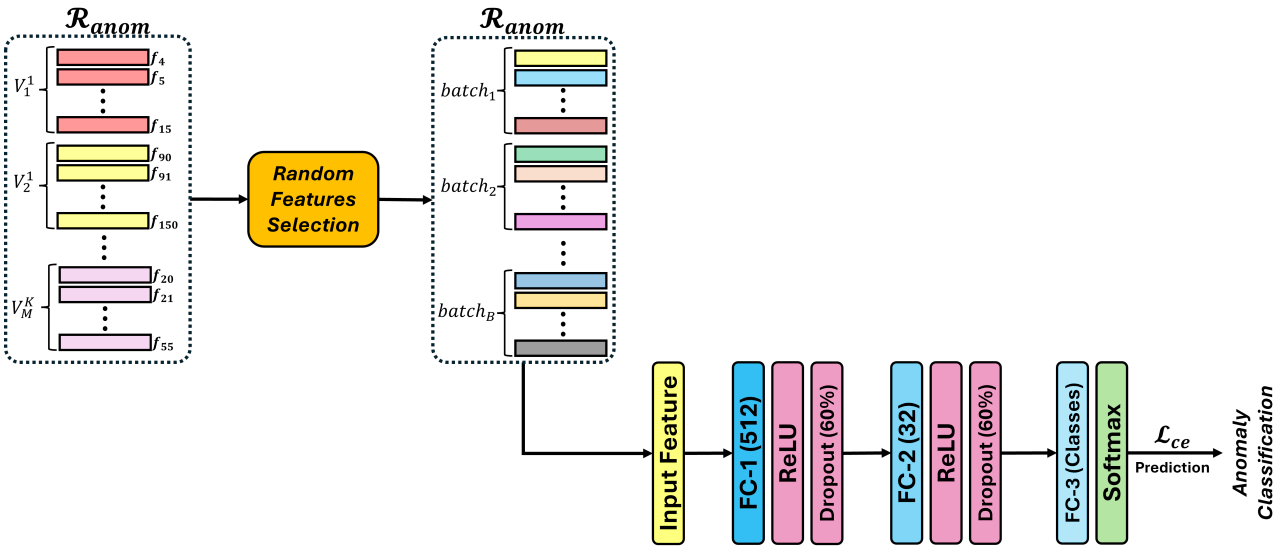

Figure 3. Detailed architecture of our proposed anomaly classifier anomaly classifier $C(\cdot)$.

3.2.1. Multiclass Training Subdataset Generation

1. Data Preparation: To construct the training subdataset $\mathcal{R}_{\text{anom}}$, we first define the complete set of anomalous training videos, denoted as

$$\mathcal{V}_{\text{train}}^{\text{anom}} = \{V_1^1, V_2^1, \ldots, V_{M-1}^K, V_M^K\}, \tag{1}$$

where K represents the total number of anomaly classes, which depends on the dataset used. For instance, in UCF-Crime [8] (one of the most widely used datasets in video anomaly detection, particularly under the weakly supervised paradigm), the training set includes 13 anomaly classes (e.g., $K = 13$). M denotes the number of full-length training videos available for each class, which may vary depending on the dataset distribution (e.g., if class k contains 50 videos, then $M = 50$ for that class). Each element $V_j^k \in \mathcal{V}_{\text{train}}^{\text{anom}}$ corresponds to a whole video labeled as belonging to anomaly class k, composed of T consecutive RGB frames.

2. Raw Video Preprocessing: Subsequently, each video V_j^k in the set $\mathcal{V}_{\text{train}}^{\text{anom}}$ is processed through the raw video preprocessing module, as illustrated in Figure 2. This process begins by resizing each video frame to a standard resolution $H \times W$, followed by a central crop of size $h \times w$. Once all frames are cropped, the video is divided into N_j consecutive and non-overlapping clips, each composed of T' continuous frames, where $N_j = T_j/T'$ and T_j is the total number of frames in video V_j^k. This results in a set of clips represented as

$$V_j^k = \{\text{clip}_{j,1}, \text{clip}_{j,2}, \ldots, \text{clip}_{j,N_j}\}, \quad \text{clip}_{j,i} \in \mathbb{R}^{h \times w \times T' \times Ch} \tag{2}$$

In this expression, $j \in \{1, \ldots, M\}$ indicates the index of the video within the anomaly class k, and $i \in \{1, \ldots, N_j\}$ represents the index of each clip within the video V_j^k. The variable k identifies the anomaly class to which the video belongs, while N_j corresponds to the number of clips extracted from video V_j^k, calculated as $N_j = T_j/T'$ (e.g., for a video with $T_j = 320$ frames and $T' = 16$, $N_j = 20$). Each clip $\text{clip}_{j,i} \in \mathbb{R}^{h \times w \times T' \times Ch}$ is a continuous subsequence of T' RGB frames, cropped and resized to dimensions $h \times w$ (e.g., 224 × 224 pixels), with Ch channels per frame (typically 3 for RGB). These clips serve as the basic input units to the feature extractor for obtaining clip-level feature vectors.

3. Feature Extraction: Each clip is processed through a pre-trained feature extractor $F(\cdot)$. In our case, we use UniFormer-S [17], a spatiotemporal model selected based on the results of our previous work presented at SOMET 2024 [10], which demonstrated a good balance between accuracy, efficiency, and inference speed for independent clip processing.
We define F as the output generated by the fully connected layer, applied after the global average pooling operation and located just before the classification stage in the original UniFormer-S architecture. This output transforms the input clip $\text{clip}_{j,i} \in \mathbb{R}^{h \times w \times T' \times Ch}$ into a low-dimensional abstract representation of its content.
The operation is expressed as

$$f_{j,i} = F(\text{clip}_{j,i}), \quad f_{j,i} \in \mathbb{R}^{1 \times D} \tag{3}$$

where $f_{j,i}$ is the feature vector resulting from the $clip_{j,i}$, and D denotes the dimensionality of this vector (e.g., $D = 512$).
As a result, each video V_j^k is represented by a matrix of N_j feature vectors:

$$f_j^k = \{f_{j,1}, f_{j,2}, \ldots, f_{j,N_j}\}, \quad f_j^k \in \mathbb{R}^{N_j \times D} \tag{4}$$

where j is the index of the set of feature vectors associated with video V_j^k in class k, i is the index of the feature vector corresponding to the i-th processed clip of that video, and N_j is the total number of feature vectors in f_j^k, which corresponds to the same number N_j of clips extracted from V_j^k (e.g., for $N_j = 20$ and $D = 512$, $f_j^k \in \mathbb{R}^{20\times512}$).

4. Segment Processing: Once the feature matrix f_j^k has been obtained, it becomes necessary to identify an efficient way to select the most relevant representations for training the anomaly classifier $C(\cdot)$. A straightforward option would be to feed all the features in f_j^k directly into the classifier. However, this approach is not viable for several reasons. First, anomalies are, by definition, rare and short-lived events, meaning that most clips in an anomalous video contain normal content and thus irrelevant information. Including all these features not only introduces unnecessary noise into the training process but also significantly increases the computational complexity without offering clear benefits.
For this reason, it is essential to filter out the most anomalous features, as they are the most likely to contain patterns specific to each anomaly. A direct solution to this problem is to use a pre-existing anomaly detector, which assigns an anomaly score to each clip. However, this approach presents an additional challenge: determining how to select the most representative clips once these scores are available.
To address this limitation, we first identified the anomaly detector best suited to our scenario. Based on our previous research [10], we determined that the proposal given by Weijun et al. [9] called MIL + BERT is an effective alternative, as it demonstrated excellent performance in combination with the UniFormer-S feature extractor [17] during our experiments. This integration proved particularly robust, optimally leveraging the representations generated by this extractor.
With the detector established, we can outline the strategy for selecting the most anomalous feature vectors. The MIL + BERT approach, originally designed for offline environments, follows the fundamental MIL structure, where the feature vectors in f_j^k are grouped into a fixed number of segments, denoted as $\hat{T}$ (e.g., $\hat{T} = 32$). The resulting segmented matrix is denoted as $X_j^k \in \mathbb{R}^{\hat{T}\times D}$, where $\hat{T}$ is the fixed number of segments into which each video V_j^k is divided, and D is the dimensionality of each feature vector (e.g., $D = 512$). To enable this segmentation, the number of feature vectors N_j must be divisible by $\hat{T}$. However, this condition is rarely met because the number of extracted feature vectors per video varies, which may result in a portion of the video not being analyzed. To address this, we adopt a "rewind" strategy, where the first feature vectors of the video are repeatedly reused until the total number of feature vectors becomes divisible by $\hat{T}$.
After completing this adjustment, the feature vectors in f_j^k (including those from the rewind process) are grouped into $\hat{T}$ segments. Each segment, indexed by $t \in \{1, \ldots, \hat{T}\}$, contains a fixed number of feature vectors denoted as N', where N' is computed as $N_j/\hat{T}$. The change of index from i (feature vector index) to t (segment index) reflects the transition from processing individual feature vectors to processing groups of feature vectors aggregated into fixed-length segments.
For each segment t, the element-wise mean of its N' feature vectors $f_{j,i}$ is computed to obtain a representative segment vector x_t. Each vector $x_t \in \mathbb{R}^{1\times D}$ represents the aggregated features of segment t and is then normalized using the L2 norm to ensure numerical stability and scale consistency. This process is formalized as

$$x_t = \frac{1}{N'} \sum_{i=1}^{N'} f_{j,i}, \quad x_t \leftarrow \frac{x_t}{\|x_t\|_2}, \quad t = 1, \ldots, \hat{T} \tag{5}$$

where t is the segment index, N' is the number of feature vectors per segment, D is the dimensionality of each vector, and j is the video index in class k.
Thus, the final segmented matrix is obtained:

$$X_j^k = \{x_{j,1}, x_{j,2}, \ldots, x_{j,\hat{T}}\}, \quad X_j^k \in \mathbb{R}^{\hat{T} \times D}. \tag{6}$$

where each row $x_{j,t} \in \mathbb{R}^{1 \times D}$ represents the L2-normalized vector obtained from the element-wise mean of the N' feature vectors. This segmented matrix is the direct input to the pre-trained anomaly detector.

5. Selection of Most Anomalous Segments: With the segmented matrix X_j^k obtained, the next step is to feed the entire matrix into the pre-trained anomaly detector $S(\cdot)$, implemented in our case using the MIL+BERT model [9]. The detector processes all segment vectors in X_j^k jointly and outputs a vector of anomaly scores $\text{score}(x_{j,t}) \in [0, 1]$ for $t \in \{1, \ldots, \hat{T}\}$, where each score indicates the degree of anomaly of the corresponding segment. Values close to 1 represent a high probability of anomaly, while values close to 0 indicate that the segment is likely normal. For example, a value of $\text{score}(x_{j,t}) = 0.85$ suggests that the segment contains patterns strongly associated with anomalous behaviors, whereas a value of 0.10 suggests the opposite. Once the score vector is obtained, the index of the most anomalous segment is identified as

$$t_{\max} = \arg \max_{t \in [1, \hat{T}]} \text{score}(x_{j,t}), \tag{7}$$

where $\text{score}(x_{j,t}) \in [0, 1]$ denotes the anomaly probability assigned by the detector to segment t, and arg max returns the index of the segment with the maximum score (ties are resolved by selecting the first occurrence, e.g., if the scores are $[0.12, 0.83, 0.45, 0.20, 0.10]$, then $t_{\max} = 2$ since the second segment has the highest anomaly score).
6. Training subdataset Creation: Once $t_{\max}$ is determined, the feature vectors $f_{j,i}$ that constitute this segment are selected. These features, along with their corresponding class label, are added to the set $\mathcal{R}_{\text{anom}}$ to form the subdataset that will be used to train the classifier $C(\cdot)$.

This procedure is repeated for each video in the set $\mathcal{V}_{\text{train}}^{\text{anom}}$ to complete the construction of the training subdataset.

It is worth emphasizing that all the steps described above (including the use of the MIL+BERT detector, fixed segmentation into $\hat{T}$ parts, and the selection of the most anomalous segment) are applied exclusively during the training phase of the multiclass classification module $\mathcal{C}(\cdot)$. These operations are necessary to build a representative subdataset of anomaly-specific feature vectors under a weakly supervised learning setting, in which temporal annotations are not available. Once trained, the classifier is integrated into the framework modular pipeline and used during inference in an online mode, classifying each clip individually only when it has been previously identified as anomalous, as described in Section 3.1.

3.2.2. Training of the Multiclass Classifier

Once $\mathcal{R}_{\text{anom}}$ has been identified, the training of the multiclass classification module $\mathcal{C}(\cdot)$ is carried out. This classifier is responsible for assigning the appropriate anomaly category to each feature vector f_i, which represents a segment extracted from a video clip and previously encoded by the feature extractor $F(\cdot)$ (e.g., UniFormer-S).

As illustrated in Figure 3, the classification module is implemented as a fully connected neural network with three linear layers: an input layer with 512 units, a hidden layer with

32 units, and an output layer whose number of units corresponds to the total number of anomaly classes. ReLU activation functions are applied between layers, and a dropout layer with a rate of 60% is used after each hidden layer to reduce overfitting. The output layer uses a Softmax function to yield a probability distribution over the classes.

Training is performed using batches of fixed size, composed of randomly sampled feature vectors from the set $\mathcal{R}_{\text{anom}}$, in order to prevent order bias and improve generalization. The optimization uses the cross-entropy loss, defined as

$$\mathcal{L}_{\text{ce}} = -\sum_{k=1}^{K} y_k \log(\hat{y}_k) \tag{8}$$

where y_k is the true label and $\hat{y}_k$ is the predicted probability for class k. This function measures the dissimilarity between the predicted probability distribution and the true label distribution. Its optimal value is 0, achieved when the predicted probability for the correct class is 1, while its theoretical maximum tends to infinity when the predicted probability for the correct class approaches 0.

4. Results

4.1. Database

The dataset used in this research is UCF-Crime [8], proposed by Sultani et al. [8]. This dataset contains 1900 videos divided into two subsets: 1610 training videos (810 anomalous and 800 normal) labeled at the video level, and 290 testing videos (140 anomalous and 150 normal) labeled at the frame level. The videos cover 13 categories of anomalies, including abuse, assault, arrest, arson, burglary, explosion, fighting, robbery, accident, shooting, stealing, vandalism, and road accident. All videos have a resolution of 240 $\times$ 320 pixels and a frame rate of 30 fps.

For the training of the anomaly classifier $C(\cdot)$, only the anomalous videos from the training set were used, organized according to their class. On the other hand, the evaluation was carried out on the complete UCF-Crime testing set, including both normal and anomalous videos.

4.2. Metrics

4.2.1. General Evaluation Metrics

To evaluate the proposed framework, the main metric used was video-level accuracy, following the criteria established by previous works on anomaly classification [14–16]. This metric is computed over the entire test set and allows for direct comparison with the state of the art.

Additionally, metrics such as accuracy, precision, recall, and F1-score are computed at the clip level. While the overall analysis remains consistent with previous studies, our approach aims to operate at a finer granularity, reducing the time required to identify events. These metrics therefore help assess how effective the system is at classifying individual clips rather than entire videos. They are defined as

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{10}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{11}$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{12}$$

where TP (true positives) are clips labeled as anomalous in the ground truth and correctly predicted as anomalous; TN (true negatives) are clips labeled as normal and correctly predicted as normal; FP (false positives) are clips labeled as normal but incorrectly predicted as anomalous; and FN (false negatives) are clips labeled as anomalous but incorrectly predicted as normal.

The Area Under the ROC Curve (AUC) is also calculated at the frame level, as is common in detection tasks [8,9]. However, in this case, it is used solely for comparative purposes within the internal evaluations of the framework (see Section 4.5). The goal is to analyze how the detection capability of each model directly influences the classifier's accuracy, given that the proposed component is not a detector but rather a classifier that relies on the output of various existing detectors.

4.2.2. Evaluation Strategy for Anomalous Clip Classification

To evaluate the performance of the anomaly classification component, a ground truth at the clip level must be constructed, as the UCF-Crime dataset [8] provides only frame-level annotations indicating the start and end of anomalous events.

Following the proposed methodology, each anomalous video in the test set is segmented into consecutive, non-overlapping clips of 16 frames. A clip is labeled as anomalous if at least one of its frames falls within an annotated anomalous segment. This labeling strategy accounts for the gradual onset and progression of many anomalies, where even a single labeled frame may represent a meaningful portion of the event.

Only the clips labeled as anomalous according to this criterion are considered for evaluation. Normal clips present in anomalous videos are excluded, as the objective is not to detect the presence of an anomaly but to classify its type after detection has occurred.

To calculate video-level metrics, a majority voting strategy is used: the most frequently predicted anomaly class among the classified clips of a video is selected as the final label for that video. This approach is consistent with existing works and enables reliable metric computation at both the clip and video levels.

Table 1 presents the resulting number of clips per anomaly class, divided into training and testing sets. This distribution provides a clear reference for the data volume used in the evaluation process.

Table 1. Distribution of anomalous clips used for training and testing, grouped by class.

Class	Training	Testing
Abuse	395	13
Arrest	541	495
Arson	497	521
Assault	222	536
Burglary	813	1019
Explosion	381	502
Fighting	505	282
RoadAccidents	520	162
Robbery	882	243
Shooting	151	637
Shoplifting	503	497
Stealing	919	382
Vandalism	285	136
Total	6614	5425

4.2.3. Evaluation Strategy for the Integrated Detector and Classifier

The evaluation of the integrated anomaly detector and multiclass classifier is conducted using the full test set of the UCF-Crime dataset [8].

At the video level, the final prediction is derived through a majority voting strategy. If at least one clip in a video exceeds the anomaly detection threshold, the video is considered anomalous, and the anomaly classifier $C(\cdot)$ is applied only to those selected clips. The most frequent predicted class among these clips is assigned as the video's final label. If no clip exceeds the threshold, the video is labeled as normal. This prediction is then compared against the ground-truth label of the video to compute global performance metrics.

At the clip level, each clip is evaluated independently. If its anomaly score exceeds the threshold, one of the thirteen anomaly categories is assigned using the classifier. Otherwise, the clip is labeled as normal. Each predicted label is then compared with the corresponding ground truth, enabling an assessment of classification accuracy at the clip level.

4.3. Implementation Details

For all experiments in this research, the UniFormer-S feature extractor was used, configured to process clips composed of $T' = 16$ consecutive, non-overlapping frames. Each frame, originally composed of $C_h = 3$ RGB channels, is resized to a resolution of $H \times W = 240 \times 320$ pixels before processing. From each clip, crops of fixed size $h \times w = 224 \times 224$ pixels are extracted. The features produced by UniFormer-S for each clip are vectors of dimension $D = 512$. For the subsequent segmentation process, the number of segments was fixed to $\hat{T} = 32$ for all videos. As defined in the UCF-Crime dataset, the anomaly classes K include 13 predefined categories.

The anomaly classifier $C(\cdot)$ was trained with a batch size of 128, the AdamW optimizer, a learning rate of 0.001, and a total of 100 epochs. The number of training steps per epoch was denoted as B, which corresponds to the number of batches required to iterate over the entire training set once. Since the UCF-Crime dataset does not include an explicit validation split, 10% of the training set was reserved for validation, following common practices in previous studies. To address class imbalance, a weighted cross-entropy loss function was employed, assigning greater weight to underrepresented classes.

For the integration of the full modular framework, four anomaly detectors $\mathcal{S}(\cdot)$ were used: MIL, Modified MIL, MIL + BERT, and C2FPL. MIL, Modified MIL, and MIL + BERT operate using only the central crop per clip, while C2FPL uses all ten crops per clip.

The anomaly detection threshold *thr* (used to decide whether a clip is anomalous or not) for each $\mathcal{S}(\cdot)$ was selected using the ROC curve computed on the test set. Specifically, the threshold chosen corresponds to the midpoint of the list of thresholds generated by Scikit-learn's internal ROC computation. This criterion ensures consistency across all detectors and avoids manual tuning.

The entire implementation was developed using PyTorch 2.4.1 on a machine equipped with an Intel Core i7-13700F processor, 32 GB of RAM, and an Nvidia GeForce RTX 4060 Ti GPU with 16 GB of memory. To promote reproducible research, our implementation is publicly available on GitHub [27].

For clarity, a summary of the main variables and their values is provided in Table A1 in Appendix A.

4.4. Evaluation and Selection of the Best Multiclass Classifier

To determine the most effective model for categorizing anomalous events, a comparative evaluation was conducted among four classifiers: a fully connected neural network, K-Nearest Neighbors (KNN), XGBoost (XGB), and Support Vector Machines (SVM). All

models were trained under the same conditions using the features described previously, ensuring a fair comparison.

FC-Proposed (detailed in Section 3.2.1) achieved the highest performance, with an accuracy of 33.47% at the clip level and 39.28% at the video level (Table 2). SVM followed with 30.52% and 36.42%, respectively, while XGBoost and KNN showed lower results, with 28.25%/31.42% and 25.34%/26.42%. These metrics confirm the superior performance of the FC-Proposed across both evaluation levels.

Table 2. Comparison of classification accuracy for anomaly type recognition using different multiclass classifiers.

Classifier	Clip-Level Accuracy (%)	Video-Level Accuracy (%)
KNN	25.34	26.42
XGBoost [28]	28.25	31.42
SVM (Linear Kernel)	30.52	36.42
FC-Proposed	**33.47**	**39.28**

Bold values indicate the highest accuracy in each column. The FC-Proposed classifier corresponds to the fully connected neural network developed as part of the proposed framework.

To better understand the behavior of the two best-performing classifiers, confusion matrices were analyzed (Figure 4), along with the distribution of samples per class (Table 2). The neural network outperformed SVM in 7 out of 13 categories, particularly in classes such as Explosion, Fighting, Shoplifting, and Stealing, where high intra-class variability and abundant data likely enhanced its generalization capabilities. In contrast, SVM demonstrated strength in more structured or less represented classes like Arson, Assault, Burglary, and Shooting, highlighting its effectiveness in settings with clearer decision boundaries and limited training samples.

One noteworthy exception was the Abuse category, where no classifier produced correct predictions. Despite having 395 training clips, only 13 were available in the test set, and the visual and contextual variability of this class likely hindered consistent pattern recognition.

At the video level, majority voting helped smooth out inconsistencies in clip-level predictions, reducing performance gaps across models. However, the FC-Proposed maintained its advantage, especially in classes with high variability. In contrast, SVM continued to perform competitively in more homogeneous categories. Interestingly, both models achieved equal performance on the Vandalism class.

Considering both quantitative results and practical aspects, the FC-Proposed was selected as the final classifier in the proposed framework. Its superior accuracy, seamless integration into deep learning pipelines, and compatibility with GPU-accelerated hardware make it an ideal choice for deployment in scalable and adaptive anomaly classification systems.

4.5. Joint Evaluation of Detection and Classification Modules

Once the effectiveness of the proposed multiclass classifier was validated, it was integrated into the full framework to evaluate its performance when paired with different anomaly detectors. The goal of this phase is to verify that the system maintains its classification capabilities when connected to various detection modules, thus reinforcing its modular nature and applicability in both offline and online schemes.

Four detectors were selected for this evaluation: MIL [8], Modified MIL [9], MIL + BERT [9], and C2FPL [18]. The first three are based on the Multiple Instance Learning (MIL) paradigm. MIL and MIL + BERTL follow existing configurations from the literature, while Modified MIL refers to a variant of MIL + BERT in which the BERT component is used

only during training to guide the MIL optimization, but removed during inference. This intermediate configuration allows the classifier's behavior to be observed with a detector trained under stronger supervision, but operating as a pure MIL during testing. On the other hand, C2FPL was selected for its suitability in online environments, aligning with the long-term goal of enabling continuous, frame-by-frame video analysis.

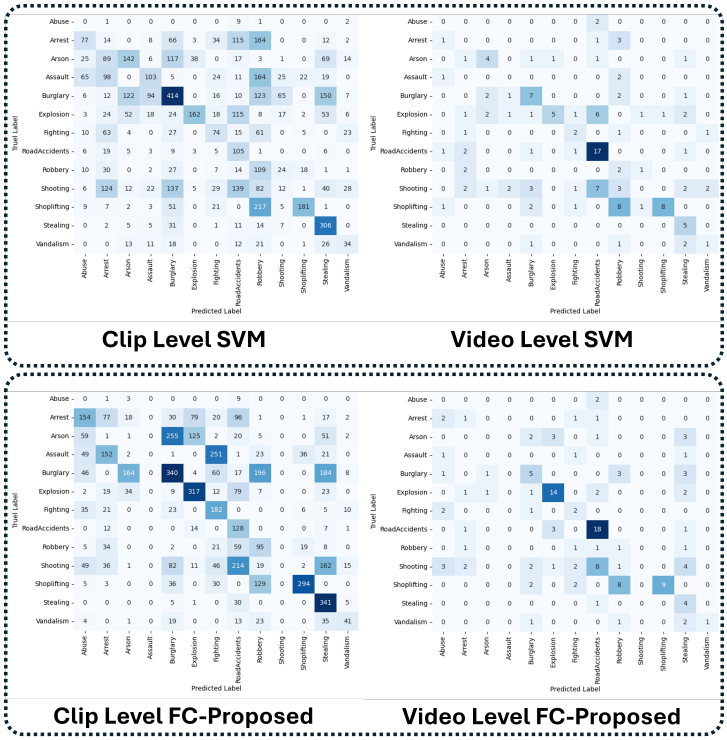

Figure 4. Confusion matrices of the best-performing classifiers. **Top row:** SVM. **Bottom row:** proposed fully connected classifier. **Left:** clip-level evaluation. **Right:** video-level evaluation. Diagonal values indicate correct predictions.

The results are presented in Table 3. The system demonstrated consistent behavior when integrated with all four evaluated detectors, confirming the flexibility of the proposed framework. A direct correlation was observed between the quality of the detector (measured by AUC) and the classification performance by anomaly type.

Table 3. Performance of the full framework using different state-of-the-art anomaly detectors.

Detector	Mode	Detection Metric (%)	Classification Metrics at Clip Level (%)				Classification Metric at Video-Level (%)
		AUC	Precision	Recall	F1-Score	Accuracy	Accuracy
MIL [8]	Offline	72.43	37.21	32.99	31.59	51.00	31.72
Modified MIL [9]	Offline	76.47	43.19	39.90	37.99	53.07	40.68
MIL + BERT [9]	Offline	79.74	48.75	47.74	45.40	53.99	52.41
C2FPL [18]	Online	**82.27**	**63.41**	**64.10**	**62.54**	**74.77**	**58.96**

Bold values indicate the best performance in each column.

The C2FPL detector, with an AUC of 82.27%, achieved the best overall performance. At the clip level, it reached a precision of 63.41%, recall of 64.10%, F1-score of 62.54%, and accuracy of 74.77%. At the video level, it achieved an accuracy of 58.96%, the highest among all methods. This suggests that more accurate detection enables the classifier to operate on more representative clips, thereby improving the overall system performance.

MIL + BERT (79.74% of AUC) and Modified MIL (76.47% of AUC) also yielded competitive results, clearly outperforming the original MIL. The latter, with an AUC of 72.43%, was the weakest performer, achieving only 31.72% video-level accuracy, which highlights how weak detection directly impacts subsequent classification.

Beyond the obtained values, this analysis highlights a key strength of the framework: its ability to integrate with various detectors without requiring structural changes. This adaptability was especially evident with C2FPL, a detector designed for online operation. Its smooth integration demonstrates that our system is well suited for continuous operation scenarios where decisions must be made as data is received.

Thus, in addition to validating the robustness of the multiclass classifier, the results confirm that the proposed framework can scale toward real-world implementations where detection and classification must work jointly and efficiently without needing to process the entire video before making decisions.

4.6. Comparison with State-of-the-Art

This section analyzes the efficiency of the proposed framework in comparison with previous methods that address the problem of anomaly-type classification. To this end, we selected methods that simultaneously meet the following three criteria: (i) they perform classification of anomalous events, (ii) they use the same dataset (UCF-Crime), and (iii) they report metrics that are compatible with our evaluation protocol, which follows the conventions adopted in most works addressing classification on UCF-Crime.

As shown in Table 4, the model proposed by Sultani et al. [8] includes two versions. The first, and more basic, version uses features extracted with C3D and additionally generates a global summary of each video using the L2 norm. The problem is then approached as a conventional multiclass classification task using a Nearest Neighbor scheme, achieving an accuracy of 23.00%. The improved version maintains the same structure but uses features processed by TCNN, reaching an accuracy of 28.40%. These results validate the effectiveness of the approach presented in this work, even in comparison with improved configurations of classical methods.

Table 4. Comparison of video-level accuracy between state-of-the-art methods and the proposed framework.

Method	Video-Level Accuracy (%)
Sultani et al. (C3D) [8]	23.00
Sultani et al. (TCNN) [8]	28.40
Wu et al. [15] *	41.43
Mumtaz et al. [29]	47.00
Li et al. [30] *	47.14
Ganagavalli et al. [31]	47.70
Our framework	**58.96**

* Evaluated only on anomalous events without including a 'Normal' class. Bold values indicate the best performance.

Wu et al. [15] and Li et al. [30] employ CLIP-based architectures that align visual features with semantic representations to enhance anomaly recognition. Both approaches evaluate classification performance only on anomalous events, excluding the Normal class, which inflates performance in practical deployments. Wu's method achieves 41.43% accuracy, while Li's approach reaches 47.14%, benefiting from a richer prompt-based representation. In contrast, our framework not only outperforms both by more than 11 and 17 percentage points, respectively, but also preserves this performance when normal events are included, which is essential for real-world applicability.

Mumtaz et al. [29] propose a 3D CNN with multiple Inception blocks for direct classification into the 14 categories of the UCF-Crime dataset, obtaining 47.00% video-level

accuracy. While effective at capturing multi-scale spatiotemporal patterns, their design couples detection and classification into a single stage, limiting adaptability to alternative detection schemes. Our modular approach decouples these stages, allowing flexible integration with different detectors while improving accuracy by almost 12 percentage points.

Ganagavalli et al. [31] combine YOLO-based object detection with activity recognition to classify 13 types of criminal activities, achieving 47.70% video-level accuracy. This dependency on object detection makes performance sensitive to occlusions and crowded scenes. In contrast, our framework leverages spatio-temporal clip features, maintaining robustness under such conditions and outperforming this method by more than 11 percentage points.

It is important to note that, unlike Sultani et al.'s [8] approach (which addresses the problem using global video representations), our framework enables more precise identification of both the occurrence and the type of anomalous event without depending on the analysis of the full video. Moreover, unlike the methods of Wu et al. [15] and Li et al. [30], which require complete video processing and omit normal scenarios, our approach operates in a continuous manner across multiple classes, including normal events, while surpassing all reported results in the literature. The modularity of the proposed system also provides an advantage over architectures such as those in Mumtaz et al. [29] and Ganagavalli et al. [31], enabling flexible integration with different detection backbones while maintaining superior accuracy.

5. Discussion

The main objective of this work was to design a modular framework capable of detecting and classifying anomalous events by type, operating clip by clip without requiring full video analysis. To achieve this, a multiclass classifier was developed and integrated into the proposed framework. The classifier, based on fully connected neural networks, achieved the best performance among all evaluated classification methods, with an accuracy of 33.47% at the clip level and 39.28% at the video level when evaluated only on anomalous clips. This demonstrates its ability to learn complex patterns, even in visually similar or imbalanced classes.

Furthermore, when integrated with the C2FPL detector [18], the system reached a video-level accuracy of 58.96% on the full test set, including both normal and anomalous videos, and a clip-level accuracy of 74.77%. These results validate the compatibility of the framework with online detection approaches, reinforcing its online applicability and responsiveness in sequential video processing contexts. This flexibility in integration is one of the key advantages of the proposed modular design.

These results significantly outperform reference methods. On the one hand, the proposed framework outperforms the approach by Sultani et al. [8], which relies on global video representations and classical classification models (C3D+NN: 23.00%, TCNN+NN: 28.40%). On the other hand, it surpasses recent CLIP-based approaches by Wu et al. [15] and Li et al. [30], which achieve 41.43% and 47.14% accuracy, respectively, but only consider anomalous events and exclude the Normal class. Our proposal not only improves these results by more than 17 and 11 percentage points, respectively, but also incorporates normal scenarios in the evaluation, which is essential for practical applications. The ability to handle both normal and anomalous events using the same evaluation setup represents another practical advantage of our system.

These findings confirm that the combination of local detection by clips, type-specific classification, and compatibility with continuous detection makes our framework a more robust and realistic alternative for video anomaly recognition tasks. Each of the evaluated detectors (MIL, Modified MIL, BERT+MIL, and C2FPL) was integrated without requiring

structural changes, further validating the modular design. This confirms the scalability and adaptability of the framework across multiple inference strategies and detection schemes.

6. Conclusions

This work proposed a modular framework for video anomaly detection and classification, focused on achieving a balance between simplicity, effectiveness, and adaptability. Unlike traditional approaches that require analyzing the entire video, our method enables continuous detection and classification of anomalous events from individual clips of just 16 frames, making it highly effective for real-world scenarios where immediate and accurate responses are needed.

One of the main strengths of the framework is its modular design, which allows any of its components (feature extractor, anomaly detector, or classifier) to be independently replaced or improved. This flexibility makes it a sustainable long-term solution, capable of adapting to future technological advances without requiring a complete system redesign.

The proposed classifier, based on a fully connected neural network and specifically trained to operate online, proved to be a powerful option despite its simple structure. Its ability to process clip-by-clip without requiring access to the entire video makes it ideal for integration into systems designed for continuous analysis. When integrated with the C2FPL detector [18], the system achieved a video-level accuracy of 58.98%, clearly outperforming state-of-the-art methods such as those by Sultani et al. [8] and Wu et al. [15], which rely on global video representations and do not include normal classes, thereby limiting their practical applicability.

Finally, although the results obtained are solid, the framework is not limited to the configuration presented. Its structure allows the integration of newer or more advanced detectors and classifiers as required by the operational environment. As future work, we plan to explore its deployment on embedded devices for local implementations, as well as the development of more robust classifiers that retain their online nature, further improving accuracy without sacrificing efficiency.

Author Contributions: Conceptualization, J.F.-M., G.B.-G. and M.N.-M.; methodology, J.F.-M. and M.N.-M.; software, J.F.-M.; validation, G.B.-G. and M.N.-M.; formal analysis, G.B.-G.; investigation, J.F.-M.; data curation, J.F.-M.; writing—original draft preparation, J.F.-M.; writing—review and editing, J.F.-M., G.B.-G. and M.N.-M.; visualization, J.F.-M.; supervision, M.N.-M. and H.T.; project administration, H.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: UCF-Crime: https://www.crcv.ucf.edu/research/real-world-anomaly-detection-in-surveillance-videos/ (accessed on 1 August 2025).

Acknowledgments: The authors thank the Instituto Politecnico Nacional (IPN) as well as the Consejo Nacional de Ciencia y Tecnologia de Mexico (CONACYT) for the support provided during the realization of this research.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

To facilitate understanding of the proposed framework, Table A1 presents a summary of the main variables used throughout this work, along with their respective values when applicable.

Table A1. Summary of the main variables used in the framework.

Variable	Meaning	Value Used
$\mathcal{V}_{\text{train}}^{\text{anom}}$	Set of anomalous training videos	-
V	Input video	-
K	Anomaly classes	-
k	Number of anomaly classes	-
M	Videos per anomaly class	-
V_j^k	Video j of the k class	$H \times W \times T \times Ch$
$H \times W$	Resized frame resolution	240×320
T	Number of frames per video	-
Ch	Number of channels	3 (RGB)
N_j	Number of video clips in a video V_j^k	-
$clip_i$	Individual video clip	$C_r \times h \times w \times T' \times Ch$
C_r	Number of crops per clip	1 to 10
$h \times w$	Cropped frame resolution	224×224
T'	Number of frames per clip	16
f_j^k	All vector features of clips from video V_j^k	$N_j \times D$
f_i	Feature vectors of $clip_i$	$C_r \times D$
D	Feature vector dimension	-
$f_{j,i}$	Feature vector of clip i from video j	-
X_j^k	Segmented feature matrix of video V_j^k	$\hat{T} \times D$
$\hat{T}$	Number of segments per video	32
N'	Feature vectors per Fix-segment	-
$x_{j,t}$	Mean vector of segment t from video V_j^k	$1 \times D$
$t_{\max}$	Index of most anomalous segment	$\arg\max \text{score}(x_{j,t})$
thr	Anomaly detection threshold	Midpoint from ROC
$\mathcal{L}_{\text{ce}}$	Cross-entropy loss function	-
y_k	True class indicator	-
$\hat{y}_k$	Predicted probability for class k	-
B	Steps per epoch (batches per epoch)	-
$F(\cdot)$	Feature Extractor	-
$S(\cdot)$	Anomaly detector	-
$C(\cdot)$	Anomaly classifier	-
$\mathcal{R}_{\text{anom}}$	Training subdataset with selected anomalous clips	-

References

1. Encuesta Nacional de Victimización y Percepción Sobre Seguridad Pública (ENVIPE). 2023. Available online: https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2023/ENVIPE/ENVIPE23.pdf (accessed on 19 November 2023).
2. Newton, A. Crime on Public Transport. In *Encyclopedia of Criminology and Criminal Justice*; Bruinsma, G., Weisburd, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 709–720.
3. Towards Safe and Empowering Streets and Public Transport Systems for Women and Girls in Latin America. Available online: https://womenmobilize.org/towards-safe-and-empowering-streets-and-public-transport-systems-for-women-and-girls-in-latin-america/ (accessed on 11 May 2024).
4. Homel, R. Can police prevent crime. In *Unpeeling Tradition: Contemporary Policing*; Macmillan Education Australia: Melbourne, VIC, Australia, 1994.
5. Presencia Policial es Una Herramienta Efectiva Contra la Delincuencia. Available online: https://www.udep.edu.pe/hoy/2015/10/presencia-policial-es-una-herramienta-efectiva-contra-la-delincuencia/ (accessed on 12 November 2024).
6. The Role of Video Monitoring Services in Crime Prevention. Available online: https://www.gps-securitygroup.com/role-of-video-monitoring-services-in-crime-prevention/ (accessed on 12 December 2024).
7. Nobili, G.G. Los sistemas de vídeovigilancia para prevenir la delincuencia: Lecciones aprendidas. *Constr. Criminol.* **2021**, *1*, 97–110.
8. Sultani, W.; Chen, C.; Shah, M. Real-world anomaly detection in surveillance videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6479–6488.
9. Tan, W.; Yao, Q.; Liu, J. Overlooked video classification in weakly supervised video anomaly detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW), Waikoloa, HI, USA, 1–6 January 2024; pp. 202–210.

10. Flores-Monroy, J.; Benitez-Garcia, G.; Nakano, M.; Takahashi, H. Optimal Feature Extractor for Video Anomaly Detection in Public Transportation Applications. *New Trends Intell. Softw. Methodol. Tools Tech.* **2024**, 249–262.
11. Zhong, J.-X.; Li, N.; Kong, W.; Liu, S.; Li, T.H.; Li, G. 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, Long Beach, CA, USA, 15–20 June 2019; pp. 1237–1246.
12. Tian, Y.; Pang, G.; Chen, Y.; Singh, R.; Verjans, J.W.; Carneiro, G. Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 4975–4986.
13. Karim, H.; Doshi, K.; Yilmaz, Y. Real-time weakly supervised video anomaly detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2024; pp. 6848–6856.
14. Majhi, S.; Das, S.; Brémond, F.; Dash, R.; Sa, P.K. Weakly-supervised joint anomaly detection and classification. In Proceedings of the 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), Jodhpur, India, 15–18 December 2021; pp. 1–7.
15. Wu, P.; Zhou, X.; Pang, G.; Sun, Y.; Liu, J.; Wang, P.; Zhang, Y. Open-vocabulary video anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 18297–18307.
16. Ullah, W.; Ullah, A.; Hussain, T.; Khan, Z.A.; Baik, S.W. An efficient anomaly recognition framework using an attention residual LSTM in surveillance videos. *Sensors* **2021**, *21*, 2811. [CrossRef] [PubMed]
17. Li, K.; Wang, Y.; Gao, P.; Song, G.; Liu, Y.; Li, H.; Qiao, Y. UniFormer: Unified transformer for efficient spatiotemporal representation learning. *arXiv* **2022**, arXiv:2201.04676. [CrossRef]
18. Al-Lahham, A.; Tastan, N.; Zaheer, M.Z.; Nandakumar, K. A coarse-to-fine pseudo-labeling (c2fpl) framework for unsupervised video anomaly detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2024; pp. 6793–6802.
19. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
20. Carreira, J.; Zisserman, A. Quo vadis, action recognition? A new model and the kinetics dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
21. Lin, J.; Gan, C.; Han, S. TSM: Temporal shift module for efficient video understanding. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7083–7093.
22. Öztürk HI, Can AB. Adnet: Temporal anomaly detection in surveillance videos. In *Pattern Recognition, Proceedings of the ICPR International Workshops and Challenges, Virtual Event, 10–15 January 2021*; Proceedings, Part IV; Springer: Berlin/Heidelberg, Germany, 2021; pp. 88–101.
23. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the 38th International Conference on Machine Learning (ICML 2021), Virtual Event, 18–24 July 2021; Meila, M., Zhang, T., Eds.; Proceedings of Machine Learning Research; PMLR: Online, 2021; Volume 139, pp. 8748–8763. Available online: https://proceedings.mlr.press/v139/radford21a.html (accessed on 13 August 2025).
24. Devlin, J. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
25. Hou, R.; Chen, C.; Shah, M. Tube convolutional neural network (T-CNN) for action detection in videos. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5822–5831.
26. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
27. Released Our Code. Available online: https://github.com/JonathanFlores2503/TransLowNet (accessed on 13 August 2025).
28. Chen, T.; He, T.; Benesty, M.; Khotilovich, V.; Tang, Y.; Cho, H.; Chen, K.; Mitchell, R.; Cano, I.; Zhou, T.; et al. Xgboost: Extreme Gradient Boosting. R Package Version 0.4-2, 2015. Available online: https://cran.r-project.org/web/packages/xgboost/index.html (accessed on 13 August 2025).
29. Mumtaz, A.; Sargano, A.B.; Habib, Z. Robust learning for real-world anomalies in surveillance videos. *Multimed. Tools Appl.* **2023**, *82*, 20303–20322. [CrossRef]

30. Li, F.; Liu, W.; Chen, J.; Zhang, R.; Wang, Y.; Zhong, X.; Wang, Z. Anomize: Better Open Vocabulary Video Anomaly Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 10–17 June 2025; pp. 29203–29212.
31. Ganagavalli, K.; Santhi, V. YOLO-based anomaly activity detection system for human behavior analysis and crime mitigation. *Signal Image Video Process.* **2024**, *18* (Suppl. S1), 417–427. [CrossRef]