sensors

MDPI

Article

CNN-ViT Supported Weakly-Supervised Video Segment Level Anomaly Detection

Md. Haidar Sharif *, Lei Jiao and Christian W. Omlin

Department of ICT, University of Agder, 4630 Kristiansand, Norway; lei.jiao@uia.no (L.J.); christian.omlin@uia.no (C.W.O.)
* Correspondence: md.h.sharif@uia.no

Abstract: Video anomaly event detection (VAED) is one of the key technologies in computer vision for smart surveillance systems. With the advent of deep learning, contemporary advances in VAED have achieved substantial success. Recently, weakly supervised VAED (WVAED) has become a popular VAED technical route of research. WVAED methods do not depend on a supplementary self-supervised substitute task, yet they can assess anomaly scores straightway. However, the performance of WVAED methods depends on pretrained feature extractors. In this paper, we first address taking advantage of two pretrained feature extractors for CNN (e.g., C3D and I3D) and ViT (e.g., CLIP), for effectively extracting discerning representations. We then consider long-range and short-range temporal dependencies and put forward video snippets of interest by leveraging our proposed temporal self-attention network (TSAN). We design a multiple instance learning (MIL)-based generalized architecture named CNN-ViT-TSAN, by using CNN- and/or ViT-extracted features and TSAN to specify a series of models for the WVAED problem. Experimental results on publicly available popular crowd datasets demonstrated the effectiveness of our CNN-ViT-TSAN.

Keywords: attention; convolutional neural network (CNN); Mahalanobis distance; multiple instance learning (MIL); vision transformer (ViT); weakly supervised video anomaly event detection

Citation: Sharif, M.H.; Jiao, L.; Omlin, C.W. CNN-ViT Supported Weakly-Supervised Video Segment Level Anomaly Detection. *Sensors* **2023**, *23*, 7734. https://doi.org/10.3390/s23187734

Academic Editor: Amir Atapour-Abarghouei

Received: 1 August 2023
Revised: 1 September 2023
Accepted: 4 September 2023
Published: 7 September 2023

1. Introduction

Fully supervised, unsupervised, and weakly supervised are the three dominant paradigms in video anomaly event detection (VAED). The fully supervised paradigm mostly gives a high performance [1]. Nevertheless, frame-level normal or abnormal annotations in the training data are essential, which requires the video annotators to localize and label abnormalities in videos. As abnormalities can take place at any time, nearly all frames need to be spotted by the annotators. Unfortunately, it can be a non-automated and time-consuming process to accumulate a fully annotated large-scale dataset for supervised VAED.In the unsupervised paradigm, the models are trained on samples of normal events solely, along with a common assumption that the unseen anomaly videos will have high reconstruction errors [2–4]. Unluckily, the performance of unsupervised VAED is commonly inferior, due to its lack of advance understanding of anomalies, as well as its inability to capture all kinds of normality variants [5]. The weakly supervised approaches are thus considered to be the most practical paradigm, over both unsupervised and supervised paradigms, due to their competitive performance as well as annotation cost-effectiveness, by applying video-level labels to lower the cost of laborious fine-grained annotations [6,7].

Nowadays, WVAED has become an established VAED technical route of research [6–16]. The WVAED problem is mainly regarded as an MIL (multiple instance learning) problem [8]. In general, WVAED models directly output anomaly scores by comparing the spatiotemporal features of normal and abnormal events through the MIL. The MIL pertains to training data organized in sets, called positive and negative bags. A video in MIL is regarded as a bag holding multiple instances, where each instance belongs to a video snippet.

A negative bag contains all normal snippets, whereas a positive one contains both normal and abnormal snippets, without any temporal information about the beginning and end of abnormal events. The standard MIL assumes that all negative bags accommodate only negative snippets, and that positive bags carry no less than one positive snippet. Supervision is provided solely for complete sets, and the isolated label of the snippets contained in the bags is not provided [17]. As WVAED can understand the essential variability between normal and abnormal, its outputs are fundamentally more reliable than those of unsupervised VAED [18]. However, in WVAED, abnormal-labeled frames of the positive bag tend to be influenced by normal-labeled frames in the negative bag, while the abnormality will not certainly stand out in opposition to the normality. Subsequently, sometimes it becomes difficult to detect anomalous snippets. Many researchers (e.g., [8–10,19,20]) have made efforts to take this problem forward using MIL frameworks. Many of the existing approaches encode the extracted visual content by applying a backbone (e.g., C3D [21], I3D [22]), which are pretrained on action recognition tasks. However, VAD depends on discriminative representations that clearly represent the events in a scene. Thus, these existing backbones are not suitable for VAD, due to the domain gap [1]. To address this limitation, and inspired by the success of the recent vision-language works of [23–25], which proved the potency of feature representation learned via contrastive language-image pretraining (CLIP) [26], Joo et al. [20] employed the vision transformer (ViT) encoded visual features from CLIP [26]. However, the performance of MIL-based WVAED methods heavily depends on the pretrained feature extractors.

In this paper, we first propose utilizing pretrained feature extractors using backbones of both CNN (e.g., C3D [21], I3D [22]) and ViT (e.g., CLIP [26]) for extracting discerning representations effectively. We propose a temporal self-attention network (TSAN) to generate the reweighed attention feature by modeling the continuity between snippets of a video and selecting the top-k most relevant snippets. Later, the reweighed attention features are used to produce anomaly scores using a multi-layer perceptron (MLP) based score allocator. In the TSAN pipeline, we utilize the statistically most significant features as probabilities by employing a temporal scoring technique considering Mahalanobis distances instead of the mean feature magnitudes of snippets. The motivations behind the usage of the Mahalanobis metric over the mean are as follows: (i) It can correct the correlations between the different features; (ii) It automatically accounts for the scaling of the coordinate axes; (iii) It can provide curved as well as linear decision boundaries. Our ablation study showed that maximum mean of 5.34% better performance can be achieved empirically by employing the Mahalanobis metric. In addition, the TSAN also deals with an arbitrary number of abnormal snippets in an abnormal video. The top-k selector in the TSAN addresses k-snippets of interest in the video. We model long-range and short-range temporal dependencies and put forward the snippets of interest by supporting TSAN. In brief, we design a MIL-based generalized architecture of CNN-ViT-TSAN, as portrayed in Figure 1, to specialize five different models, namely C3D-TSAN, I3D-TSAN, CLIP-TSAN, C3D-CLIP-TSAN, and I3D-CLIP-TSAN, for WVAED problems. Each model consists of three main modules responsible for (i) Feature encoding by the CNN and/or ViT; (ii) Patterning snippet consistency in the temporal dimension using TSAN; and (iii) Identifying abnormal snippets in connection with the separation maximization supervisor (SMS), where the SMS trains the abnormal snippets to have a high value and the normal snippets to have a low value. The C3D-TSAN and I3D-TSAN models do not require ViT-based feature extraction, while the CLIP-TSAN model does not need CNN-based feature extraction. Information fusion takes place in the TSAN for C3D-CLIP-TSAN and I3D-CLIP-TSAN models only, whereas the models for C3D-TSAN, I3D-TSAN, and CLIP-TSAN skip it. Each of our proposed models is based on a distinct degree of feature extraction and usability capabilities required for crowd video anomaly detection. Consequently, in experimental setups considering UMN, UCSD-Ped1, UCSD-Ped2, ShanghaiTech, and UCF-Crime datasets, some of these models demonstrated inferior results, while others showed superior results. For example, the model I3D-CLIP-TSAN demonstrated the best results and outperformed its alterna-

tives by extracting and using high-quality features from the available videos, as well as confirming a better normal—abnormal snippet separability.

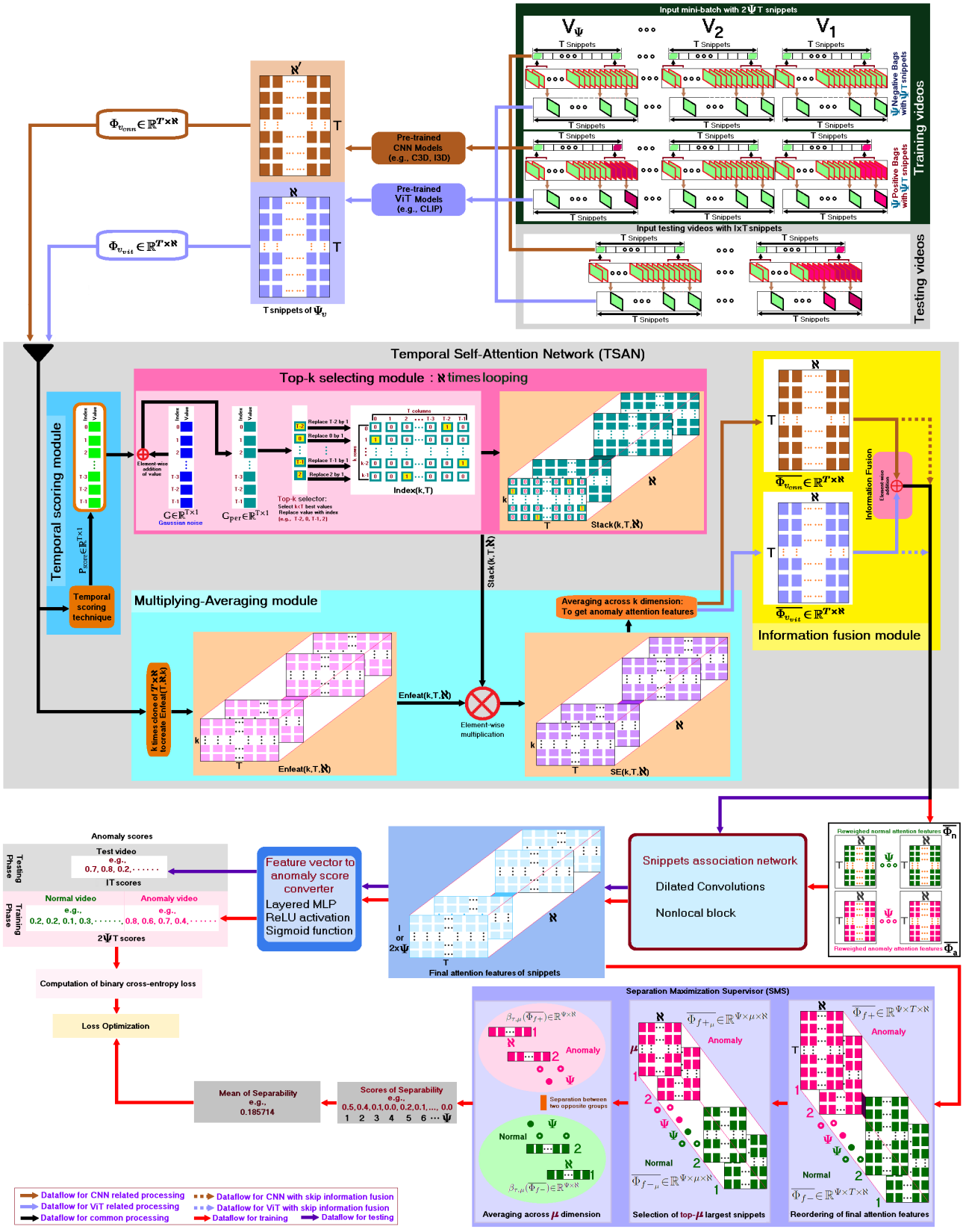


Figure 1. Generalized architecture of our proposed CNN-ViT-TSAN framework.

The unique contributions and advancements that our proposed CNN-ViT-TSAN framework brings to the field of WVAED problems are recapitulated as follows:

- We propose five deep models for WVAED problems by designing a MIL-based generalized framework CNN-ViT-TSAN. The information fusion between CNN and ViT is a unique contribution;
- We propose a TSAN that helps to provide anomaly scores for video snippets in WVAED problems;
- We uniquely introduce the usage of the Mahalanobis metric for calculating probability scores in the TSAN;
- Experiments on several benchmark datasets demonstrated the superiority of our models compared with the state-of-the-art approaches.

The rest of this paper is organized as follows; Section 2 addresses the most relevant previous studies. Section 3 discusses our proposed generalized framework. Section 4 explains the experimental setup; the results obtained on public datasets; as well as a comparison, reasons for superiority, best network analysis, ablation study, and limitations of our models. Section 5 concludes the paper with a few clues for further study.

2. Related Work

Methods of WVAED are based on video-level labels, which always follow the MIL ranking framework [8]. Based on MIL, a method of WVAED trains a regression model to assign scores for video snippets, assuming that the maximum score of the positive bag is higher than that of the negative bag. The existing methods of WVAED can be roughly categorized into two broad kinds on the basis of the pretrained models used, namely: CNN-based and ViT-based WVAED methods, as summarized below.

2.1. CNN-Based WVAED Methods

Sultani et al. [8], Tian et al. [19], Zhang et al. [9], Zhong et al. [6], and Zhu et al. [11] employed CNN-based pretrained models in their experimental setups. Sultani et al. [8] also pre-collected annotated normal and abnormal video events at video-level to build their popular UCF-Crime dataset and applied it with their weakly supervised framework for detecting anomalies. In their framework, after extracting C3D features [27] for video segments, they trained a fully connected neural network by applying a ranking loss function, which computed the ranking loss between the highest scored instances in the positive bag and the negative bag. Tian et al. [19] treated C3D [27] and I3D [22] as feature extractors for their WVAED model. They claimed that the selection of the top-3 features based on their magnitude can introduce a greater partition between normal and anomalous videos, where if more than one abnormal snippet exists per anomalous video, the mean snippet feature magnitude of the anomalous videos is larger than that of normal videos.

Zhang et al. [9] trained a temporal convolution network between the preceding adjacent segment and current segment for extracting positive and negative video segment C3D features [27]. Afterwards, they trained two branches of a fully connected neural network using an inner and outer bag ranking loss, considering the highest and lowest scored segments in the positive and the negative bags. Zhong et al. [6] and Zhu et al. [11] trained both a feature encoder and classifier together. Zhong et al. [6] addressed WVAED as a supervised learning task under noise labels. However, to verify the widespread applicability of their model, they carried out extensive experiments considering a C3D [27] and a temporal segment network [28]. Zhu et al. [11] included the temporal context into their MIL ranking model by applying an attention block. They claimed that features containing motion information extracted by C3D [27] and I3D [22] performed better than features extracted from separate images using VGG16 [29] and Inception [30], regardless of the network depth and feature dimension.

2.2. ViT-Based WVAED Methods

ViT-based pretrained models can be categorized into single-stream or dual-stream types. The single-stream model applies a single transformer to model both image (or video) and text representations in a combined framework, whereas the dual-stream model independently encodes image (or video) and text with a decoupled encoder. Examples of ViT feature extractors include VisualBERT [31], ViLBERT [32], CLIP [26], and data efficient CLIP [33]. Recently, Joo et al. [20] proposed a CLIP-assisted [26] temporal self-attention framework for the WVAED problem. They conducted experiments on publicly available datasets to verify their end-to-end WVAED framework. Li et al. [34] suggested a transformer-based multi-instance learning network to learn video-level anomaly probability and snippet-level anomaly scores. In the inference stage, they employed the video-level anomaly probability to suppress the fluctuation of snippet-level anomaly scores. Lv et al. [35] presented an unbiased MIL scheme that learned an unbiased anomaly classifier and a tailored representation for WVAED.

In view of the existing solutions, we found that, generally, a CNN and ViT are employed separately. To take advantage of both CNN- and ViT-based pretrained models, we designed an MIL-supported generalized architecture named CNN-ViT-TSAN to specify a series of models for the WVAED problem.

3. Proposed Generalized Framework

Our generalized framework follows the MIL model, in which the positive bag represents an anomaly and the negative bag denotes normality. Its constituent components are discussed in the following subsections.

3.1. Feature Extraction

Videos in the training set are only labeled at video-level in WVAED. Assume that a set of weakly labeled training videos $W = \{V_v, y_v\}_{v=1}^{|W|}$ are available, where each video $V_v = \{Frame_i\}_{i=1}^{N_v} \in \mathbb{R}^{N_v \times W \times H}$ hints at a sequence of N_v frames with W pixels for width and H pixels for height. Here, $y_v = \{0, 1\}$ indicates the video-level label of video V_v with respect to anomaly, i.e., it is 1 for an anomaly video that holds at least one abnormal event, otherwise 0. For a video $V_v = \{Frame_i\}_{i=1}^{N_v}$, we divide it into a set of $\{\gamma_i\}_{i=1}^{\lfloor \frac{N_v}{\Delta} \rfloor}$ equal number of non-overlapping temporal snippets each with a length of Δ-frame.

3.1.1. Feature Extraction Using a Pretrained CNN

Convolutional neural networks (CNN), as one of the most representative deep learning models, exhibit great potential in the field of image classification. CNN-based C3D (Convolutional 3D) [21] and I3D [22] are two common feature extractors. As a feature extractor, the C3D is generic, compact, simple, and efficient. Tran et al. [27] showed that C3D can model appearance and motion information simultaneously and outperformed the 2D CNN features in various video-analysis tasks. Carreira et al. [22] introduced a two-stream (i.e., RGB and Flow) Inflated 3D CNN (I3D). Ideally, feature extraction can be efficiently performed by either C3D or I3D. We considered the C3D feature of Ji et al. [21] and I3D feature of Carreira-Zisserman [22]. We computed features of T snippets with feature dimension $\aleph'$ using both C3D and I3D separately. Let $\Phi'_{v_{cnn}} = \{\phi_i\}_{i=1}^{T_v} \in \mathbb{R}^{T_v \times \aleph'}$ be the extracted features of V_v, where T_v belongs to the number of snippets for V_v.

For the dimensionality reduction technique, the principal component analysis (PCA) works under the assumption that the data follow a normal distribution. For this reason, they may be very sensitive to the variance of the variables. In addition, as the extracted data are not normalized, the reduced dimensions using PCA or other similar techniques would give erroneous results. However, the low-variance-filter is an advantageous dimensionality reduction algorithm often used in machine learning on numerical data. Instead of using PCA, we apply the low-variance-filter algorithm to reduce the dimensionality of

the extracted data. Upon dimensionality reduction, $\Phi'_{v_{cnn}} \in \mathbb{R}^{T_v \times \aleph'}$ can obtain the shape of $\Phi_{v_{cnn}} \in \mathbb{R}^{T \times \aleph}$, i.e., $\aleph$-dimensional feature of the T snippets.

3.1.2. Feature Extraction Using a Pretrained ViT

Vision-language pretrained models extract the relationships between objects/actions in a video and objects/actions in text using vision transformers (ViTs). Based on the suitability of applications, various kinds of ViTs exist, e.g., VisualBERT [31], ViLBERT [32], CLIP [26], and data efficient CLIP [33]. In general, CLIP [26] is a multi-modal vision and language model, which utilizes a ViT as a backbone for visual features. We assume that the middle frame $d_j = \lceil \frac{\Delta}{2} \rceil$ represents the snippet γ_j, instead of considering all frames in a snippet γ_j. Following Joo et al. [20], we apply CLIP [26] to the d_j of the snippet γ_j to represent its feature as $\phi_j \in \mathbb{R}^{\aleph}$ with feature dimensions $\aleph$, and then V_v can be constituted as a set of video feature vectors $\Phi_{v_{vit}} = \{\phi_j\}_{j=1}^{T_v} \in \mathbb{R}^{T \times \aleph}$.

3.2. Temporal Self-Attention Network (TSAN)

Figure 1 visualizes our proposed TSAN mechanism, which models the snippet coherency and selects the top-k most significant snippets. It maximizes the attention on a subset of features, while it minimizes attention on noise. The pipeline of TSAN consists of four components namely: (i) a temporal scoring module, (ii) top-k selecting module, (iii) multiplying-averaging module, and (iv) information fusion module.

3.2.1. Temporal Scoring Module

The temporal scoring technique utilizes the statistically most significant features as probabilities, considering Mahalanobis distances instead of the mean feature magnitudes of the snippets. The mathematical exposition is given in Algorithm 1. The scores of $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$ are employed to estimate anomaly attention features, upon extracting the k most significant snippets from the video using Algorithm 2. Concisely, each of $\Phi_{v_{cnn}} \in \mathbb{R}^{T \times \aleph}$ and $\Phi_{v_{vit}} \in \mathbb{R}^{T \times \aleph}$ can be converted into a probability score vector $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$ using Algorithm 1, where each score represents a snippet. The scores of $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$ are fed to the top-k selector module for further processing. The model CLIP-TSAN does not expect the $\Phi_{v_{cnn}} \in \mathbb{R}^{T \times \aleph}$ to be processed using Algorithm 1 to obtain $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$. In this case, the final output $\overline{\Phi_{v_{cnn}}} \in \mathbb{R}^{T \times \aleph}$ of the multiplying-averaging module has no active function in the information fusion module. Thus, solely $\Phi_{v_{vit}} \in \mathbb{R}^{T \times \aleph}$ is processed using Algorithm 1 to obtain $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$ for feeding to the top-k selecting module. Conversely, the model of C3D-TSAN does not look for the $\Phi_{v_{vit}} \in \mathbb{R}^{T \times \aleph}$ to be processed using Algorithm 1 to obtain $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$. In this instance, the final output $\overline{\Phi_{v_{vit}}} \in \mathbb{R}^{T \times \aleph}$ of the multiplying-averaging module has no operational function in the information fusion module. Consequently, only $\Phi_{v_{cnn}} \in \mathbb{R}^{T \times \aleph}$ is processed considering Algorithm 1 to obtain $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$. Likewise, the model I3D-TSAN does not expect the scores of $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$ obtained from $\Phi_{v_{vit}} \in \mathbb{R}^{T \times \aleph}$. However, the models of C3D-CLIP-TSAN and I3D-CLIP-TSAN need the scores of $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$ obtained from both $\Phi_{v_{cnn}} \in \mathbb{R}^{T \times \aleph}$ and $\Phi_{v_{vit}} \in \mathbb{R}^{T \times \aleph}$. They use Algorithm 1 to obtain $\mathbf{P_{score}} \in \mathbb{R}^{T \times 1}$ in a sequential manner, such as in CLIP-TSAN, C3D-TSAN, and/or I3D-TSAN. In the case of either C3D-CLIP-TSAN or I3D-CLIP-TSAN, the final outputs of $\overline{\Phi_{v_{cnn}}} \in \mathbb{R}^{T \times \aleph}$ and $\overline{\Phi_{v_{vit}}} \in \mathbb{R}^{T \times \aleph}$ from the multiplying-averaging module are stored in the information fusion module for element-wise addition.

Algorithm 1: Calculation of the probability scores $\mathbf{P_{score}}$ considering Mahalanobis distances

Input: ⇒ Data matrix: $\mathbf{D} \in \mathbb{R}^{m \times n}$

Output: ⇒ Probability scores: $\mathbf{P_{score}} \in \mathbb{R}^{m \times 1}$

Description: ⇒ $D(m,n)$: two-dimensional (2D) real valued data matrix; m: total number of rows; n: total number of columns; i: column counter variable; j: rows counter variable; c: a counter variable; $D_{mean}(n)$: a 1D array of mean values; $D_{std}(n)$: a 1D array of standard deviations; $Z(m,n)$: normalized matrix of $D(m,n)$; $Z_q(m,n)$: squared values of $Z(m,n)$; $ss(n)$: a 1D array with length of n; $s(n,n)$: a 2D square matrix, $CorMat(n,n)$: a square matrix for correlation, $GaussErrMat(n,n)$: a 2D square Gaussian error matrix with values ranging from 0.000001 to 0.0000000001, det: determinant of $CorMat(n,n)$; $InvCorMat(n,n)$: inverse of $CorMat(n,n)$; δ: degrees of freedom, $\Gamma(.)$: Gamma function.

Define: ⇒ set δ, $i = 1$, $j = 1$, $c = 1$.

for $i \leq n$ **do**
  $D_{mean}(i) = \frac{1}{rows} \sum_{j=1}^{rows} D(j,i)$
  $D_{std}(i) = \sqrt{\frac{\sum_{j=1}^{rows} (D(j,i) - D_m(i))^2}{rows-1}}$
for $j \leq m$ **do**
  for $i \leq n$ **do**
    $Z(j,i) = \frac{D(j,i) - D_{mean}(i)}{D_{std}(i)}$
    $Z_q(j,i) = (Z(j,i))(Z(j,i))$
for $i \leq n$ **do**
  $ss(i) = \sqrt{\frac{\sum_{j=1}^{m} Z_q(j,i)}{m-1}}$
for $i \leq n$ **do**
  for $c \leq n$ **do**
    $s(i,c) = \frac{\sum_{j=1}^{m} Z(j,i) Z(j,c)}{m-1}$
    $CorMat(i,c) = \frac{s(i,c)}{(ss(i))(ss(c))}$
Calculate det
if $det = 0$ **then**
  Generate a GaussErrMat
  $CorMat = CorMat + GaussErrMat$
Calculate InvCorMat
for $j \leq m$ **do**
  /* Calculation of Mahalanobis distance *MahalDist*. */

$$MahalDist(j) = \sqrt{\left[\frac{Z(j,1)\ Z(j,2)\ Z(j,3)\ \ldots\ Z(j,n)}{n}\right][InvCorMat]\begin{bmatrix} Z(j,1) \\ Z(j,2) \\ Z(j,3) \\ \vdots \\ Z(j,n) \end{bmatrix}}$$

  /* Calculation of the probability using the cumulative distribution function of the chi-square distribution, each value of MahalDist, and δ. */

$$P_{score}(j) = 1 - \left[\int_0^{MahalDist(j)} \frac{t^{(\frac{\delta-2}{2})} e^{\frac{-t}{2}}}{2^{\frac{\delta}{2}} \Gamma(\frac{\delta}{2})} dt\right]$$

Algorithm 2: Processing of the probability scores $\mathbf{P_{score}}$ in the TSAN

Input: ⇒ Probability scores: $\mathbf{P_{score}} \in \mathbb{R}^{m\times 1}$.
Output: ⇒ Attention feature matrix: $\overline{\mathbf{D}} \in \mathbb{R}^{m\times n}$.
Description: ⇒ $D(m,n)$: matrix used in Algorithm 1; c_1 and c_2: two counter variables; $G_{noise}(m)$: a 1D array of m random values from a Gaussian distribution with zero mean and any specified value of standard deviation ranging from $2/n$ to $6/n$; $Mix(m)$: an array with length of m; $Top(k)$: an array containing the best k values considering $k < m$; $Index(k,m)$: a 2D integer matrix; $Stack(k,m,n)$: a tensor; $Enfeat(k,m,n)$: a tensor of encoded feature values; $SE(k,m,n)$: a resulting tensor from element-wise multiplication of $Stack(k,m,n)$ and $Enfeat(k,m,n)$; $\overline{D(m,n)}$: reweighed and normalized attention feature matrix.
Define: ⇒ set $c_1 = 1$, $c_2 = 1$.
`/* Top-k selecting module : ℵ times looping. */`
1 **for** $c_1 \leq n$ **do**
2 *Generate the c_1-th $G_{noise}(m)$*
3 **for** $c_2 \leq m$ **do**
4 $Mix(c_2) = P_{score}(c_2) + G_{noise}(c_2)$
5 *Sort the values of Mix(m) along with indices in descending order.*
6 *Clone top-k values from Mix(m) into Top(k).*
7 *Replace the values of Top(k) with their respective indices.*
8 *Fill Index(k,m) with 1 and 0 as hinted in Figure 1*
9 *Create $Stack(k,m,c_1)$ by stacking or concatenating each Index(k,m) in sequence depth-wise along a third axis.*
`/* Multiplying-Averaging module. */`
10 *Create Enfeat(m,n,k) by stacking or concatenating each D(m,n) in sequence depth-wise along a third axis.*
11 *Rearrange Enfeat(k,m,n) by swaping axis of Enfeat(m,n,k).*
12 *Element-wise multiplication of Stack(k,m,n) and Enfeat(k,m,n) to get SE(k,m,n).*
13 *Get reweighted attention feature Element-wise multiplication of Stack(k,m,n) and Enfeat(k,m,n) to get SE(k,m,n).*
14 *Get $\overline{D(m,n)}$ by averaging of SE(k,m,n) across k dimension.*

3.2.2. Top-*k* Selecting Module

This module extracts the $k < T$ most interesting snippets from a video. The value of k is determined using Equations (1) and (2) as

$$k = \left\lfloor \frac{\mu \ \log_2(T)}{\log_e(2)} \ \sin\left(\frac{H}{W}\right) \right\rfloor, \tag{1}$$

$$\mu = \left\lceil \frac{1}{\sqrt{2}} + \frac{\log_2(T)}{2} \right\rceil. \tag{2}$$

A specific Gaussian noise score vector $\mathbf{G} \in \mathbb{R}^{T\times 1}$ is generated to apply to the scores of $\mathbf{P_{score}} \in \mathbb{R}^{T\times 1}$ for producing Gaussian-perturbed scores of $\mathbf{G_{per}} \in \mathbb{R}^{T\times 1}$ using Equation (3) as

$$\mathbf{G_{per}} \in \mathbb{R}^{T\times 1} = \mathbf{P_{score}} \in \mathbb{R}^{T\times 1} \bigoplus \mathbf{G} \in \mathbb{R}^{T\times 1}, \tag{3}$$

where $\bigoplus$ belongs to an element-wise addition. The values of $\mathbf{G_{per}} \in \mathbb{R}^{T\times 1}$ are sorted along with the indices in descending order. Replacement of the k-best values with their respective indices is achieved as shown in Figure 1. For example, if the 1st k-best value has the index of $T-2$, then this 1st k-best value will be replaced by $T-2$. Afterwards, the value of its corresponding 2D matrix's $(T-2)^{th}$ column of the 0th row will be filled with 1, but all other columns of the 0th row will be filled with 0. Similarly, if the 2nd k-best value has an index of 0, then this 2nd k-best value will be replaced by 0. The value of its corresponding 2D matrix's 0th column for the 1st row will be filled with 1 but all other columns of the 1st row will be filled with 0, and so on. However, the aforementioned procedure is repeated $\aleph$ times and the results are stacked or concatenated to obtain a 3D tensor, which is fed to the multiplying-averaging module. Line 1 to Line 9 of Algorithm 2 represent the top-k selecting module.

3.2.3. Multiplying-Averaging Module

Taking into account $\Phi_{v_{cnn}} \in \mathbb{R}^{T\times\aleph}$ or $\Phi_{v_{vit}} \in \mathbb{R}^{T\times\aleph}$, a 3D tensor is created by cloning k times of $T \times \aleph$. The tensor is reshaped to perform an element-wise multiplication with the

output of the 3D tensor of the top-k selecting module. The final product is converted from a 3D tensor to 2D matrix using an averaging technique. Line 10 to Line 14 of Algorithm 2 illustrate the multiplying-averaging module. The output of Algorithm 2 is a reweighed and normalized anomaly attention feature matrix.

3.2.4. Information Fusion Module

This module holds any output of the multiplying-averaging module (i.e., Algorithm 2) for a video V_v. Explicitly, the reweighed and normalized anomaly attention feature matrices of $\overline{\Phi_{v_{cnn}}} \in \mathbb{R}^{T\times\aleph}$ and $\overline{\Phi_{v_{vit}}} \in \mathbb{R}^{T\times\aleph}$ are stored in two memory locations. Based on our five different modeling options, the information of $\overline{\Phi_{v_{cnn}}} \in \mathbb{R}^{T\times\aleph}$ and $\overline{\Phi_{v_{vit}}} \in \mathbb{R}^{T\times\aleph}$ either can or cannot be fused. In the case of the C3D-TSAN, I3D-TSAN, and CLIP-TSAN models, the information fusion of $\overline{\Phi_{v_{cnn}}} \in \mathbb{R}^{T\times\aleph}$ and $\overline{\Phi_{v_{vit}}} \in \mathbb{R}^{T\times\aleph}$ is not required.Conversely, in the case of the C3D-CLIP-TSAN and I3D-CLIP-TSAN models, the information fusion of $\overline{\Phi_{v_{cnn}}} \in \mathbb{R}^{T\times\aleph}$ and $\overline{\Phi_{v_{vit}}} \in \mathbb{R}^{T\times\aleph}$ takes place by considering the mode of element-wise addition.

3.3. Training Phase

In the MIL framework, accurate temporal locations of abnormal events in videos are unspecified. Instead, only video-level labels specifying the existence of an abnormal event in the whole video is needed. A video is called a bag. It is labeled as a positive bag if it holds a minimum of one snippet of an abnormal event, otherwise it is labeled as a negative bag. In the negative bag, none of the snippets contain an abnormal event. The concept is that the anomalous snippets have higher anomaly scores than the normal snippets.

We normalized the video feature length for training. The training of a mini-batch may face problems due to the difference in video embedding feature length T between samples in the mini-batch. Suppose that the video feature vectors of videos V_{v-2} and V_{v-1} are $\Phi_{v-2} = \{\phi_i\}_{i=1}^{T_{k-2}}$ and $\Phi_{k-1} = \{\phi_j\}_{j=1}^{T_{k-1}}$, respectively, where $T_{v-2} \neq T_{v-1}$. It is difficult to train the features in the batches due to $T_{v-2} \neq T_{v-1}$, i.e., the lack of a uniform shape in temporal dimension. Explicitly, it is important to reshape T_{v-2} and T_{v-1} into the same size of T. To handle an arbitrary length of videos in the training phase only, we follow the same normalization technique as Sultani et al. [8] and Joo et al. [20]. As the testing videos are assessed individually, we assume that it is not required to send the features through the normalization process in the testing phase.

Assume that an input mini-batch of 2Ψ videos $\{V_v\}_{v=1}^{2\Psi}$ is available, as shown in Figure 1, where none of the first half $\{V_v\}_{v=1}^{\Psi}$ contains an anomaly snippet and at least one (or more) of the snippets contains an anomaly in the second half $\{V_v\}_{v=\Psi}^{2\Psi}$. Let $\Upsilon \in \mathbb{R}^{2\Psi\times T\times\aleph}$ indicate the extracting features upon using pretrained feature extractors in Section 3.1. During training, the first half of the mini-batch $\Upsilon_n \in \mathbb{R}^{\Psi\times T\times\aleph}$, which has none of the snippets containing an anomaly feature, is loaded with a set of negative bags, while the second half $\Upsilon_a \in \mathbb{R}^{\Psi\times T\times\aleph}$, which has at least one of the snippets containing an anomaly feature, is loaded with a set of positive bags in order within the mini-batch. Subsequently, both $\Upsilon_n \in \mathbb{R}^{\Psi\times T\times\aleph}$ and $\Upsilon_a \in \mathbb{R}^{\Psi\times T\times\aleph}$ go through the stage of TSAN. The output of TSAN is a set of reweighed normal attention features $\overline{\Phi_n}$, as well as a set of reweighed anomaly attention features $\overline{\Phi_a}$. Then the reweighed attention features undergo the snippet association network, which consists of a pyramid of dilated convolutions [36] and non-local block [37], to determine the long-term and short-term association between snippets, in accordance with the reweighed magnitudes of $\overline{\Phi_n}$ and $\overline{\Phi_a}$. The output of the snippet association network is the final attention features $\overline{\Phi_f} \in \mathbb{R}^{2\Psi\times T\times\aleph}$, which are then passed to a layered MLP-based score converter that converts the feature vectors into a set of $2\Psi T$ anomaly scores. This set of scores is used for computing the score-based binary cross-entropy loss.

Let p_i be the anomaly score of the i^{th} snippet. Given the snippet-wise anomaly scores $\{\widetilde{z}_i\}_{i=1}^{2\Psi T}$, the cross-entropy loss over the top-k snippets can be calculated using Equation (4) as

$$Cross\,entropy\,loss = -\frac{1}{|\varpi|}\sum_{i\in\varpi}[\widetilde{z}_i\,\log(p_i) + (1-\widetilde{z}_i)\,\log(1-p_i)], \tag{4}$$

where ϖ belongs to the set of top-k snippets.

3.4. Separation Maximization Supervisor (SMS) Learning

We employ a SMS, denoted by $\xi_{\tau,\mu}$, to maximize the separation between the top snippets of the positive and negative bags. Each attention feature of $\overline{\Phi_f} \in \mathbb{R}^{2\Psi\times T\times \aleph}$, irrespective of normal and abnormal bags, undergoes SMS. First, $\overline{\Phi_f} \in \mathbb{R}^{2\Psi\times T\times \aleph}$ is rearranged to make it suitable for SMS processing by specifying $\overline{\Phi_{f+}} \in \mathbb{R}^{\Psi\times T\times \aleph}$ and $\overline{\Phi_{f-}} \in \mathbb{R}^{\Psi\times T\times \aleph}$ as anomaly and normal attention features, respectively. Then, we select the top-μ snippets from $\overline{\Phi_{f+}} \in \mathbb{R}^{\Psi\times T\times \aleph}$ and $\overline{\Phi_{f-}} \in \mathbb{R}^{\Psi\times T\times \aleph}$ using feature magnitude. This produces $\overline{\Phi_{f+_\mu}} \in \mathbb{R}^{\Psi\times \mu\times \aleph}$ and $\overline{\Phi_{f-_\mu}} \in \mathbb{R}^{\Psi\times \mu\times \aleph}$ as subsets of $\overline{\Phi_{f+}} \in \mathbb{R}^{\Psi\times T\times \aleph}$ and $\overline{\Phi_{f-}} \in \mathbb{R}^{\Psi\times T\times \aleph}$, respectively. Then, both $\overline{\Phi_{f+_\mu}} \in \mathbb{R}^{\Psi\times \mu\times \aleph}$ and $\overline{\Phi_{f-_\mu}} \in \mathbb{R}^{\Psi\times \mu\times \aleph}$ are averaged out across the top-μ snippets to produce $\beta_{\tau,\mu}(\overline{\Phi_{f+}}) \in \mathbb{R}^{\Psi\times \aleph}$ and $\beta_{\tau,\mu}(\overline{\Phi_{f-}}) \in \mathbb{R}^{\Psi\times \aleph}$ to represent the Ψ-anomaly and Ψ-normal bags, each with a feature vector of length $\aleph$, respectively. Both $\beta_{\tau,\mu}(\overline{\Phi_{f+}})$ and $\beta_{\tau,\mu}(\overline{\Phi_{f-}})$ depend on the parameters of τ, as well as μ. The τ indicates the dependency on the snippet association network, whereas μ points to the selection of the top-μ snippets with the largest temporal feature magnitude. The separability is computed using Equation (5) as

$$\xi_{\tau,\mu}(\overline{\Phi_{f+}}, \overline{\Phi_{f-}}) = \|\beta_{\tau,\mu}(\overline{\Phi_{f+}})\| - \|\beta_{\tau,\mu}(\overline{\Phi_{f-}})\|. \tag{5}$$

Equation (5) maximizes the separability of the top-μ feature snippets from each positive and negative bag by leveraging the theorem of Tian et al. [19].

3.5. Loss Optimization

The feature vectors of $\xi_{\tau,\mu}(\overline{\Phi_{f+}}, \overline{\Phi_{f-}})$ can be averaged across $\aleph$ dimensions to obtain Ψ numerical values of separability for the mini-batch of 2Ψ training videos. These numerical values are averaged out and then used as a portion of the optimized loss. Basically, this portion of loss, as well as the score-based binary cross-entropy loss computed using Equation (4) for $2\Psi T$ anomaly scores are applied to optimize the total loss of the model.

3.6. Testing Phase

During testing, we assumed that the extracted video feature vectors need not move through the feature length normalization process to be reshaped to the common size of T, as the testing videos were assessed independently. The extracted video feature vectors went through TSAN processing to generate the reweighed attention features. They were then passed into the snippet association network, followed by the MLP-based feature vector to anomaly score converter, to obtain a set of scores. Each of these scores portrays the anomaly probability of the snippet at the associated index and conveys a numerical value between 0 and 1. Each score is repeated Δ times to replicate a vector with the usual frame length of the video. It also preserves the original order of the video and utilizes an evaluation against the ground truth labels.

4. Experimental Setup and Results

4.1. Used Datasets

We evaluated our models on the following benchmark datasets:

4.1.1. UMN Dataset

This dataset [38] comprises five dissimilar staged videos, where people walk around and eventually start running in distinct directions. The abnormal events are characterized by running episodes.

4.1.2. UCSD-Peds Dataset

This is a small-scale dataset, which consists of two sub-datasets; namely, UCSD-Ped1 with 70 videos and UCSD-Ped2 with 28 videos. These videos were captured at one location. The anomalies in the videos are straightforward, including people walking across a walkway, non-pedestrian entities (e.g., a skater, biker, and wheelchair) in the walkways. The default training set of UCSD-Peds does not contain anomaly videos. Preceding works [6,19,39] reorganized and utilized the dataset for weakly supervised anomaly detection by randomly selecting six anomaly videos and four normal videos for the training set, with the remainder as a testing set.

4.1.3. ShanghaiTech Dataset

This is a medium-scale dataset, which contains 317,398 frames of video clips, encompassing scenes of multiple areas of the ShanghaiTech Campus. It has 13 different background scenes, with 307 normal videos and 130 anomaly videos. The earliest dataset [40] is a common benchmark used to detect video anomaly events. The training set contains only normal videos. The testing set contains both normal and anomalous videos. Zhong et al. [6] rearranged the dataset by choosing a subset of anomalous testing videos as training data, to build a weakly supervised training set, such that both the training and testing sets covered all 13 background scenes. We used exactly the same procedure as Zhong et al. [6] to convert the ShanghaiTech dataset to the weakly supervised setting.

4.1.4. UCF-Crime Dataset

This is a large-scale anomaly detection dataset [8], which contains 1900 untrimmed videos with a total duration of 128 h from real-world street and indoor surveillance cameras. It covers 13 real-world anomalies including abuse, arrest, arson, assault, accident, burglary, explosion, fighting, robbery, shooting, stealing, shoplifting, and vandalism. Unlike the static background in ShanghaiTech [40], UCF-Crime [8] consists of complicated and diverse backgrounds. The dataset contains 1610 (i.e., 800/810:normal/anomalous) training videos with video-level labels and 290 (i.e., 150/140: normal/anomalous) testing videos with frame-level labels.

4.2. Implementation Details

Following Sultani et al. [8], Tian et al. [19], and Joo et al. [20], each video was divided into 32 snippets (i.e., $T = 32$) with the snippet length set to $\Delta = 16$ frames and $\mu = 4$ by following Equation (2). Referring to Equation (1), the values of k were 19, 17, 24, and 19 for UMN, Peds, ShanghaiTech, and UCF-Crime, respectively. Each mini-batch consisted of samples from 32 randomly selected normal and abnormal videos. We employed C3D [21], I3D [22], and CLIP [26] for feature extraction. The $\aleph$ was set as 512 for all experiments with $\aleph' > \aleph$. The thresholds of the low-variance-filter were 0.00723, 0.00835, 0.00875, and 0.00911 for the datasets of UMN, Peds, ShanghaiTech, and UCF-Crime, respectively. The three-layered MLP of 512, 256, and 1 units with its hidden layer was followed by a ReLU activation function, and its final layer was followed by a sigmoid function, to produce a value between 0 and 1. Our model was trained in an end-to-end manner and implemented using PyTorch [41]. We used the Adam optimizer [42] with a weight decay of 0.0005 and a batch size of 32 for 50 epochs. The learning rate was set to 0.001 for all datasets. We employed an Intel® Core™ i7-7800X CPU @3.50 GHz, along with an NVIDIA graphics card GeForce GTX 1080 for both training and evaluation of the model. We also adopted OpenAI, Google Colab, and Google Drive for feature extraction. We used the area under the receiver operating characteristic (ROC) curve (AUC) to evaluate the overall model

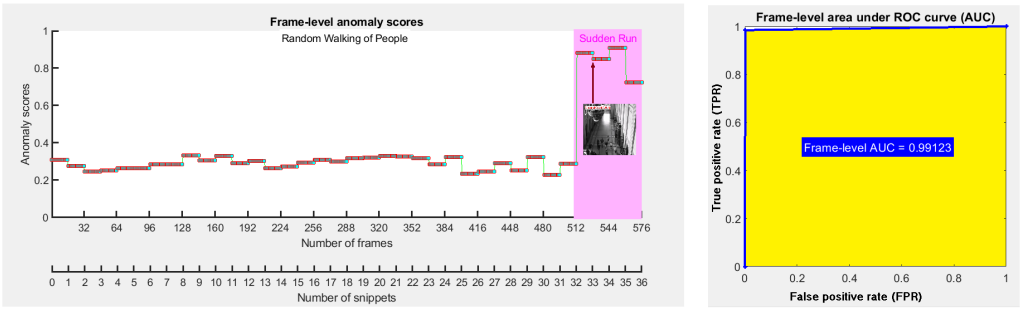
performance. The $0 \leq AUC \leq 1$ is one of the most frequently used metrics for evaluating various flows and events in crowd videos [8,43,44]. The predictions of a model were 100% wrong or correct if $AUC = 0$ or $AUC = 1$, respectively. Intuitively, a larger AUC implies a larger margin between the normal and abnormal snippet predictions, thus resulting in a better anomaly classifier. The sensitivity, recall, hit rate, and true positive rate (TPR) can be formulated using Equation (6) as

$$TPR = \frac{t_p}{t_p + f_n}, \tag{6}$$

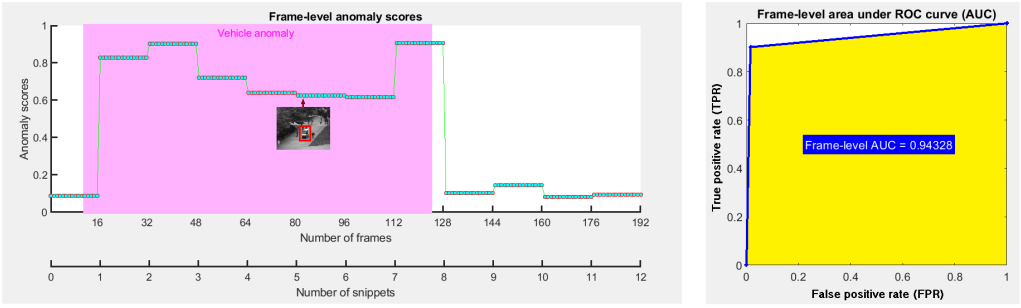
where t_p and f_n specify the number of true positive frames and the number of false negative frames, respectively. The fall-out or false positive rate (FPR) can be formulated using Equation (7) as

$$FPR = 1 - \frac{t_n}{t_n + f_p} = \frac{f_p}{f_p + t_n}, \tag{7}$$

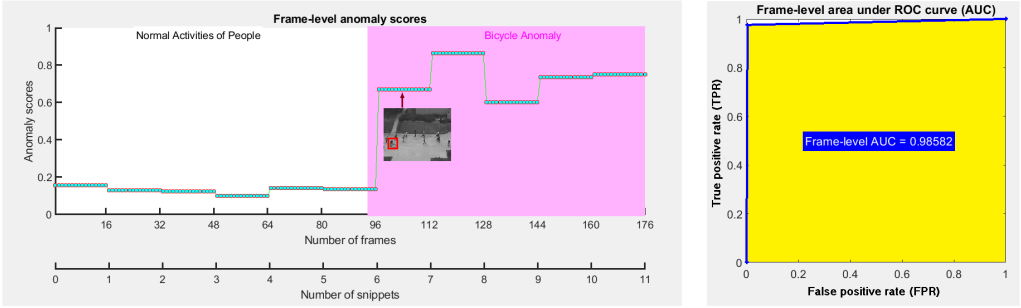
where f_p and t_n indicate the number of false positive frames and the number of true negative frames, respectively. The ROC curve is a two-dimensional graphical visualization, in which the FPR is plotted on the X-axis and the TPR is plotted on the Y-axis (e.g., right side subgraphs of Figure 2). The values of AUC are calculated as the areas below the ROC curves (e.g., the yellow colored regions of Figure 2). Mathematically, the value of AUC can be calculated using the trapezoidal numerical integration method [45].



(a) A sample output from UMN dataset



(b) A sample output from UCSD-Ped1 dataset



(c) A sample output from UCSD-Ped2 dataset

Figure 2. *Cont.*

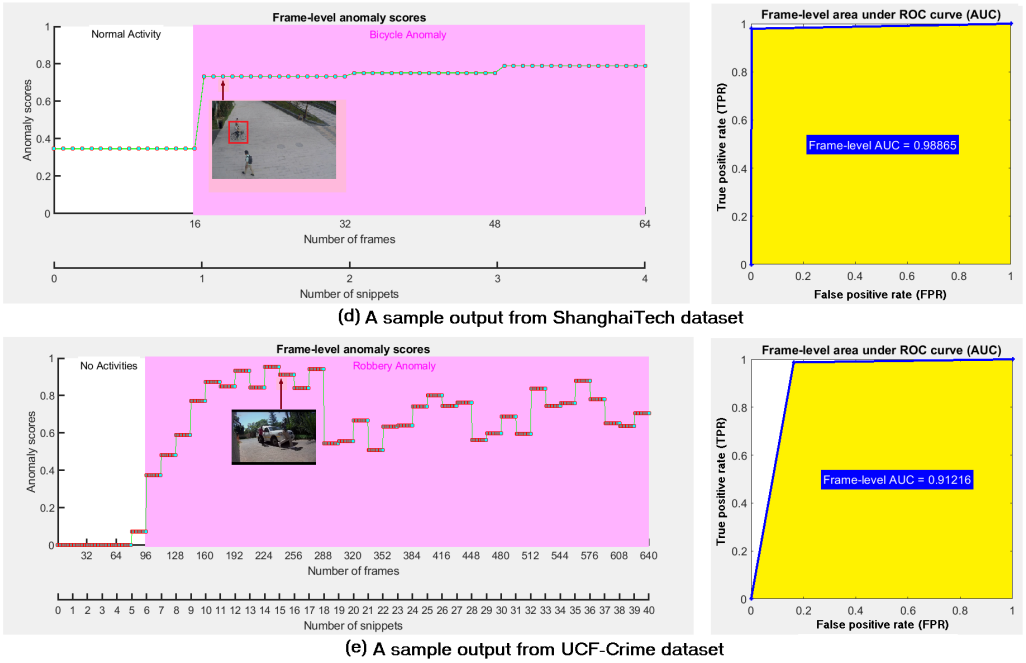


Figure 2. Visualization of sample testing results with various datasets. Pink regions show the manually labeled abnormal events, while the yellow regions indicate the areas below the ROC curves.

4.3. Results on Various Datasets

As real-world abnormal events are miscellaneous and hard to predict, to demonstrate the applicability of our generalized framework to multiple environments, we ran experiments on frequently used VAED evaluation datasets, e.g., UMN, UCSD-Ped1, UCSD-Ped2, ShanghaiTech, and UCF-Crime. Figure 2 visualizes the sample testing results of I3D-CLIP-TSAN (Ours) with videos from the UMN, UCSD-Ped1, UCSD-Ped2, ShanghaiTech, and UCF-Crime datasets, including abnormal events with sudden running of people, vehicles passing between bidirectional flows of people, bicycle riders in a pedestrian zone, bicycles crossing, and the action of taking something from a person forcefully as well as unlawfully, respectively. The obtained frame-level AUC scores of the sample testing videos in Figure 2 were 0.991, 0.943, 0.986, 0.989, and 0.912, consecutively. Although UMN, UCSD-Ped1, and UCSD-Ped2 are popular benchmarks for video anomaly detection, they are small in terms of number of videos and the duration of the video. Alterations in the anomalies are also very narrow. Furthermore, some abnormalities are not practical or sometimes the spatial annotation is not very clear. For these reasons, few authors have conducted experiments with these datasets explicitly. However, we considered all these datasets, to show the generalizability of our models. From Figure 2, it is noticeable that I3D-CLIP-TSAN (ours) was suitable for detecting various anomaly events, ranging from simple datasets (e.g., UMN, UCSD-Ped1, and UCSD-Ped2) to large-scale datasets (e.g., ShanghaiTech and UCF-Crime).

4.4. Performance Comparison

Assume that AUC_o denotes the AUC computed on the overall testing videos in a dataset. Table 1 compares the frame-level AUC_o performance scores of our models for the UCSD-Ped2, ShanghaiTech, and UCF-Crime datasets, along with state-of-the-art methods. It seems that our proposed models could be generalized for detecting various abnormal events from those datasets. In general, both ShanghaiTech and UCF-Crime would be called wide-scale anomaly detection datasets. All authors in Table 1 considered the ShanghaiTech and UCF-Crime datasets for conducting their experiments.

Table 1. Frame-level AUC_o score comparison of the various weakly-supervised methods and datasets. Column-wise the best score is bolded and the second best score is underlined.

Year	Weakly Supervised Model	Feature	Frame-Level Performance Scores from Different Datasets					
			UCSD-Ped2		ShanghaiTech		UCF-Crime	
			AUC_o	$1-AUC_o$	AUC_o	$1-AUC_o$	AUC_o	$1-AUC_o$
2018	Sultani et al. [8]	C3D	—	—	0.8317	0.1683	0.7541	0.2459
	Sultani et al. [8]	I3D	—	—	0.8533	0.1467	0.7792	0.2208
2019	Zhong et al. [6]	C3D	—	—	0.7644	0.2356	0.8108	0.1892
	Zhong et al. [6]	TSN	0.9320	0.0680	0.8444	0.1556	0.8212	0.1788
	Zhang et al. [9]	C3D	—	—	0.8250	0.1750	0.7870	0.2130
2020	Zaheer et al. [46]	C3D-self	0.9447	0.0553	0.8416	0.1584	0.7954	0.2046
	Zaheer et al. [7]	C3D	—	—	0.8967	0.1033	0.8303	0.1697
	Wan et al. [47]	I3D	—	—	0.8538	0.1462	0.7896	0.2104
2021	Purwanto et al. [13]	TRN	—	—	0.9685	0.0315	0.8500	0.1500
	Tian et al. [19]	C3D	—	—	0.9151	0.0849	0.8328	0.1672
	Majhi et al. [48]	I3D	—	—	0.8822	0.1178	0.8267	0.1733
	Tian et al. [19]	I3D	**0.9860**	**0.0140**	0.9721	0.0279	0.8430	0.1570
	Wu et al. [49]	I3D	—	—	0.9748	0.0252	0.8489	0.1511
	Yu et al. [50]	I3D	—	—	0.8783	0.1217	0.8215	0.1785
	Lv et al. [12]	I3D	—	—	0.8530	0.1470	0.8538	0.1462
	Feng et al. [51]	C3D	—	—	0.9313	0.0687	0.8140	0.1860
	Feng et al. [51]	I3D	—	—	0.9483	0.0517	0.8230	0.1770
2022	Zaheer et al. [3]	ResNext	—	—	0.8621	0.1379	0.7984	0.7984
	Zaheer et al. [52]	C3D	0.9491	0.0509	0.9012	0.0988	0.8337	0.1663
	Zaheer et al. [52]	3DResNext	0.9579	0.0421	0.9146	0.0854	0.8416	0.1584
	Joo et al. [20]	C3D	—	—	0.9719	0.0281	0.8394	0.1606
	Joo et al. [20]	I3D	—	—	0.9798	0.0202	0.8466	0.1534
	Joo et al. [20]	CLIP	—	—	<u>0.9832</u>	<u>0.0168</u>	<u>0.8758</u>	<u>0.1242</u>
	Cao et al. [53]	I3D	—	—	0.9645	0.0355	0.8587	0.1413
	Li et al. [34]	I3D	—	—	0.9608	0.0392	0.8530	0.1470
	Cao et al. [54]	I3D-graph	—	—	0.9605	0.0395	0.8467	0.1533
	Tan et al. [55]	I3D	—	—	0.9754	0.0246	0.8671	0.1329
	Li et al. [34]	VideoSwin	—	—	0.9732	0.0268	0.8562	0.1438
	Yi et al. [56]	I3D	—	—	0.9765	0.0235	0.8429	0.1571
	Yu et al. [57]	C3D	—	—	0.8835	0.1165	0.8208	0.1792
	Yu et al. [57]	I3D	—	—	0.8991	0.1009	0.8375	0.1625
	Gong et al. [58]	I3D	—	—	0.9010	0.0990	0.8100	0.1900
2023	Majhi et al. [59]	13D-Res	—	—	0.9622	0.0378	0.8530	0.1470
	Park et al. [60]	C3D	—	—	0.9602	0.0398	0.8343	0.1657
	Park et al. [60]	I3D	—	—	0.9743	0.0257	0.8563	0.1437
	Pu et al. [61]	I3D	—	—	0.9814	0.0186	0.8676	0.1324
	Lv et al. [35]	X-CLIP	—	—	0.9678	0.0322	0.8675	0.1325
	Sun et al. [62]	C3D	—	—	0.9656	0.0344	0.8347	0.1653
	Sun et al. [62]	I3D	—	—	0.9792	0.0208	0.8588	0.1412
	Wang et al. [63]	C3D	—	—	0.9401	0.0599	0.8148	0.1852
	C3D-TSAN (Ours)	C3D	0.9675	0.0325	0.9608	0.0392	0.8578	0.1422
	I3D-TSAN (Ours)	I3D	0.9758	0.0242	0.9743	0.0257	0.8650	0.1350
	CLIP-TSAN (Ours)	CLIP	0.9811	0.0189	0.9806	0.0194	0.8763	0.1237
	C3D-CLIP-TSAN (Ours)	C3D+CLIP	0.9824	0.0176	0.9813	0.0187	0.8802	0.1198
	I3D-CLIP-TSAN (Ours)	I3D+CLIP	<u>0.9839</u>	<u>0.0161</u>	**0.9866**	**0.0134**	**0.8897**	**0.1103**

The reported results in Table 1 indicate that the improvements in performance by our proposed methods on the ShanghaiTech and UCF-Crime datasets were more remarkable than those for the UCSD-Ped2 dataset. However, for a coherent and intelligible comparaison of the performance of the various methods, we performed a non-parametric statistical investigation based on the results presented in Table 1, considering two categories: the first category consisted of ShanghaiTech and UCF-Crime datasets only, while the second category consideredthe UCSD-Ped2, ShanghaiTech, and UCF-Crime datasets.

Figure 3 depicts the Nemenyi [64] post hoc critical distance diagram at a level of significance of $\alpha = 0.05$, considering $1 - AUC_o$ scores in Table 1 for the first category with the existing models of Sultani et al. (2018) [8], Zhong et al. (2019) [6], Zhang et al. (2019) [9], Zaheer et al. (2020) [46], Zaheer et al. (2020) [7], Wan et al. (2020) [47], Purwanto et al. (2021) [13], Tian et al. (2021) [19], Majhi et al. (2021) [48], Wu et al. (2021) [49], Yu et al. (2021) [50], Lv et al. (2021) [12], Feng et al. (2021) [51], Zaheer et al. (2022) [3], Zaheer et al. (2022) [52], Joo et al. (2022) [20], Cao et al. (2022) [53], Li et al. (2022) [34], Cao et al. (2022) [54], Tan et al. (2022) [55], Li et al. (2022) [34], Yi et al. (2022) [56], Yu et al. (2022) [57], Gong et al. (2022) [58], Majhi et al. (2023) [59], Park et al. (2023) [60], Pu et al. (2023) [61], Lv et al. (2023) [35], Sun et al. (2023) [62], and Wang et al. (2023) [63]. If the distance between the two models is less than the Nemenyi [64] post hoc critical distance at a certain p-value (e.g., 0.05), there is no statistically significant difference between them. Explicitly, two models are considered significantly different if their performance variation is greater than the Nemenyi [64] post hoc critical distance. To this end, from Figure 3, it is noticeable that at $\alpha = 0.05$, none of the model pairs are statistically significant, as the heavy red line of length 51.7871 (which is called the Nemenyi [64] post hoc critical distance) is greater than the heavy pink line. For example, the distance between the hypothesis of I3D-CLIP-TSAN (ours) vs. Sultani et al. 2018 (C3D) [8] is $|44 - 1| = 43$ (heavy pink line), which is less than 51.7871 at $\alpha = 0.05$ (i.e., 95% confidence limit). In other words, their distance difference was lacking by a numerical value of $|51.7871 - 43| = 8.7871$. Consequently, I3D-CLIP-TSAN (ours) and Sultani et al. 2018 (C3D) [8] were not statistically significant. Similarly, the hypothesis on the difference by Joo et al. 2022 (CLIP) [20] vs. Sultani et al. 2018 (C3D) [8] was not statistically significant, as their distance difference was lacking by a numerical value of $|51.7871 + 3 - 44| = 10.7871$. However, the model I3D-CLIP-TSAN (ours) was $1 - 8.7871/10.7871 = 18.54\%$, more statistically significant than that of Joo et al. 2022 (CLIP) [20].This implies that I3D-CLIP-TSAN (ours) was slightly better generalized for divergent anomaly event detection from videos from the ShanghaiTech and UCF-Crime datasets than any other model in Table 1.

Figure 4 shows a Nemenyi [64] post hoc critical distance diagram at the level of significance $\alpha = 0.10$ considering the $1 - AUC_o$ scores in Table 1 for the second category with the existing models of Zhong et al. (2019) [6], Zaheer et al. (2020) [46], Tian et al. (2021) [19], and Zaheer et al. (2022) [3]. Few models fell into this category, due to the avoidance of the UCSD-Ped2 dataset by many authors. However, from Figure 4, it is noticeable that the result of the difference of I3D-CLIP-TSAN (Ours) vs. Zaheer et al. 2022 (C3D) is statistically significant, as their distance difference (i.e., $|9.6667 - 1.3333| = 8.3334$) was greater than 7.2184 at a 90% confidence limit. Similarly, the results for the differences of I3D-CLIP-TSAN (Ours) vs. Zhong et al. 2019 (TSN) and C3D-CLIP-TSAN (Ours) vs. Zaheer et al. 2022 (C3D) were statistically significant. However, other results for the differences of this category were not statistically significant, as their distance differences were less than 7.2184.

In summery, some of our proposed methods demonstrated their superiority among many existing state-of-the-art methods, as indicated in Table 1. Notably, the aforementioned statistical analysis shows that the method I3D-CLIP-TSAN (ours) took the top place in the rankings of each category. This implies that I3D-CLIP-TSAN (ours) has the ability to utilize good features from the pretrained CNN-ViT feature extractors considering the available videos and confirmed the high disconnectedness between the standard and abnormal snippets for VAED.

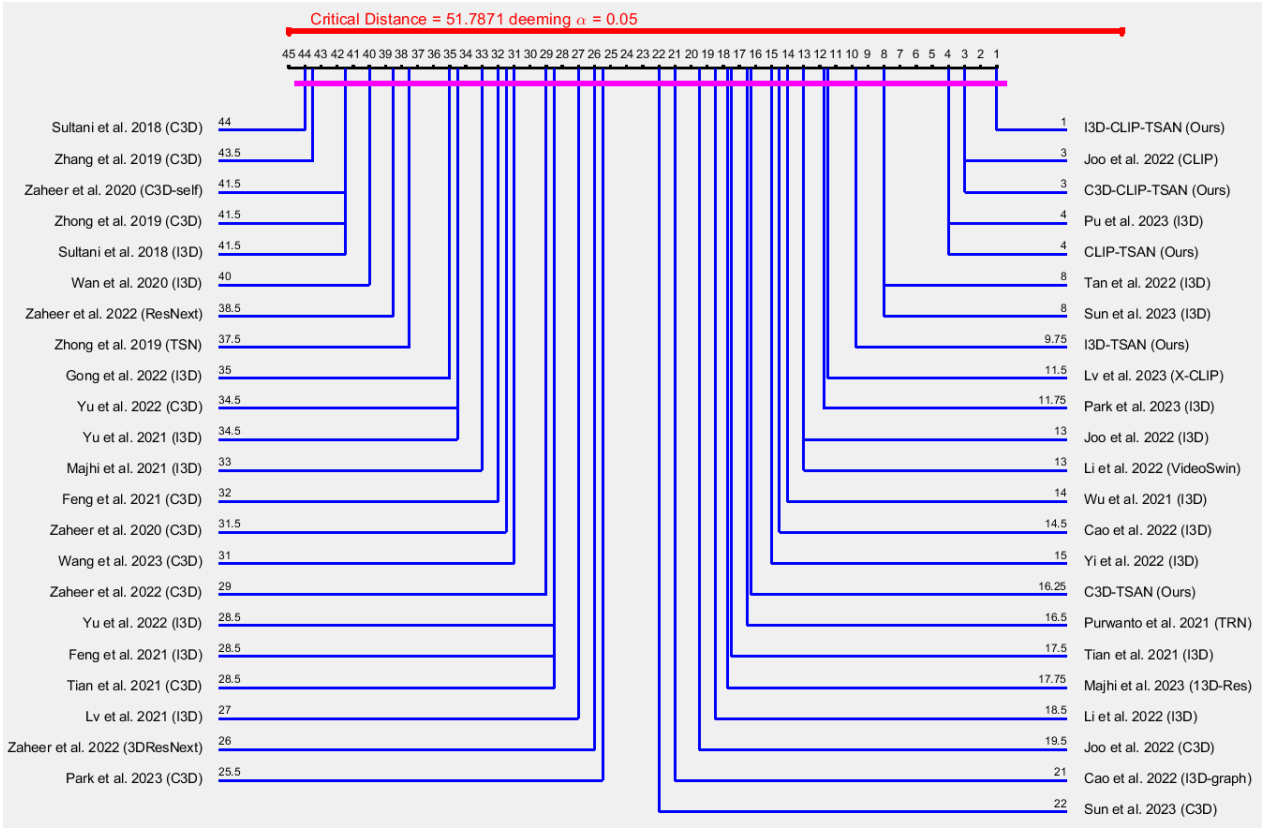


Figure 3. Nemenyi [64] post hoc critical distance diagram with $\alpha = 0.05$ using $1 - AUC_o$ scores in Table 1 for the ShanghaiTech and UCF-Crime datasets.

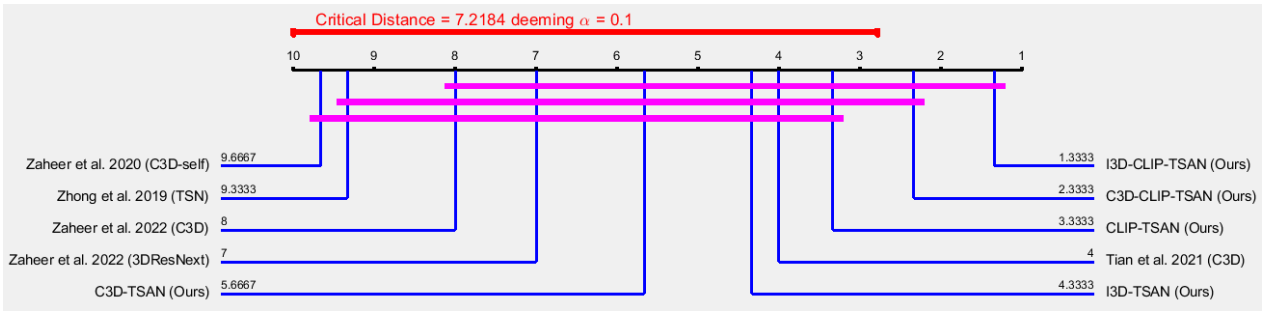


Figure 4. Nemenyi [64] post hoc critical distance diagram with $\alpha = 0.10$ using the $1 - AUC_o$ scores in Table 1 for the UCSD-Ped2, ShanghaiTech, and UCF-Crime datasets.

4.5. Reasons for Superiority

4.5.1. Advantage of Information Fusion

In TSAN, both CNN- and ViT-related processing can produce their own reweighed attention features (e.g., $\Phi_{v_{cnn}} \in \mathbb{R}^{T \times \aleph}$ and $\Phi_{v_{vit}} \in \mathbb{R}^{T \times \aleph}$), which can be directly used by C3D-TSAN, I3D-TSAN, and CLIP-TSAN models, as the features can individually provide necessary (but possibly not sufficient) information for producing the anomaly scores used for anomaly detection. However, the information fusion (e.g., $\Phi_{v_{fusion}} \in \mathbb{R}^{T \times \aleph} = \Phi_{v_{cnn}} \in \mathbb{R}^{T \times \aleph} + \Phi_{v_{vit}} \in \mathbb{R}^{T \times \aleph}$) of these two atypical backbones can augment the quality of feature representation. Both the C3D-CLIP-TSAN and I3D-CLIP-TSAN models applied $\Phi_{v_{fusion}} \in \mathbb{R}^{T \times \aleph}$ and achieved superior performance, as compared to the other models. For example, from Table 1, using the UCF-Crime dataset, the model I3D-CLIP-TSAN (ours) achieved a $1 - 0.8897/0.8650 \approx 3\%$ and $1 - 0.8897/0.8763 \approx 2\%$ better performance with respect to I3D-TSAN (ours) and CLIP-TSAN (ours), respectively. Clearly, the performance

gains of 3% and 2% for I3D-CLIP-TSAN (ours) were the contribution of the information fusion in the TSAN.

4.5.2. Better Information Gains with the Mahalanobis Metric

Tian et al. [19] assumed that the mean feature magnitude of abnormal snippets is larger than that of the normal snippets. However, we applied the measure of Mahalanobis distances, which is much larger and more accurate than that of the mean feature magnitudes. We provide a simple example using the UMN dataset [38].

Usually, any video from the UMN dataset [38] starts with a normal event and ends with an abnormal event. Assume that we obtained the spatiotemporal information of each frame f (where $f \in \{1, 2, \ldots, 900\}$) in a video (e.g., third video) from the UMN dataset [38] using an existing optical-flow method. For any f, irrespective of normal or abnormal events, we consider the spatiotemporal information of five features that are observed in time and put in the form of a matrix $\mathbf{M} \in \mathbb{R}^{n \times 5}$, as follows:

$$\mathbf{M}(\mathbf{u})(\mathbf{v}) = \begin{bmatrix} x(1)(1) & x(1)(2) & x(1)(3) & x(1)(4) & x(1)(5) \\ . & . & . & . & . \\ x(i)(1) & x(i)(2) & x(i)(3) & x(i)(4) & x(i)(5) \\ . & . & . & . & . \\ x(n)(1) & x(n)(2) & x(n)(3) & x(n)(4) & x(n)(5) \end{bmatrix}, \tag{8}$$

where $u \in \{1, 2, \ldots, n\}$; $i \in u$; $v \in \{1, 2, 3, 4, 5\}$; $x(i)(1) \mapsto x$-coordinate of i; $x(i)(2) \mapsto y$-coordinate of i; $x(i)(3) \mapsto x$-velocity of i; $x(i)(4) \mapsto y$-velocity of i; and $x(i)(5) \mapsto$ resulting motion direction of i.

We calculate the sum of the mean feature magnitudes of f denoted as $S_{mean}(f)$ and the sum of Mahalanobis distances (considering Algorithm 1) denoted as $S_{Mahal}(f)$ using Equations (9) and (10), respectively:

$$S_{mean}(f) = \sum_{i=1}^{5} \left(\frac{1}{n} \sum_{i=1}^{n} M(i)(j) \right), \tag{9}$$

$$S_{Mahal}(f) = \sum_{i=1}^{n} MahalDist(i). \tag{10}$$

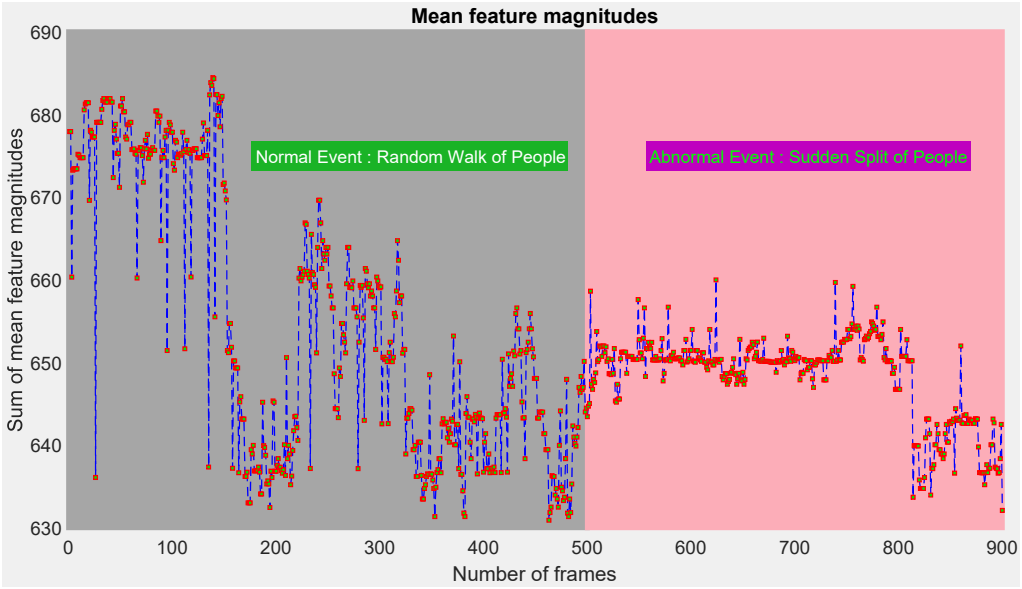
Figure 5 shows a numerical comparison of the sum of mean feature magnitudes and the sum of Mahalanobis distances for a video from the UMN dataset [38]. It is noticeable that the normal and abnormal frames cannot be marked using mean feature magnitudes, whereas the Mahalanobis distances can somewhat find them.Thus, the Mahalanobis distance is more accurate for the ground truth than the mean feature magnitudes. We estimated the probabilities of $S_{mean}(f)$ and $S_{Mahal}(f)$ using Equations (11) and (12), respectively, as

$$P_{mean}(f) = 4\, e^{-\sqrt{\left(\frac{S_{mean}(f)}{65}\right)}}, \tag{11}$$

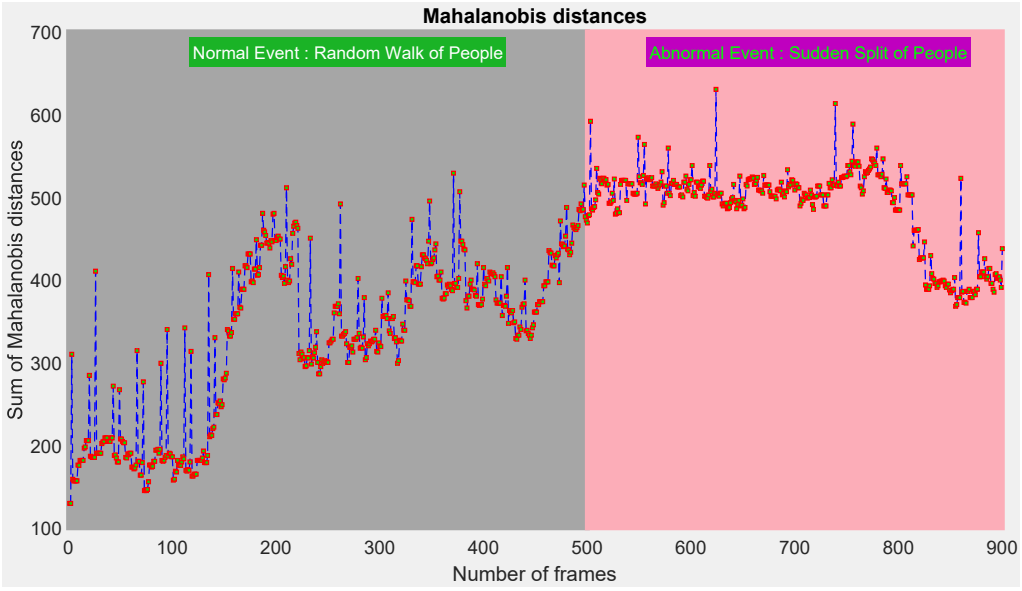
$$P_{Mahal}(f) = 4\, e^{-\sqrt{\left(\frac{S_{Mahal}(f)}{65}\right)}}. \tag{12}$$

In machine learning, the information gain is defined as the amount of information gained for a random variable or a signal from observing another random variable. For such a measure, Kullback—Leibler divergence $D_{\mathrm{KL}}(P_{Mahal}(:) \parallel P_{mean}(:))$ [65] can be applied, where the distributions of $P_{Mahal}(:)$ and $P_{mean}(:)$ include probability values of 900 frames. Equation (13) can be called the information gain achieved, if $P_{Mahal}(:)$ is employed as an alternative to $P_{mean}(:)$. If $P_{Mahal}(:)$ and $P_{mean}(:)$ perfectly match, then $D_{\mathrm{KL}}(P_{Mahal}(:) \parallel P_{mean}(:)) = 0$, or else it can take values between 0 and ∞.

$$D_{\mathrm{KL}}(P_{Mahal}(:) \parallel P_{mean}(:)) = \sum_{f=1}^{900} \left((P_{Mahal}(f)) \left(\log \left(\frac{P_{Mahal}(f)}{P_{mean}(f)} \right) \right) - P_{Mahal}(f) + P_{mean}(f) \right). \quad (13)$$



(**a**)



(**b**)

Figure 5. Numerical comparison of mean feature magnitudes and Mahalanobis distances. (**a**) Normal and abnormal frames cannot be distinguished using mean feature magnitudes. (**b**) Mahalanobis Distances can somewhat make difference between normal and abnormal frames.

The calculated score of 118.41 in Equation (13) quantifies how much the probability distribution of $P_{Mahal}(:)$ differs from the $P_{mean}(:)$ probability distribution on identical grounds. Explicitly, the information gain achieved by $P_{Mahal}(:)$ with respect to $P_{mean}(:)$ was about 118. To keep pace with ground truth, the sum of the mean feature magnitudes for an abnormal event should be either greater or lesser than that of a normal event, but Figure 5a does not reflect this. On the other hand, to keep pace with the ground truth, the sum of Mahalanobis distances for an abnormal event should be either greater or lesser than that of a normal event, and Figure 5b reflects this. As Figure 5 shows that the measure of

Mahalanobis distance is closer to the ground truth, the measure of Mahalanobis distance is more accurate than that of the mean feature magnitudes. The practical results for different datasets on identical grounds also reflected this proposition.

4.6. Analysis of the Best Network

From the input videos, the spatial features of the independent frames conveyed information about the depicted scenes and objects, whereas the temporal features of the frame sequences deal with the information of motion and movement of the objects. A 2D-CNN can learn various spatial features (e.g., edges, corners, and textures) by combining the input frame with a number of filters. The 2D-CNN is highly effective in extracting spatial features from individual frames of a video, but it is not well-suited for capturing temporal information. To accurately capture the temporal dynamics of objects in a video, a different type of neural network must be utilized. A long short-term memory (LSTM) network is a better choice for capturing temporal information. A LSTM network is a deep learning architecture based on an artificial recurrent neural network (RNN). It was specifically designed to handle sequential data, including videos, when modeling the short-range and long-range relationships of sequence features [66]. It also resolves the gradient vanishing problem of the RNN. It is usually used for time series predictions [67]. However, to apply an LSTM network for temporal feature extraction, the output of the 2D-CNN spatial feature extractor can be fed to the LSTM network as input [66]. This can be performed by utilizing the output of the last fully connected layer of the 2D-CNN as the input for the LSTM. In this fashion, the LSTM network can utilize the spatial information extracted by the CNN, together with its capacity to recall past inputs to make predictions regarding the temporal relationships in the video.

Both RNNs and LSTMs are laborious to train because they need memory-bandwidth-bound computation, which is laborious for hardware designers and eventually limits the applicability of neural networks solutions. By combining 2D-CNN and LSTM, it is possible to extract both spatial and temporal features from a videos. One of the reasons why researchers are more partial to using 2D-CNN over LSTM is the amount of training time required. The contemporary generation of well-known deep learning hardware applications mostly use Nvidia graphics cards, and they are optimized for processing 2D data with the greatest possible parallelism and speed, which 2D-CNN brings into service. Nevertheless, one of the main disadvantages of LSTM is its inability to handle temporal dependencies that are longer than a few steps. For example, when an LSTM was trained on a dataset with long-term dependencies (e.g., 100 steps), the network struggled to learn the task and generalize to new examples [68]. Furthermore, on the whole, when data are scarce or noisy, an LSTM tends to overfit the training data and suffers the loss of generalization ability [69]. As a result, it is discouraged to use an LSTM for extracting temporal features. A better solution for extracting temporal features is to employ a C3D network. For example, to take advantage of a 2D-CNN architecture, all filters and pooling kernels of 2D-CNN models can be inflated to a 3D-CNN, by equipping them with an additional temporal dimension, i.e., $\eta \times \eta$ filters become $\eta \times \eta \times \eta$ filters. Afterwards, the weights of 2D filters can be repeated η times along the temporal dimension, to bootstrap parameters from pretrained 2D-CNN models to the 3D-CNN models [70].

We propose TSAN, which generates reweighed attention features by measuring the degree of abnormality of snippets. Explicitly, the mechanism of TSAN maximizes attention on a subset of features, while minimizing the attention on noise. To a large extent, our exceptional performance comes from the utilization of the TSAN along with the fusion of the features of I3D and the rich contextual vision-language features of CLIP.

Most of the existing approaches in Table 1 encode visual content by applying a CNN-based backbone of either C3D or I3D. Like the existing C3D or I3D based models in Table 1, our proposed C3D-TSAN and I3D-TSAN models demonstrated a performance of a comparable nature. Nevertheless, the I3D-TSAN model showed superior performance to the C3D-TSAN model on identical setups. The C3D was more suitable for spatiotemporal

feature learning compared to the 2D CNN [27]. Fundamentally, the operation of 2D convolution tries to convolve an image and the 2D convolution kernel to extract the spatial features from an image, whereas the function of 3D convolution is to convolve the cube constructed by stacking several successive video frames and the 3D convolution kernel for extracting video features in the spatiotemporal dimension. More specifically, the C3D is an excellent model for applying 3D convolution kernels, which is natural for processing signals with spatiotemporal features, such as videos. Even so, its complicated structure stops it becoming deeper [71]. The I3D is an improved model based on C3D. Basically, I3D puts into practice an inflated version of the inception module architecture [30]. The fundamental features of the inception module are the employment of the incorporated effects of filters with various sizes and pooling kernels, all in one layer; as well as the manipulation of 1×1 convolutional filters, which not only assist in lessening the number of parameters but also put in place updated combinations of features to the next layers. This reveals the fact that the performance of the I3D-TSAN network is better than that of the C3D-TSAN, due to the improved architecture and more generalized features of the I3D.

On the other hand, the ViT based CLIP-TSAN model showed the best performance among the three proposed models of C3D-TSAN, I3D-TSAN, and CLIP-TSAN. Both C3D and I3D have a traditional method of convolution, where some channels may be less useful information and consume computational power [72]. Basically, both C3D and I3D were pretrained on action recognition tasks. Differently from the action recognition problem, video anomaly detection depends on discriminative representations that clearly present the events in a scene. Thus, the existing C3D and I3D backbones are not suitable due to the issue of domain gap [1]. To explain this impediment, recently, ViT-based pretrained models (e.g., CLIP, X-CLIP, VideoSwin) were leveraged [20,34,35], which proved the effectiveness of feature representation learning. For example, the ViT-based method of Joo et al. [20] outperformed all existing CNN-based methods in Table 1. Similarly, our proposed CLIP-TSAN model showed almost the same performance as the model of Joo et al. [20]. Our proposed model C3D-CLIP-TSAN demonstrated a better performance than CLIP-TSAN, due to the information fusion [4] from CNN and ViT. Nevertheless, the C3D-CLIP-TSAN model showed slightly inferior performance to I3D-CLIP-TSAN on identical grounds. This was largely due to the I3D simply having a better architecture than the C3D [22]. For instance, the I3D operates on two 3D stream inputs, whereas the C3D operates on single 3D stream input [73].

4.7. Ablation Study

We conducted an ablation study to investigate the effectiveness of the Mahalanobis metric for our generalized framework of CNN-ViT-TSAN. We conducted the experiments in two cases: (i) with the Mahalanobis metric and (ii) without the Mahalanobis metric but with a mean feature magnitude of snippets for identical configuration settings. Table 2 reports their performance. From Table 2, it can be observed that the maximum 5.01%, 5.18%, 4.99%, 5.25%, and 5.56% performance gains were obtained for the UMN, UCSD-Ped1, UCSD-Ped2, ShanghaiTech, and UCF-Crime datasets by applying the Mahalanobis metric. In summery, for these datasets, on average, a maximum 5% better performance could obtained empirically by employing the Mahalanobis metric (i.e., without using the mean snippet feature magnitude).

4.8. Limitation of Our Model

Our WVAED models utilize extracted feature representations using CNN- and/or ViT-based pretrained feature extractors as input. As a result, the performance of our models partially depends on the pretrained feature extractors, making the calculation costly. In the testing phase, if the length of a snippet is Δ frames, then less than Δ frames video clips can be discarded or padded with the final label of the video. In this paper, we chose the former case with $\Delta = 16$ frames. Thus, less than 16 frames of video clips were ignored, which might contain useful information for performance evaluation.

Table 2. Ablation study of Mahalanobis metric on various datasets. Column-wise the best score is bolded and the second best score is underlined.

Feature	Mahalanobis Metric Included?	Frame-Level Performance Scores from Different Datasets									
		UMN		UCSD-Ped1		UCSD-Ped2		ShanghaiTech		UCF-Crime	
		AUC_o	Gain	AUC_o	Gain	AUC_o	Gain	AUC_o	Gain	AUC_o	Gain
C3D	No	0.9136	1.00	0.8553	1.00	0.9214	1.00	0.9129	1.00	0.8262	1.00
	Yes	0.9517	4.17%	0.8996	**5.18%**	0.9675	**4.99%**	0.9608	**5.25%**	0.8578	3.82%
I3D	No	0.9362	1.00	0.8903	1.00	0.9489	1.00	0.9359	1.00	0.8401	1.00
	Yes	0.9644	3.01%	0.9085	2.04%	0.9758	2.83%	0.9743	4.09%	0.8650	2.96%
CLIP	No	0.9417	1.00	0.9063	1.00	0.9597	1.00	0.9391	1.00	0.8346	1.00
	Yes	0.9731	3.33%	0.9274	2.33%	0.9811	2.23%	0.9806	4.42%	0.8763	4.99%
C3D+CLIP	No	0.9405	1.00	0.8871	1.00	0.9396	1.00	0.9422	1.00	0.8348	1.00
	Yes	0.9876	**5.01%**	0.9315	5.01%	0.9824	4.56%	0.9813	4.15%	0.8812	**5.56%**
I3D+CLIP	No	0.9461	1.00	0.8943	1.00	0.9448	1.00	0.9400	1.00	0.8462	1.00
	Yes	**0.9903**	4.67%	**0.9402**	5.13%	**0.9839**	4.14%	**0.9866**	4.96%	**0.8897**	5.14%

5. Conclusions

We proposed an MIL-based generalized architecture named CNN-ViT-TSAN by applying CNN- and/or ViT-extracted features and the use of TSAN, to design a series of deep models for the WVAED problem. Our proposed TSAN mechanism minimized the attention on noise but maximized attention on a subset of features. Instead of using the mean feature magnitude, we uniquely introduced the usage of the Mahalanobis distance for the WVAED problem. At least a 5% performance gain was empirically recorded by employing the Mahalanobis distance with an identical setup as for the mean snippet feature magnitude. The information fusion between CNN and ViT was a unique contribution of this paper. Our deep models possessed a distinct degree of feature extraction ability and usability. One of our models (I3D-CLIP-TSAN) was capable of utilizing a better quality of features and confirmed a high separability between normal and abnormal snippets for VAED. The empirical results from several publicly available crowd datasets demonstrated the generalization ability and applicability of our models against the state-of-the-art approaches to the WVAED problem.

Fundamentally, our model is a natural extension of video classification based on pretrained feature extractors from CNN and ViT. ViT technology has been gaining great interest and its utilization has spread broadly in computer vision. It is assumed that ViT can better capture long-range contextual relationships in videos. We employed CLIP [26] as a ViT feature extractor, and other options including VisualBERT [31], ViLBERT [32], and data efficient CLIP [33] could be employed. Recently, the XD-Violence [10] dataset has become a common benchmark for WVAED [10,19,20]. However, we could not use the XD-Violence [10] dataset due to some nontechnical reason regarding its accessibility (e.g., not being approved by the Norwegian Data Protection Authority); however, in future, we wish to test our models with it.

Author Contributions: Conceptualization, M.H.S.; methodology, M.H.S.; software, M.H.S.; validation, M.H.S., L.J. and C.W.O.; formal analysis, M.H.S., L.J. and C.W.O.; investigation, M.H.S., L.J. and C.W.O.; resources, M.H.S., L.J. and C.W.O.; data curation, M.H.S., L.J. and C.W.O.; writing—original draft preparation, M.H.S., L.J. and C.W.O.; writing—review and editing, M.H.S., L.J. and C.W.O.; visualization, M.H.S.; supervision, L.J. and C.W.O. All authors have read and agreed to the published version of the manuscript.

Funding: This work is a part of the AI4CITIZENS research project (number 320783) supported by the Research Council of Norway.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in this study are openly available and downloadable from http://mha.cs.umn.edu/proj_events.shtml#crowd, http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm, www.cse.cuhk.edu.hk/leojia/projects/detectabnormal/dataset.html, https://svip-lab.github.io/dataset/campus_dataset.html, and https://webpages.charlotte.edu/cchen62/dataset.html, accessed on 28 March 2023. Those datasets, expect http://mha.cs.umn.edu/proj_events.shtml#crowd, were approved by the Sikt (Norwegian Agency for Shared Services in Education and Research) with the reference number of 720663.

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

References

1. Liu, K.; Ma, H. Exploring Background-bias for Anomaly Detection in Surveillance Videos. In Proceedings of the International Conference on Multimedia (MM), Nice, France, 21–25 October 2019; pp. 1490–1499.
2. Gong, D.; Liu, L.; Le, V.; Saha, B.; Mansour, M.R.; Venkatesh, S.; van den Hengel, A. Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection. In Proceedings of the International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1705–1714.
3. Zaheer, M.Z.; Mahmood, A.; Khan, M.H.; Segu, M.; Yu, F.; Lee, S.I. Generative Cooperative Learning for Unsupervised Video Anomaly Detection. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 14724–14734.
4. Sharif, M.; Jiao, L.; Omlin, C. Deep Crowd Anomaly Detection by Fusing Reconstruction and Prediction Networks. *Electronics* **2023**, *12*, 1517.
5. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection: A survey. *ACM Comput. Surv.* **2009**, *41*, 15.
6. 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 Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 1237–1246.
7. Zaheer, M.Z.; Mahmood, A.; Astrid, M.; Lee, S. CLAWS: Clustering Assisted Weakly Supervised Learning with Normalcy Suppression for Anomalous Event Detection. In Proceedings of the European Conference Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; Volume 12367, pp. 358–376.
8. Sultani, W.; Chen, C.; Shah, M. Real-World Anomaly Detection in Surveillance Videos. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 6479–6488.
9. Zhang, J.; Qing, L.; Miao, J. Temporal Convolutional Network with Complementary Inner Bag Loss for Weakly Supervised Anomaly Detection. In Proceedings of the International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 4030–4034.
10. Wu, P.; Liu, J.; Shi, Y.; Sun, Y.; Shao, F.; Wu, Z.; Yang, Z. Not only Look, But Also Listen: Learning Multimodal Violence Detection Under Weak Supervision. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; Volume 12375, pp. 322–339.
11. Zhu, Y.; Newsam, S.D. Motion-Aware Feature for Improved Video Anomaly Detection. In Proceedings of the British Machine Vision Conference (BMVC), Cardiff, UK, 9–12 September 2019; p. 270.
12. Lv, H.; Zhou, C.; Cui, Z.; Xu, C.; Li, Y.; Yang, J. Localizing Anomalies From Weakly-Labeled Videos. *IEEE Trans. Image Process.* **2021**, *30*, 4505–4515. [CrossRef]
13. Purwanto, D.; Chen, Y.T.; Fang, W.H. Dance with Self-Attention: A New Look of Conditional Random Fields on Anomaly Detection in Videos. In Proceedings of the International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 173–183.
14. Thakare, K.V.; Sharma, N.; Dogra, D.P.; Choi, H.; Kim, I.J. A multi-stream deep neural network with late fuzzy fusion for real-world anomaly detection. *Expert Syst. Appl.* **2022**, *201*, 117030.
15. Sapkota, H.; Yu, Q. Bayesian Nonparametric Submodular Video Partition for Robust Anomaly Detection. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 3202–3211.
16. Liu, Y.; Liu, J.; Ni, W.; Song, L. Abnormal Event Detection with Self-guiding Multi-instance Ranking Framework. In Proceedings of the International Joint Conference on Neural Networks, IJCNN 2022, Padua, Italy, 18–23 July 2022; pp. 1–7.
17. Carbonneau, M.A.; Cheplygina, V.; Granger, E.; Gagnon, G. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognit.* **2018**, *77*, 329–353.
18. Liu, Y.; Yang, D.; Wang, Y.; Liu, J.; Song, L. Generalized Video Anomaly Event Detection: Systematic Taxonomy and Comparison of Deep Models. *arXiv* **2023**, arXiv:2302.05087.

19. 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 International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 4955–4966.
20. Joo, H.K.; Vo, K.; Yamazaki, K.; Le, N. CLIP-TSA: CLIP-Assisted Temporal Self-Attention for Weakly-Supervised Video Anomaly Detection. *arXiv* **2022**, arXiv:2212.05136.
21. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 221–231.
22. Carreira, J.; Zisserman, A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4724–4733.
23. Patashnik, O.; Wu, Z.; Shechtman, E.; Cohen-Or, D.; Lischinski, D. StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. In Proceedings of the International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 2065–2074.
24. Ho, V.K.V.; Truong, S.; Yamazaki, K.; Raj, B.; Tran, M.T.; Le, N. AOE-Net: Entities Interactions Modeling with Adaptive Attention Mechanism for Temporal Action Proposals Generation. *Int. J. Comput. Vis.* **2023**, *131*, 302–323.
25. Yamazaki, K.; Vo, K.; Truong, S.; Raj, B.; Le, N. VLTinT: Visual-Linguistic Transformer-in-Transformer for Coherent Video Paragraph Captioning. *arXiv* **2022**, arXiv:2211.15103.
26. 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 International Conference on Machine Learning (ICML), Virtual, 18–24 July 2021; Volume 139, pp. 8748–8763.
27. Tran, D.; Bourdev, L.D.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
28. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Gool, L.V. Temporal Segment Networks for Action Recognition in Videos. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 2740–2755. [PubMed]
29. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
30. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.E.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9.
31. Li, L.H.; Yatskar, M.; Yin, D.; Hsieh, C.; Chang, K. VisualBERT: A Simple and Performant Baseline for Vision and Language. *arXiv* **2019**, arXiv:1908.03557.
32. Lu, J.; Batra, D.; Parikh, D.; Lee, S. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 13–23.
33. Li, Y.; Liang, F.; Zhao, L.; Cui, Y.; Ouyang, W.; Shao, J.; Yu, F.; Yan, J. Supervision Exists Everywhere: A Data Efficient Contrastive Language-Image Pre-training Paradigm. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual, 25–29 April 2022.
34. Li, S.; Liu, F.; Jiao, L. Self-Training Multi-Sequence Learning with Transformer for Weakly Supervised Video Anomaly Detection. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), Conference on Innovative Applications of Artificial Intelligence (IAAI), Symposium on Educational Advances in Artificial Intelligence (EAAI), Virtual, 22 February–1 March 2022; pp. 1395–1403.
35. Lv, H.; Yue, Z.; Sun, Q.; Luo, B.; Cui, Z.; Zhang, H. Unbiased Multiple Instance Learning for Weakly Supervised Video Anomaly Detection. *arXiv* **2023**, arXiv:2303.12369.
36. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions. In Proceedings of the International Conference on Learning Representations (ICLR), Puerto Rico, PR, USA, 2–4 May 2016.
37. Wang, X.; Girshick, R.B.; Gupta, A.; He, K. Non-Local Neural Networks. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
38. University, M. Detection of Unusual Crowd Activities in Both Indoor and Outdoor Scenes. 2021. Available online: http://mha.cs.umn.edu/proj_events.shtml#crowd (accessed on 28 March 2023).
39. He, C.; Shao, J.; Sun, J. An anomaly-introduced learning method for abnormal event detection. *Multim. Tools Appl.* **2018**, *77*, 29573–29588.
40. Liu, W.; Luo, W.; Lian, D.; Gao, S. Future Frame Prediction for Anomaly Detection - A New Baseline. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 6536–6545.
41. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 8–14 December 2019; pp. 8024–8035.
42. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations, (ICLR), San Diego, CA, USA, 7–9 May 2015.

43. Sharif, M.H. An Eigenvalue Approach to Detect Flows and Events in Crowd Videos. *J. Circuits Syst. Comput.* **2017**, *26*, 1750110. [CrossRef]
44. Sharif, M.H.; Jiao, L.; Omlin, C.W. Deep Crowd Anomaly Detection: State-of-the-Art, Challenges, and Future Research Directions. *arXiv* **2022**, arXiv:2210.13927.
45. Rahman, Q.I.; Schmeisser, G. Characterization of the speed of convergence of the trapezoidal rule. *Numer. Math.* **1990**, *57*, 123–138. [CrossRef]
46. Zaheer, M.Z.; Mahmood, A.; Shin, H.; Lee, S.I. A Self-Reasoning Framework for Anomaly Detection Using Video-Level Labels. *IEEE Signal Process. Lett.* **2020**, *27*, 1705–1709. [CrossRef]
47. Wan, B.; Fang, Y.; Xia, X.; Mei, J. Weakly Supervised Video Anomaly Detection via Center-Guided Discriminative Learning. In Proceedings of the International Conference on Multimedia and Expo (ICME), London, UK, 6–10 July 2020; pp. 1–6.
48. Majhi, S.; Das, S.; Brémond, F. DAM: Dissimilarity Attention Module for Weakly-supervised Video Anomaly Detection. In Proceedings of the International Conference on Advanced Video and Signal Based Surveillance (AVSS), Washington, DC, USA, 16–19 November 2021; pp. 1–8.
49. Wu, P.; Liu, J. Learning Causal Temporal Relation and Feature Discrimination for Anomaly Detection. *IEEE Trans. Image Process.* **2021**, *30*, 3513–3527. [CrossRef]
50. Yu, S.; Wang, C.; Ma, Q.; Li, Y.; Wu, J. Cross-Epoch Learning for Weakly Supervised Anomaly Detection in Surveillance Videos. *IEEE Signal Process. Lett.* **2021**, *28*, 2137–2141. [CrossRef]
51. Feng, J.C.; Hong, F.T.; Zheng, W.S. MIST: Multiple Instance Self-Training Framework for Video Anomaly Detection. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, 19–25 June 2021; pp. 14009–14018.
52. Zaheer, M.Z.; Mahmood, A.; Astrid, M.; Lee, S. Clustering Aided Weakly Supervised Training to Detect Anomalous Events in Surveillance Videos. *arXiv* **2022**, arXiv:2203.13704.
53. Cao, C.; Zhang, X.; Zhang, S.; Wang, P.; Zhang, Y. Weakly Supervised Video Anomaly Detection Based on Cross-Batch Clustering Guidance. *arXiv* **2022**, arXiv:2212.08506.
54. Cao, C.; Zhang, X.; Zhang, S.; Wang, P.; Zhang, Y. Adaptive graph convolutional networks for weakly supervised anomaly detection in videos. *arXiv* **2022**, arXiv:2202.06503.
55. Tan, W.; Yao, Q.; Liu, J. Overlooked Video Classification in Weakly Supervised Video Anomaly Detection. *arXiv* **2022**, arXiv:2210.06688.
56. Yi, S.; Fan, Z.; Wu, D. Batch feature standardization network with triplet loss for weakly-supervised video anomaly detection. *Image Vis. Comput.* **2022**, *120*, 104397. [CrossRef]
57. Yu, S.; Wang, C.; Xiang, L.; Wu, J. TCA-VAD: Temporal Context Alignment Network for Weakly Supervised Video Anomly Detection. In Proceedings of the International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, 18–22 July 2022; pp. 1–6.
58. Gong, Y.; Wang, C.; Dai, X.; Yu, S.; Xiang, L.; Wu, J. Multi-Scale Continuity-Aware Refinement Network for Weakly Supervised Video Anomaly Detection. In Proceedings of the International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, 18–22 July 2022; pp. 1–6.
59. Majhi, S.; Dai, R.; Kong, Q.; Garattoni, L.; Francesca, G.; Bremond, F. Human-Scene Network: A Novel Baseline with Self-rectifying Loss for Weakly supervised Video Anomaly Detection. *arXiv* **2023**, arXiv:2301.07923.
60. Park, S.; Kim, H.; Kim, M.; Kim, D.; Sohn, K. Normality Guided Multiple Instance Learning for Weakly Supervised Video Anomaly Detection. In Proceedings of the Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 2–7 January 2023; pp. 2664–2673.
61. Pu, Y.; Wu, X.; Wang, S. Learning Prompt-Enhanced Context Features for Weakly-Supervised Video Anomaly Detection. *arXiv* **2023**, arXiv:2306.14451.
62. Sun, S.; Gong, X. Long-Short Temporal Co-Teaching for Weakly Supervised Video Anomaly Detection. *arXiv* **2023**, arXiv:2303.18044.
63. Wang, L.; Wang, X.; Liu, F.; Li, M.; Hao, X.; Zhao, N. Attention-guided MIL weakly supervised visual anomaly detection. *Measurement* **2023**, *209*, 112500. [CrossRef]
64. Nemenyi, P. Distribution-Free Multiple Comparisons. Ph.D. Thesis, Princeton University, Princeton, NJ, USA, 1963.
65. Kullback, S.; Leibler, R. On information and sufficiency. *Ann. Math. Stat.* **1951**, *22*, 79–86. [CrossRef]
66. Bousmina, A.; Selmi, M.; Ben Rhaiem, M.A.; Farah, I.R. A Hybrid Approach Based on GAN and CNN-LSTM for Aerial Activity Recognition. *Remote Sens.* **2023**, *15*, 3626. [CrossRef]
67. Aksan, F.; Li, Y.; Suresh, V.; Janik, P. CNN-LSTM vs. LSTM-CNN to Predict Power Flow Direction: A Case Study of the High-Voltage Subnet of Northeast Germany. *Sensors* **2023**, *23*, 901. [CrossRef] [PubMed]
68. Trinh, T.H.; Dai, A.M.; Luong, T.; Le, Q.V. Learning Longer-term Dependencies in RNNs with Auxiliary Losses. In Proceedings of the International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018; Volume 80, pp. 4972–4981.
69. Suzgun, M.; Belinkov, Y.; Shieber, S.M. On Evaluating the Generalization of LSTM Models in Formal Languages. In Proceedings of the Society for Computation in Linguistics (SCiL), New York, NY, USA, 3–6 January 2019; pp. 277–286.
70. Nguyen, N.G.; Phan, D.; Lumbanraja, F.R.; Faisal, M.R.; Abapihi, B.; Purnama, B.; Delimayanti, M.K.; Mahmudah, K.R.; Kubo, M.; Satou, K. Applying Deep Learning Models to Mouse Behavior Recognition. *J. Biomed. Sci. Eng.* **2019**, *12*, 183–196. [CrossRef]

71. Wang, X.; Miao, Z.; Zhang, R.; Hao, S. I3D-LSTM: A New Model for Human Action Recognition. In Proceedings of the International Conference on Advanced Materials, Intelligent Manufacturing and Automation (AMIMA), Zhuhai, China, 17–19 May 2019; pp. 1–6.
72. Liu, G.; Zhang, C.; Xu, Q.; Cheng, R.; Song, Y.; Yuan, X.; Sun, J. I3D-Shufflenet Based Human Action Recognition. *Algorithms* **2020**, *13*, 301. [CrossRef]
73. Obregon, D.F.; Navarro, J.L.; Santana, O.J.; Sosa, D.H.; Santana, M.C. Towards cumulative race time regression in sports: I3D ConvNet transfer learning in ultra-distance running events. In Proceedings of the International Conference on Pattern Recognition (ICPR), Montreal, QC, Canada, 21–25 August 2022; pp. 805–811.