2023 IEEE International Conference on Image Processing (ICIP) | 978-1-7281-9835-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICIP49359.2023.10222289

CLIP-TSA: CLIP-ASSISTED TEMPORAL SELF-ATTENTION FOR WEAKLY-SUPERVISED VIDEO ANOMALY DETECTION

Hyekang Kevin Joo[†], *Khoa Vo*[‡], *Kashu Yamazaki*[‡], *Ngan Le*[‡]

Dept. of Computer Science, University of Maryland, College Park, MD, USA[†]
Dept. of Computer Science and Computer Engineering, University of Arkansas, Fayetteville, AR, USA[‡]

ABSTRACT

Video anomaly detection (VAD) – commonly formulated as a multiple-instance learning problem in a weakly-supervised manner due to its labor-intensive nature – is a challenging problem in video surveillance where the frames of anomaly need to be localized in an untrimmed video. In this paper, we first propose to utilize the ViT-encoded visual features from CLIP, in contrast with the conventional C3D or I3D features in the domain, to efficiently extract discriminative representations in the novel technique. We then model temporal dependencies and nominate the snippets of interest by leveraging our proposed Temporal Self-Attention (TSA). The ablation study confirms the effectiveness of TSA and ViT feature. The extensive experiments show that our proposed CLIP-TSA outperforms the existing state-of-the-art (SOTA) methods by a large margin on three commonly-used benchmark datasets in the VAD problem (UCF-Crime, ShanghaiTech Campus and XD-Violence). Our source code is available at `https://github.com/joos2010kj/CLIP-TSA`.

Index Terms— video anomaly detection, temporal self-attention, weakly supervised, multimodal model, subtlety

1. INTRODUCTION

Video action understanding is an active research field with many applications, e.g., action localization [1, 2, 3, 4] action recognition [5, 6, 7, 8, 9, 10], video captioning [11, 12, 13, 14], etc [15]. Video anomaly detection (VAD) is the task of localizing anomalous events in a given video with three main paradigms, *i.e.*, Fully-supervised (Sup.) [16] , Un-Sup [17] , and Weakly-Sup [18]. While it generally yields high performance, Fully-Sup VAD requires fine-grained anomaly labels (*i.e., frame-level* normal/abnormal annotations), and the problem has traditionally suffered from the laborious nature of data annotation. In Un-Sup VAD, one-class classification (OCC) [19] is a common approach, where the model is trained on only normal class samples with the assumption that unseen abnormal videos have high reconstruction errors. However, the performance of Un-Sup VAD is usually poor as it lacks prior knowledge of abnormality and from its inability to capture all normality variations [20]. Compared to both Un-Sup and Sup VAD, Weakly-Sup VAD is considered the most practical approach because of its competitive performance and annotation efficiency by employing *video-level* labels to reduce the cost of manual fine-grained annotations [21].

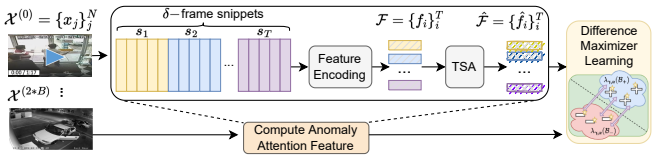

Fig. 1. Overall chart of CLIP-TSA in train time, with $\mathcal{X}^{2*B}$ as input. Each video $\mathcal{X}$ consisting of N frames (*i.e.,* $\mathcal{X} = \{x_j\}_j^N$) is divided into a set of δ-frame snippets $\{s_i\}_i^T$. Each δ-frame snippet s_i is represented by a V-L feature $f_i \in \mathbb{R}^d$. Then, the video is represented by $\mathcal{F} = \{f_i\}_i^T$. Our TSA is then applied to obtain anomaly attention feature $\hat{\mathcal{F}} = \{\hat{f}_i\}_i^T$. The features $\hat{\mathcal{F}}$ from $2 * B$ videos then undergo difference maximization trainer to weakly-train anomaly classifier.

In the Weakly-Sup VAD, there exist two fundamental problems. First, anomalous-labeled frames tend to be dominated by normal-labeled frames, as the videos are untrimmed and there is no strict length requirement for the anomalies in the video. Second, the anomaly may not necessarily stand out against normality. As a result, it occasionally becomes challenging to localize anomaly frames. Thus, the problem is commonly designed with multiple instance learning (MIL) framework [18], where a video is treated as a bag containing multiple instances, each instance being a video snippet. The video is labeled as anomalous if any of its snippets are anomalous, and normal if all of its snippets are normal. Anomalous-labeled videos belong to the positive bag and normal-labeled videos belong to the negative bag. However, the existing MIL-based Weakly-Sup VAD approaches are limited in dealing with an arbitrary number of abnormal snippets in an abnormal video. To address such an issue, we are inspired by the differentiable top-K operator [22] and introduce a novel technique, termed top-κ function, that localizes κ snippets of interest in the video with differentiable hard attention in the similar MIL setting to demonstrate its

[†]Corresponding author's email address: hkjoo@cs.umd.edu

effectiveness and applicability to the traditional, popular setting. Furthermore, we introduce the Temporal Self-Attention (TSA) Mechanism, which generates the reweighed attention feature by measuring the degree of abnormality of snippets.

In addition, the existing approaches encode visual content by applying a backbone, *e.g.*, C3D [23], I3D [24], which are pre-trained on action recognition tasks. Different from the action recognition problem, VAD depends on discriminative representations that clearly represent the events in a scene. Thus, those existing backbones are not suitable because of the domain gap [16]. To address such limitation, we leverage the success of the recent "vision-language" (V-L) works [25], which have proved the effectiveness of feature representation learned via Contrastive Language-Image Pre-training (CLIP) [26, 13, 14]. Our proposed CLIP-TSA follows the MIL framework and consists of three components: (i) Feature Encoding by CLIP; (ii) Modeling snippet coherency in the temporal dimension with our TSA; and (iii) Weakly-training the model with Difference Maximization Trainer.

2. PROPOSED METHOD

2.1. Problem Setup

Let there be a set of weakly-labeled training videos $S = \{\mathcal{X}^{(k)}, y^{(k)}\}_{k=1}^{|S|}$, where a video $\mathcal{X}^{(k)} \in \mathbb{R}^{N_k \times W \times H}$ is a sequence of N_k frames that are W pixels wide and H pixels high, and $y^{(k)} = \{0, 1\}$ is the *video-level* label of video $\mathcal{X}^{(k)}$ in terms of anomaly (*i.e.,* 1 if the video contains anomaly). Given a video $\mathcal{X}^{(k)} = \{x_j\}|_{j=1}^{N_k}$, we first divide $\mathcal{X}^{(k)}$ into a set of δ-frame snippets $\{s_i\}_{i=1}^{\lceil \frac{N_k}{\delta} \rceil}$. Feature representation of each snippet is extracted by applying a V-L model onto the middle frame. In this work, CLIP is chosen as a V-L model; however, it can be substituted by any V-L model. Thus, each δ-frame snippet s_i is represented by a V-L feature $f_i \in \mathbb{R}^d$ and the video $\mathcal{X}^{(k)}$ is represented by a set of video feature vectors $\mathcal{F}_k = \{f_i\}|_{i=1}^{T_k}$, where $\mathcal{F}_k \in \mathbb{R}^{T_k \times d}$ and T_k is the number snippets of $\mathcal{X}^{(k)}$. To allow batch-training, the input features $\mathcal{F}$ come post-normalized into the uniform video feature length in its temporal dimension T as follows [18], with $\lfloor g \rfloor = \lfloor \frac{T_1}{T} \rfloor$ and $\lfloor g \rfloor = \lfloor \frac{T_2}{T} \rfloor$ for $\mathcal{F}_1$ and $\mathcal{F}_2$, respectively:

$$\mathcal{F} = \{f_{i'}\}|_{i'=1}^{T} = \frac{1}{\lfloor g \rfloor} \sum_{i=\lfloor g \times (i'-1) \rfloor}^{\lfloor g \times i' \rfloor} f_i \qquad (1)$$

2.2. Feature Encoding

We choose the middle frame I_i that represents each snippet s_i. We first encode frame I_i with the pre-trained Vision Transformer [27] to extract visual feature I_i^f. We then project feature I_i^f onto the visual projection matrix L, which was pre-trained by CLIP to obtain the image embedding $f_i = L \cdot I_i^f$. Thus, the embedding feature $\mathcal{F}_k$ of video $\mathcal{X}$, which consists of T_k snippets $\mathcal{X} = \{s_i\}|_{i=1}^{T_k}$, is defined in Eq. 2b. Finally, we apply the video normalization as in Eq. 1 onto the embedding feature to obtain the final embedding feature $\mathcal{F}$ as in Eq. 2c.

$$f_i = L \cdot I_i^f \quad \text{where } f_i \in \mathbb{R}^d \qquad (2a)$$
$$\mathcal{F}_k = \{f_i\}|_{i=1}^{T_k} \quad \text{where } \mathcal{F}_k \in \mathbb{R}^{T_k \times d} \qquad (2b)$$
$$\mathcal{F} = Norm(\mathcal{F}_k) \quad \text{where } \mathcal{F} \in \mathbb{R}^{T \times d} \qquad (2c)$$

Algorithm 1:
TSA mechanism σ to produce anomaly attention features $\hat{\mathcal{F}}$

Data: Feature $\mathcal{F} \in \mathbb{R}^{T \times d}$,
Top snippet count $\kappa \in \mathbb{R}^1$
Result: Anomaly attent. feature $\hat{\mathcal{F}}$
$\omega \leftarrow \phi_s(\mathcal{F})$
$\hat{\mathcal{V}} \leftarrow$ Top-κ Score(M, κ, ω)
$\tilde{\mathcal{V}} \leftarrow$ Make d clones of $\hat{\mathcal{V}}$
$\tilde{\mathcal{F}} \leftarrow$ Make κ clones of $\mathcal{F}$
$\mathcal{Q} \leftarrow \tilde{\mathcal{V}} \otimes \tilde{\mathcal{F}}$
$\hat{\mathcal{F}} \leftarrow$ summation of $\mathcal{Q}$ across dim κ
return $\hat{\mathcal{F}}$

Algorithm 2:
Top-κ Score function

Data: Sample count M,
Top snippet count κ,
Score vector ω
Result: A stack of soft one-hot vectors $\hat{\mathcal{V}}$
set $\bar{\omega}$ to M clones of ω
set $\mathcal{G}$ to Gaussian noise
$\hat{\omega} \leftarrow \mathcal{G} \oplus \bar{\omega}$
$\mathcal{U} \leftarrow$ indices of top-κ scores across dim M in $\hat{\omega}$
$\mathcal{V} \leftarrow$ one-hot encode κ in $\mathcal{U}$
$\hat{\mathcal{V}} \leftarrow$ average of $\mathcal{V}$ across dim M
return $\hat{\mathcal{V}}$

2.3. Temporal Self-Attention (TSA)

Our proposed TSA mechanism aims to model the coherency between snippets of a video and select the top-κ most relevant snippets. It contains three modules, *i.e.*, (i) temporal scorer network, (ii) top-κ score nominator, and (iii) fusion network, as shown in Fig. 2 and Alg. 1. In TSA, the vision language feature $\mathcal{F} \in \mathbb{R}^{T \times d}$ (from 2.2 Feature Encoding) is first converted into a score vector $\omega \in \mathbb{R}^{T \times 1}$ through a *temporal scorer network* ϕ_s, *i.e.*, $\omega = \phi_s(\mathcal{F})$. This network is meant to be shallow; thus, we choose a multi-layer perceptron (MLP) of 3 layers in this paper. The scores, each of which is representing the snippet s_i, are then passed into the *top-κ score nominator* to extract the κ most relevant snippets from the video. The top-κ score nominator is implemented by the following two steps. First, the scores $\omega \in \mathbb{R}^{T \times 1}$ are cloned M times and the cloned score $\bar{\omega} \in \mathbb{R}^{T \times M}$ is obtained; M represents the number of independent samples of score vector ω to generate for the empirical mean, which is to be used later for computing the expectation with noise-perturbed features. Second, Gaussian noise $\mathcal{G} \in \mathbb{R}^{T \times M}$ is applied to the stack of M clones by Eq. 3 to produce $\hat{\omega} \in \mathbb{R}^{T \times M}$:

$$\hat{\omega} = \mathcal{G} \oplus \bar{\omega} \qquad \text{where } \oplus \text{ is an element-wise addition} \qquad (3)$$

From the Gaussian-perturbed scores $\hat{\omega} \in \mathbb{R}^{T \times M}$, the indices of top-$\kappa$ snippets are selected based on the score magnitude independently across its M dimension to represent the most relevant snippets and are later one-hot encoded into a matrix $\mathcal{V} = \{V_i\}_{i=1}^{M}$, with each $V_i \in \mathbb{R}^{\kappa \times T}$ containing a set of one-hot vectors. More specifically, we guide the network to place the attention on κ magnitudes with the highest values because the Difference Maximization Trainer (Sec. 2.4) trains the anomalous snippets to have a high value and the normal snippets to have a low value. The matrix $\mathcal{V}$ is then

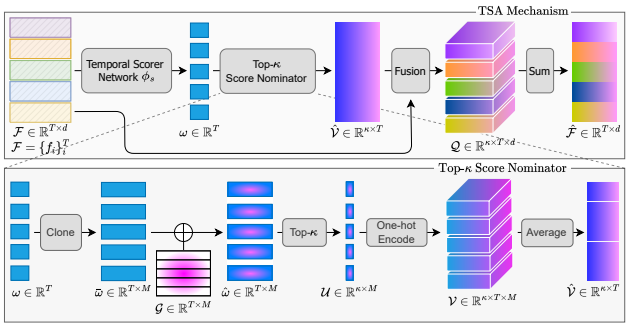

Fig. 2. Top: Our proposed TSA mechanism to model the coherency between snippets. Bottom: Details of top-κ score nominator network, which takes the score vector ω as its input and outputs a stack of soft one-hot vectors $\hat{\mathcal{V}}$.

averaged across its M dimension to produce a stack of soft one-hot vectors $\hat{\mathcal{V}} \in \mathbb{R}^{\kappa\times T}$. Through the soft one-hot encoding mechanism, the higher amount of attention, or weight, is placed near and at the indices of top-κ scores (*e.g.*, [0, 0, 1, 0] $\rightarrow$ [0, 0.03, 0.95, 0.02]). Refer to Alg. 2 for its overview.

Afterward, the stack of perturbed soft one-hot vectors $\hat{\mathcal{V}} \in \mathbb{R}^{\kappa\times T}$ is transformed into $\tilde{\mathcal{V}} \in \mathbb{R}^{\kappa\times T\times d}$ by making d clones of $\hat{\mathcal{V}}$, and the set of input feature vectors $\mathcal{F} \in \mathbb{R}^{T\times d}$ is transformed into $\tilde{\mathcal{F}} \in \mathbb{R}^{\kappa\times T\times d}$ by making κ clones of $\mathcal{F}$. Next, the matrices $\tilde{\mathcal{V}}$ and $\tilde{\mathcal{F}}$, which carry the reweighed information of snippets and represent the input video features, respectively, are fused together to create a perturbed feature $\mathcal{Q} \in \mathbb{R}^{\kappa\times T\times d}$ that represents the reweighed feature magnitudes of snippets based on previous computations as $\mathcal{Q} = \tilde{\mathcal{V}} \otimes \tilde{\mathcal{F}}$, where $\otimes$ is element-wise multiplication.

Then, each stack of perturbed feature vectors $Q \in \mathbb{R}^{\kappa\times d}$ within the perturbed feature $\mathcal{Q} = \{Q_i\}_{i=1}^T$ is independently summed up across its dimension κ to combine the magnitude information of Q_i into one vector $\hat{f}_i \in \mathbb{R}^d$. The reweighed feature vector, $\hat{f}_i \in \mathbb{R}^d$, which collectively forms $\hat{\mathcal{F}} = \{\hat{f}_i\}|_{i=1}^T$, is collectively obtained as output from TSA σ to represent an anomaly attention feature $\hat{\mathcal{F}} \in \mathbb{R}^{T\times d}$.

2.4. Difference Maximization Trainer Learning

Given a mini-batch of $2 * B$ videos $\{\mathcal{X}^{(k)}\}|_{k=1}^{2*B}$, each video $\mathcal{X}^{(k)}$ is represented by $\mathcal{F}_k = \{f_i\}|_{i=1}^T$ obtained by TSA (Section 2.3). Let the input mini-batch be represented by $\mathcal{Z} = \{\mathcal{F}_k\}|_k^{2*B} \in \mathbb{R}^{2*B\times T\times d}$, where B, T, and d denote the user-input batch size, normalized time snippet count, and feature dimension, respectively. The actual batch size is dependent on the user-input batch size, following the equation of $2 * B$, because the first half, $\mathcal{Z}_- \in \mathbb{R}^{B\times T\times d}$, is loaded with a set of normal bags, and the second half, $\mathcal{Z}_+ \in \mathbb{R}^{B\times T\times d}$, is loaded with a set of abnormal bags in order within the mini-batch.

After the mini-batch undergoes the phase of TSA, it outputs a set of reweighed normal attention features $\hat{\mathcal{Z}}_- = \{\hat{\mathcal{F}}_k\}|_{k=1}^B$ and a set of reweighed anomaly attention features $\hat{\mathcal{Z}}_+ = \{\hat{\mathcal{F}}_k\}|_{k=B}^{2*B}$. The reweighed attention features $\hat{\mathcal{Z}}$ are then passed into a convolutional network module J composed of dilated convolutions [28] and non-local block [29] to model the long- and short-term relationship between snippets based on the reweighed magnitudes. The resulting stack of convoluted attention features $\check{\mathcal{Z}} = \{\check{\mathcal{F}}_k\}|_k^{2*B}$, where $\check{\mathcal{Z}} \in \mathbb{R}^{2*B\times T\times d}$, is then passed into a shallow MLP-based score classifier network C that converts the features into a set of scores $U \in \mathbb{R}^{2*B\times T\times 1}$ to determine the binary anomaly state of feature snippets. The set of scores U is saved as part of a group of returned variables, for use in loss.

Next, each convoluted attention feature $\{\check{\mathcal{F}}_k\}|_{k=1}^{2*B}$ of the batch $\check{\mathcal{Z}}$ undergoes Difference Maximization Trainer (DMT). Leveraging the top-α instance separation idea employed by [30, 18], we use DMT, represented by $\upsilon_{\gamma,\alpha}$, in this problem to maximize the separation, or difference, between top instances of two contrasting bags, $\check{\mathcal{Z}}_-$ and $\check{\mathcal{Z}}_+$, by first picking out the top-α snippets from each convoluted attention feature $\check{\mathcal{F}}_k$ based on the feature magnitude. This produces a top-α subset $\dot{\mathcal{F}}_k \in \mathbb{R}^{\alpha\times d}$ for each convoluted attention feature $\check{\mathcal{F}}_k \in \mathbb{R}^{T\times d}$. Second, $\dot{\mathcal{F}}_k$ is averaged out across top-α snippets to create one feature vector $\ddot{\mathcal{F}}_k \in \mathbb{R}^d$ that represents the bag. The procedure is explained by Eq. 4 below:

$$\lambda_{\gamma,\alpha}(\check{\mathcal{F}}) = \ddot{\mathcal{F}} = \max_{\Omega_\alpha(\check{\mathcal{F}})\subseteq\{\check{f}_i\}_{i=1}^T} \frac{1}{\alpha} \sum_{\check{f}_i\in\Omega(\check{\mathcal{F}})} \check{f}_i \qquad (4)$$

Here, λ is parameterized by γ, which denotes its dependency on the ability of the convolutional network module J (*i.e.,* representation of $\ddot{\mathcal{F}}$ depends on the top-α positive instances selected with respect to J). Next, α denotes the size of Ω, which represents a subset of α snippets from $\check{\mathcal{F}}$. Each representative vector $\ddot{\mathcal{F}}$ is then normalized to produce $\dddot{F} \in \mathbb{R}^1$.

$$\upsilon_{\gamma,\alpha}(\check{\mathcal{F}}_+, \check{\mathcal{F}}_-) = ||\lambda_{\gamma,\alpha}(\check{\mathcal{F}}_+)|| - ||\lambda_{\gamma,\alpha}(\check{\mathcal{F}}_-)|| \qquad (5)$$

The separability is computed as in Eq. 5, where $\check{\mathcal{F}}_- = \{\check{f}_{-,i}\}|_i^T$ represents a negative bag and $\check{\mathcal{F}}_+ = \{\check{f}_{+,i}\}|_i^T$ represents a positive bag. More specifically, we leverage the theorem of expected separability [30] to maximize the separability of the top-α instances (feature snippets) from each contrasting bag: Under the assumption that Z_+ has $\epsilon \in [1, T]$ abnormal samples and $(T - \epsilon)$ normal samples, Z_- has T normal samples, and $T = |Z_+| = |Z_-|$, the separability of top-α instances (feature snippets) from positive (abnormal) and negative (normal) bags is expected to be maximized as long as $\alpha \leq \epsilon$. Afterward, to compute the loss, a batch of normalized representative features $\{\ddot{\mathcal{F}}_{normal}\}|_{k=1}^B$ and $\{\ddot{\mathcal{F}}_{abnormal}\}|_{k=B}^{2*B}$ are then measured for margins between each other. A batch of margins is then averaged out and used as part of the net loss together with the score-based binary cross-entropy loss computed using the score set U.

3. EXPERIMENTAL RESULTS

A. Dataset and Implementation Details We conduct the experiment on three datasets, *i.e.*, UCF-Crime [18], Shang-

Table 1. Summarization of Three Datasets

Dataset	# videos	Duration	Train		Test	
			Anomaly	Normal	Anomaly	Normal
UCF-Crime	1,900	128 hours	810	800	140	150
ShanghaiTech	437	317,398 frames	63	175	44	155
XD-Violence	4,754	217 hours	1,905	2,049	500	300

Table 2. Comparisons on UCF-Crime Dataset [18]

Sup.	Method	Venue	Feature	AUC@ROC ↑
Un-	Lu et al. [31]	ICCV'13	C3D	65.51
	Hasan [32]	CVPR'16	-	50.60
	BODS [33]	ICCV'19	I3D	68.26
	GODS [33]	ICCV'19	I3D	70.46
	GCL [34]	CVPR'22	ResNext	71.04
Fully-	Liu & Ma [16]	MM'19	NLN	82.0
Weakly-	GCN [21]	CVPR'19	TSN	82.12
	GCL_{WS} [34]	CVPR'21	ResNext	79.84
	Purwanto et al. [35]	ICCV'21	TRN	85.00
	Thakare et al. [36]	ExpSys'22	C3D+I3D	84.48
	Sultani et al. [18]	CVPR'18	C3D	75.41
	Zhang et al. [37]	ICIP'19		78.70
	GCN [21]	CVPR'19		81.08
	CLAWS [19]	ECCV'20		83.03
	RTFM [38]	ICCV'21		83.28
	Sultani et al. [18]	CVPR'18	I3D	77.92
	Wu et al. [39]	ECCV'20		82.44
	DAM [40]	AVSS'21		82.67
	RTFM [38]	ICCV'21		84.30
	Wu & Liu [41]	TIP'21		84.89
	BN-SVP [42]	CVPR'22		83.39
	Ours: CLIP-TSA		CLIP	**87.58**

haiTech [47], XD-Violence [39], as summarized in Table 1.

Following [18], we divide each video into 32 video snippets, *i.e.,* $T = 32$ (Eq. 1) and snippet length $\delta = 16$. The scorer network θ_s (Sec. 2.3) is defined as an MLP of three layers (512, 256, 1). Leveraging CLIP [26], d is set as 512. We set $M = 100$ for Gaussian noise in Eq. 3. Let $\kappa = \lfloor T \times r \rfloor$; r is a hyperparameter for which we choose 0.7. Our CLIP-TSA is trained in an end-to-end manner using PyTorch. We use Adam optimizer [51] with a learning rate of 0.001, a weight decay of 0.005, and a batch size of 16 for all datasets.

B. Performance Comparison Tables 2, 3, and 4 show the performance comparison between our proposed method and other SOTAs on multiple datasets. In all tables, the best scores are **bolded**, and the runner-ups are underlined. Our remarkable performance mainly comes from the two modules of: (i) rich contextual vision-language feature CLIP and (ii) TSA. To analyze the impact of each module as well as make a fair comparison, we conduct a comprehensive ablation study as in

Table 3. Comparisons on XD-Violence Dataset [39]

Sup.	Modality	Method	Venue	Feature	AUC@PR ↑
Un-	–	OCSVM [43]	NeurIPS'00	–	27.25
		Hasan et al. [32]	CVPR'16	–	30.77
Weakly-	Vision & Audio	Wu et al. [39]	ECCV'20	I3D(V) + VGGish(A)	78.64
		Wu & Liu [41]	TIP'21	I3D(V) + VGGish(A)	75.90
		Pang et al. [44]	ICASSP'21	I3D(V) + VGGish(A)	81.69
		MACIL-SD [45]	MM'22	I3D(V) + VGGish(A)	83.40
		DDL [46]	ICCECE'22	I3D(V) + VGGish(A)	**83.54**
	Vision	Sultani et al. [18]	CVPR'18	C3D(V)	73.20
		RTFM [38]	ICCV'21	C3D(V)	75.89
		RTFM [38]	ICCV'21	I3D(V)	77.81
		Ours: CLIP-TSA		CLIP(V)	82.19

Table 4. Comparisons on ShanghaiTech Campus Dataset [47]

Sup.	Method	Venue	Feature	AUC@ROC ↑
Un-	Hasan et al. [32]	CVPR'16	-	60.85
	Gao et al. [48]	ICCV'19	-	71.20
	Yu et al. [49]	MM'20	-	74.48
	GCL_{PT} [34]	CVPR'21	ResNext	78.93
Weakly-	GCN [21]	CVPR'19	TSN	84.44
	GCL_{WS} [34]	CVPR'21	ResNext	86.21
	Purwanto et al. [35]	ICCV'21	TRN	96.85
	GCN [21]	CVPR'19	C3D	76.44
	Zhang et al. [37]	ICIP'19		82.50
	CLAWS [19]	ECCV'20		89.67
	RTFM [38]	ICCV'21		91.57
	BN-SVP [42]	CVPR'22		96.00
	Sultani et al. [18]	CVPR'18	I3D	85.33
	AR-Net [50]	ICME'20		91.24
	DAM [40]	AVSS'21		88.22
	Wu & Liu [41]	TIP'21		97.48
	RTFM [38]	ICCV'21		97.21
	Ours: CLIP-TSA		CLIP	**98.32**

Table 5. Ablation Study of TSA on Various Features

Feature	TSA	UCF-Crime (AUC@ROC ↑)	ShanghaiTech (AUC@ROC ↑)	XD-Violence (AUC@PR ↑)
C3D	✗	82.59	96.73	76.84
C3D	✓	83.94	97.19	77.66
I3D	✗	83.25	96.39	77.74
I3D	✓	84.66	97.98	78.19
CLIP	✗	86.29	98.18	80.43
CLIP	✓	**87.58**	**98.32**	**82.19**

Table 5. In this table, we benchmark each module by: (i) replacing the CLIP feature with C3D [23] and I3D [24] and (ii) switching on and off our TSA module. By comparing Table 5 against Tables 2-4, we can see that C3D-TSA (replace CLIP feature with C3D) and I3D-TSA (replace CLIP feature with I3D) still outperform the other SOTA models benchmarked with C3D and I3D features on UCF-Crime and ShanghaiTech datasets. As for XD-Violence dataset, we outperform all other vision-based (V) methods, and the SOTAs with a higher score commonly leverage extra modality, audio (A), using VGGish. Evidently, Table 5 shows: (i) performance improvement for all scenarios with our TSA on, and (ii) the best performance is obtained by the proposed CLIP-TSA thanks to the contribution of both strong CLIP feature and TSA module.

Conclusion The paper presents CLIP-TSA, an effective end-to-end weakly-supervised VAD framework. Specifically, we proposed the novel TSA mechanism that maximizes attention on a subset of features while minimizing attention on noise and showed its applicability to this domain. We also applied TSA to CLIP-extracted features to demonstrate its efficacy in Vision-Language features and exploited it in the problem. Comprehensive experiments and ablation studies empirically validate the excellence of our model on the three popular VAD datasets by comparing ours against the SOTAs.

Acknowledgements This paper is supported by the National Science Foundation under Award No OIA-1946391 RII Track-1, NSF 1920920 RII Track 2 FEC, NSF 2223793 EFRI BRAID, NSF 2119691 AI SUSTEIN, and NSF 2236302.

4. REFERENCES

[1] Khoa Vo, Hyekang Kevin Joo, et al., "Aei: Actors-environment interaction with adaptive attention for temporal action proposals generation," in *BMVC*, 2021.

[2] Khoa Vo, Kashu Yamazaki, et al., "Agent-aware boundary networks for temporal action proposal generation," *IEEE Access*, vol. 9, pp. 126431–126445, 2021.

[3] Viet-Khoa Vo-Ho, Ngan Le, et al., "Agent-environment network for temporal action proposal generation," in *ICASSP*, 2021, pp. 2160–2164.

[4] Bo He, Xitong Yang, et al., "Action-aware segment modeling for weakly-supervised temporal action localization," in *CVPR*, 2022, pp. 13925–13935.

[5] Duc-Quang Vu, Ngan Le, and Jia-Ching Wang, "Teaching yourself: A self-knowledge distillation approach to action recognition," *IEEE Access*, vol. 9, pp. 105711–105723, 2021.

[6] Duc-Quang Vu, Ngan TH Le, and Jia-Ching Wang, "(2+ 1) d distilled shufflenet: A lightweight unsupervised distillation network for human action recognition," in *ICPR*, 2022, pp. 3197–3203.

[7] Zehua Sun, Qiuhong Ke, et al., "Human action recognition from various data modalities: A review," *IEEE TPAMI*, 2022.

[8] Kha Gia Quach, Ngan Le, et al., "Non-volume preserving-based fusion to group-level emotion recognition on crowd videos," *Pattern Recognition*, vol. 128, pp. 108646, 2022.

[9] Khoa Vo, Kashu Yamazaki, et al., "Contextual explainable video representation: Human perception-based understanding," in *56th Asilomar*, 2022, pp. 1326–1333.

[10] Zhen Xing, Qi Dai, et al., "Svformer: Semi-supervised video transformer for action recognition," in *CVPR*, 2023, pp. 18816–18826.

[11] Teng Wang, Huicheng Zheng, et al., "Event-centric hierarchical representation for dense video captioning," *IEEE Trans Circuits Syst Video Technol*, vol. 31, no. 5, pp. 1890–1900, 2020.

[12] Jie Lei, Liwei Wang, et al., "MART: Memory-augmented recurrent transformer for coherent video paragraph captioning," in *ACL*, 2020, pp. 2603–2614.

[13] Kashu Yamazaki, Sang Truong, Khoa Vo, Michael Kidd, Chase Rainwater, Khoa Luu, and Ngan Le, "Vlcap: Vision-language with contrastive learning for coherent video paragraph captioning," in *IEEE ICIP*, 2022, pp. 3656–3661.

[14] Kashu Yamazaki, Khoa Vo, et al., "Visual-linguistic transformer-in-transformer for coherent video paragraph captioning," in *AAAI*, 2023, pp. 3081–3090.

[15] Matthew Hutchinson and Vijay Gadepally, "Video action understanding: A tutorial," *IEEE Access*, 2021.

[16] Kun Liu and Huadong Ma, "Exploring background-bias for anomaly detection in surveillance videos," in *ACMM*, 2019, pp. 1490–1499.

[17] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel, "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *ICCV*, 2019, pp. 1705–1714.

[18] Waqas Sultani, Chen Chen, and Mubarak Shah, "Real-world anomaly detection in surveillance videos," in *CVPR*, 2018, pp. 6479–6488.

[19] Muhammad Zaigham Zaheer, Arif Mahmood, et al., "Claws: Clustering assisted weakly supervised learning with normalcy suppression for anomalous event detection," in *ECCV*, 2020, p. 358–376.

[20] Varun Chandola, Arindam Banerjee, and Vipin Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[21] Jia-Xing Zhong, Nannan Li, et al., "Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection," in *CVPR*, 2019, pp. 1237–1246.

[22] Jean-Baptiste Cordonnier, Aravindh Mahendran, et al., "Differentiable patch selection for image recognition," in *CVPR*, June 2021, pp. 2351–2360.

[23] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *TPAMI*, vol. 35, no. 1, pp. 221–231, 2013.

[24] Joao Carreira and Andrew Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *CVPR*, 2017, pp. 6299–6308.

[25] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski, "Styleclip: Text-driven manipulation of stylegan imagery," in *ICCV*, 2021.

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, et al., "Learning transferable visual models from natural language supervision," in *PMLR*, 2021, pp. 8748–8763.

[27] Alexey Dosovitskiy, Lucas Beyer, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *ICLR*, 2021.

[28] Fisher Yu and Vladlen Koltun, "Multi-scale context aggregation by dilated convolutions," in *ICLR*, 2016.

[29] Xiaolong Wang, Ross Girshick, et al., "Non-local neural networks," in *CVPR*, June 2018.

[30] Weixin Li and Nuno Vasconcelos, "Multiple instance learning for soft bags via top instances," in *CVPR*, June 2015.

[31] Cewu Lu, Jianping Shi, and Jiaya Jia, "Abnormal event detection at 150 fps in matlab," in *ICCV*, 2013, pp. 2720–2727.

[32] Mahmudul Hasan, Jonghyun Choi, et al., "Learning temporal regularity in video sequences," in *CVPR*, 2016, pp. 733–742.

[33] Jue Wang and Anoop Cherian, "Gods: Generalized one-class discriminative subspaces for anomaly detection," in *ICCV*, 2019, pp. 8201–8211.

[34] M Zaigham Zaheer, Arif Mahmood, et al., "Generative cooperative learning for unsupervised video anomaly detection," in *CVPR*, 2022, pp. 14744–14754.

[35] Didik Purwanto, Yie-Tarng Chen, and Wen-Hsien Fang, "Dance with self-attention: A new look of conditional random fields on anomaly detection in videos," in *ICCV*, October 2021, pp. 173–183.

[36] Kamalakar Vijay Thakare, Nitin Sharma, and other, "A multi-stream deep neural network with late fuzzy fusion for real-world anomaly detection," *Expert Systems with Applications*, vol. 201, pp. 117030, 2022.

[37] Jiangong Zhang, Laiyun Qing, and Jun Miao, "Temporal convolutional network with complementary inner bag loss for weakly supervised anomaly detection," in *IEEE ICIP*, 2019, pp. 4030–4034.

[38] Yu Tian, Guansong Pang, et al., "Weakly-supervised video anomaly detection with robust temporal feature magnitude learning," in *ICCV*, 2021, pp. 4975–4986.

[39] Peng Wu, Jing Liu, Yujia Shi, Yujia Sun, Fangtao Shao, Zhaoyang Wu, and Zhiwei Yang, "Not only look, but also listen: Learning multimodal violence detection under weak supervision," in *ECCV*, 2020, p. 322–339.

[40] Snehashis Majhi, Srijan Das, and François Brémond, "Dissimilarity attention module for weakly-supervised video anomaly detection," in *AVSS*, 2021, pp. 1–8.

[41] Peng Wu and Jing Liu, "Learning causal temporal relation and feature discrimination for anomaly detection," *IEEE TIP*, vol. 30, pp. 3513–3527, 2021.

[42] Hitesh Sapkota and Qi Yu, "Bayesian nonparametric submodular video partition for robust anomaly detection," in *CVPR*, June 2022, pp. 3212–3221.

[43] Bernhard Schölkopf, Robert C Williamson, Alex Smola, John Shawe-Taylor, and John Platt, "Support vector method for novelty detection," in *NIPS*, 1999, vol. 12.

[44] Wen-Feng Pang, Qian-Hua He, et al., "Violence detection in videos based on fusing visual and audio information," in *ICASSP*, 2021, pp. 2260–2264.

[45] Jiashuo Yu et al., "Modality-aware contrastive instance learning with self-distillation for weakly-supervised audio-visual violence detection," *MM*, 2022.

[46] Yujiang Pu and Xiaoyu Wu, "Audio-guided attention network for weakly supervised violence detection," in *ICCECE*, 2022, pp. 219–223.

[47] W. Liu, D. Lian W. Luo, and S. Gao, "Future frame prediction for anomaly detection – a new baseline," in *CVPR*, 2018.

[48] Xiaoyu Gao, Xiaoyong Zhao, and Lei Wang, "Memory augmented variational auto-encoder for anomaly detection," in *EI*. IEEE, 2021, pp. 728–732.

[49] Guang Yu, Siqi Wang, Zhiping Cai, En Zhu, Chuanfu Xu, Jianping Yin, and Marius Kloft, "Cloze test helps: Effective video anomaly detection via learning to complete video events," in *ACMM*, 2020, pp. 583–591.

[50] Boyang Wan, Yuming Fang, et al., "Weakly supervised video anomaly detection via center-guided discriminative learning," in *ICME*, 2020, pp. 1–6.

[51] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.