Contextual Explainable Video Representation: Human Perception-based Understanding

Khoa Vo
Dept. of CSCE
University of Arkansas
Fayetteville, AR, USA
khoavoho@uark.edu

Kashu Yamazaki
Depth. of CSCE
University of Arkansas
Fayetteville, AR, USA
kyamazak@uark.edu

Phong X. Nguyen
AI Lab
FPT Software
Ho Chi Minh City, Vietnam
phatnt21@fsoft.com.vn

Phat Nguyen
AI Lab
FPT Software
Ho Chi Minh City, Vietnam
phongnx1@fsoft.com.vn

Khoa Luu
Dept. of CSCE
University of Arkansas
Fayetteville, AR, USA
khoaluu@uark.edu

Ngan Le
Dept. of CSCE
University of Arkansas
Fayetteville, AR, USA
thile@uark.edu

2022 56th Asilomar Conference on Signals, Systems, and Computers | 978-1-6654-5906-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/IEEECONF56349.2022.10052051

***Abstract*—Video understanding is a growing field and a subject of intense research, which includes many interesting tasks to understanding both spatial and temporal information, e.g., action detection, action recognition, video captioning, video retrieval. One of the most challenging problems in video understanding is dealing with feature extraction, i.e. extract contextual visual representation from given untrimmed video due to the long and complicated temporal structure of unconstrained videos. Different from existing approaches, which apply a pre-trained backbone network as a black-box to extract visual representation, our approach aims to extract the most contextual information with an explainable mechanism. As we observed, humans typically perceive a video through the interactions between three main factors, i.e., the actors, the relevant objects, and the surrounding environment. Therefore, it is very crucial to design a contextual explainable video representation extraction that can capture each of such factors and model the relationships between them. In this paper, we discuss approaches, that incorporate the human perception process into modeling actors, objects, and the environment. We choose video paragraph captioning and temporal action detection to illustrate the effectiveness of human perception based-contextual representation in video understanding. Source code is publicly available at https://github.com/UARK-AICV/Video_Representation.**

***Index Terms*—video understanding, action detection, dense video captioning, attention, human-perception, explainable ML**

I. Introduction

Video understanding is one of the fundamental field in computer vision that comprises of a wide range of tasks that deal with datasets of videos. These tasks commonly require to extract essential information from the input videos in order to serve different goals.

Based on the present of video pre-processing, we can divide video understanding tasks into two categories of trimmed videos tasks and untrimmed videos tasks. On the one hand, tasks on trimmed videos such as action recognition [1]–[6] or video captioning require input videos to be perfectly trimmed to contain no irrelevant frames (e.g., background frames). On the other hand, tasks on untrimmed videos such as temporal action proposals generation (TAPG) [7]–[13], temporal action detection (TAD) [14], [15], video paragraph captioning (VPC) [16]–[18], video retrieval [19], [20], etc. can process on arbitrary untrimmed videos. In this paper, we focus on the tasks on untrimmed videos not only because they are more challenging in dealing with uncleaned videos but also because they are fundamental tasks to automatically trim the videos or extract crucial information and eliminate irrelevant segments. Particularly, we will provide details discussion of TAPG and VPC as specific tasks.

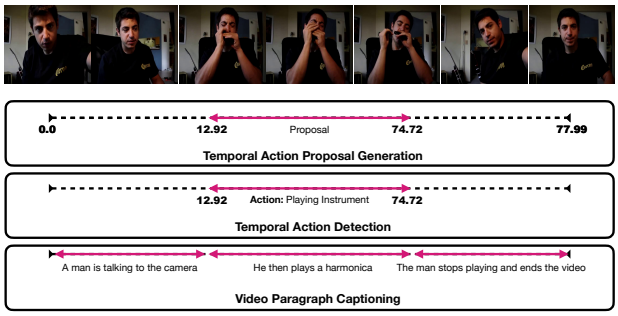


Fig. 1: An illustration of tasks on an untrimmed video, including temporal action proposals generation (top box), temporal action detection (middle box), and video paragraph captioning (bottom box).

Given an untrimmed video, TAPG requires to localize intervals for each presenting action or activity of interest. TAPG is a fundamental task for various downsteam applications, e.g., TAD and VPC. More specifically, TAD additionally requires an action label along with every proposed interval. On VPC, the intervals extracted by TAPG are jointly used to generate a coherent paragraph that describes important events of the input video.

Although TAPG and VPC methods have made great progresses in popular benchmarks of ActivityNet-1.3 [21], THUMOS-14 [22], or ActivityNet Captions [23], they still possess a common limitation, which is the overlooked video

representation. In the respective module of TAPG and VPC methods, the input video frames are clustered into snippets of δ frames, then, a pre-trained 3D convolutional network [24], [25] is used to encode each snippet to a feature vector. Despite being pre-trained on a large dataset (e.g., Kinetics [26]) and able to compress semantic and movement information of the entire snippet in just a feature vector, such feature easily misses information from humans or objects appearing in smaller regions and tends to be biased to the overall spatial environment. Such neglected video representation leads to weak representations for hard scenarios illustrated in Fig. 2. Those scenarios can be briefly described as follows:

- Scenario 1: Existing visual representation is easily biased by environment whereas the action may be independent to the environment as shown in Fig. 2a. This becomes more problematic when the actors occupy smaller regions compared to the overall environment.
- Scenario 2: An arbitrary number of actors can appear in the scene at the same time, but only a few of them are main actors that actually contribute to the formation of an action.
- Scenario 3: The main actor may not even appear inside video frames but only shows their hands interacting with objects to perform actions.

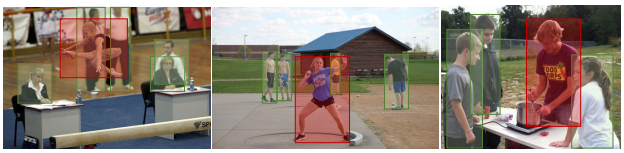
Furthermore, understanding a video involves multiple factors such as single human actor, group human actors, non-human actor, phenomenon [10]–[13]. Examples of non-human actors and phenomena performing actions include dog chasing, car running, and cloud floating.

Inspired by how humans perceives a video (i.e., at a specific timestamp, a human would look at overall scene, then localizing main actors, and perceiving objects that they interact with), our Perception-based Multi-modal Representation (PMR) is proposed in order to comprehensively capture crucial information from multiple entities in the spatial scene of each input snippet of the video. In order to do that, PMR consists of four modules: (i) Environment Beholder, which models the overall scene of input snippet, (ii) Actors Beholder, which models main actors appearing in the input snippet, (iii) Objects Beholder, which models relevant objects of the snippet, and (iv) Actors-Objects-Environment Beholder, which models the relationships between all types of entities. Furthermore, Actors Beholder and Objects Beholder are equipped with our newly proposed Adaptive Attention Mechanism (AAM) to eliminate inessential actors and irrelevant objects, respectively, appearing in the scene and only apply self-attention mechanism on main actors and most relevant objects, respectively.

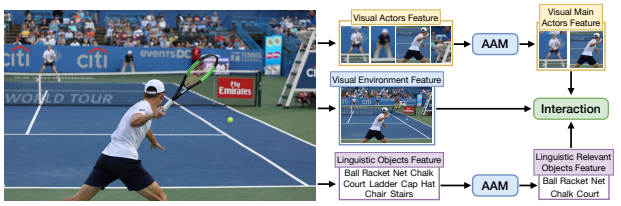
Our contribution can be summarized as follows:

- A discussion about our proposed Multi-modal Representation (PMR) that comprehensively represent video snippets.
- The integration of PMR with state-of-the-art (SOTA) methods in various tasks on untrimmed videos, including TAPG and VPC.
- Extensive experiments showing the effectiveness of PMR in the above tasks by creating a large performance margin over existing SOTAs.

(a) Examples of actions (e.g jogging) are independent to environments.



(b) Examples of how actors contribute to form actions i.e. among all actors (green and red boxes) in the scenes, only main actors (red boxes) actually commit actions.



(d) Our proposed Perception-based Representation (PMR) is modeled by both global visual environment, local visual main actors features, linguistic relevant objects features, and the interaction among them. In PMR, our proposed Adaptive Attention Mechanism (AAM) is to select main actors and relevant objects.

Fig. 2: Most existing TAPG methods [7]–[9], [14], [27] apply a 3D backbone network to entire spatial domain. However, as shown in (a), actors contribute more importance to an action than environment itself. Moreover, (b) shows that main actors who actually commit actions may be among many inessential actors, or (c) actors are not visible in the scene of egocentric videos. *This figure is cited from [13].*

II. "Graybox" Contextual Explainable Representation: A Journey

In this section, we address all aforementioned limitations by introducing a journey of developing a "graybox" contextual explainable representation. Our journey is step-by-step introduced as follows:

A. Actors - Environment Interaction

To alleviate Limitation 1 stated in Sec. I, we propose to model each snippet by two separate entities of local actors and global surrounding environment in [10], [11]. For global environment, we extracted a feature map of the snippet by a pre-trained 3D convolutional network [24] and apply average pooling on the feature map to obtain a single feature vector

Algorithm 1 AAM to extract the representation of main actors in a snippet.

Data: Feature vector f^e and features set $\mathcal{F}^a$ represent environment and all actors that appear in an input snippet, respectively.

Result: Feature vector f^a represents main actors.

$\hat{f}^e \leftarrow MLP_{\theta_e}(f^e)$
set $\tilde{\mathcal{F}}^a$, H^a to empty list ▷ $\mathcal{F}^a$ stores selected main actors, H^a stores scores of every actor
for each f_i^a in $\mathcal{F}^a$ **do**
  $\hat{f}_i^a \leftarrow MLP_{\theta_a}(f_i^a)$
  $h_i^a \leftarrow || \hat{f}_i^a \oplus \hat{f}^e ||_2$ ▷ $\oplus$: element-wise addition
  append h_i^a to H^a
end for
$H^a \leftarrow softmax(H^a)$
$\tau \leftarrow \frac{1}{|\mathbf{h}^a|}$
for each h_i^a in H^a **do**
  if $h_i^a > \tau$ **then**
    append f_i^a to $\tilde{\mathcal{F}}^a$
  end if
end for
$f^a \leftarrow self\_attention(\tilde{\mathcal{F}}^a)$

representing the environment. For local actors, we use an off-the-shelf human detector to localize them using the middle frame of the snippet, each detected bounding box is aligned onto the feature map extracted during the global environment processing to form a set of features for all actors, which in turns are fused together into a single actors feature using a self-attention module [28]. Both features of actors and environment are combined by another self-attention module to flexibly balance between local and global visual representation.

B. Main Actors - Environment Interaction

Limitation 2 poses a very common case where many actors appear in the scene but only several of them are main actors who actually contribute to the actions of interest. To resolve such case, we propose an adaptive attention mechanism (AAM) [12], which aims to (i) eliminate inessential actors who do not majorly affect the content of the scene and can be treated as background, and (ii) adaptively fuse information of selected main actors into a single feature vector.

Given M actors (or objects) obtained in the input snippet, only a few of those, i.e., $\hat{M}$ main actors (or relevant objects), actually contribute to an action. Because $\hat{M}$ is unknown and continuously changes throughout the input video, we propose AAM that inherits the merits from adaptive hard attention [29] to select an arbitrary number of main actors (or objects) and a soft self-attention mechanism [28] to extract relationships among them. Take actors beholder as an instance, AAM is described by the pseudocode in Algorithm 1.

C. Main Actors - Objects - Environment Interaction

The third limitation describes situations where the main actors even absent from the scene and only show their hands to perform actions. In these cases, our previous works [10]–[12] may not work properly due to their reliance on the off-the-shelf actors detector, which can not detect humans for actors modeling. Therefore, we introduce a new entity to comprehensively model the scene in these cases, which is the objects. Capturing objects is very challenging because of two reasons. Firstly, there are various types of objects that can appear in the scenes, and secondly, they frequently appear in very tiny regions, which challenges many existing popular objects detector. To resolve both challenges, we employ the CLIP [30], a powerful pre-trained model that can detect a large amount of objects based on the semantic correlation between their embedding features with the visual features of input image. Modeling the interactions between three types of entities, i.e., actors, objects, and environment help comprehensively capturing important information for downstream tasks. Our proposed AOE-Net [13] with such modeling method has proved to be very effective in TAPG.

In this section, we would like to detail the last model on AOE as follows: Given a N frames video $\mathcal{V} = \{v_i\}_{i=1}^N$, where v_i is the i-th frame, we first follow the standard settings from existing works by segmenting $\mathcal{V}$ into a sequence of $\delta-$frame *snippets* $s_i \mid_{i=1}^T$. Each snippet s_i consists of δ consecutive frames, therefore, $\mathcal{V}$ has a total of $T = \lceil \frac{N}{\delta} \rceil$ snippets. Let $\phi(.)$ be an encoding function to extract the visual feature f_i of a δ-frame snippet s_i; the video $\mathcal{V}$ can be represented as $\mathcal{F}$ as follows:

$$\mathcal{F} = \{f_i\}_{i=1}^T, \text{ where } f_i = \phi(s_i) \tag{1}$$

Different from the existing works [7]–[9], [14], [14], [27], [31]–[33], which simply define $\phi(.)$ as a pre-trained backbone network (e.g., C3D [24], 2Stream [34], SlowFast [35]), we model $\phi(.)$ by the proposed PMR, which is capable of encoding visual information of multiple entities using both visual and linguistic method.

As stated in Sec. I, PMR includes four modules, i.e., (i) Environment Beholder, (ii) Actors Beholder, (iii) Objects Beholder, and (iv) Actors-Objects-Environment Beholder. In the sub-sections below, we discuss about each of those modules consecutively, then, we provide details of AAM, which is the main component of Actors Beholder and Objects Beholder to eliminate inessential actors and irrelevant objects, respectively, and extract mutual relationships of main actors and most relevant objects, respectively.

1) Environment Beholder: is responsible for globally capturing visual information of the input δ-frame snippet. To extract both spatial and temporal information of the snippet, we adopt a pre-trained 3D convolutional network as a backbone feature extractor. The snippet is processed through all convolutional blocks of the backbone except the final linear layers to obtain a feature map $\mathcal{F}^{\mathcal{M}}$, then, an average pooling operator is employed to produce an environment feature vector f^e.

2) Actors Beholder: has a role of semantically extracting visual main actors representation f^a. In most cases, an action cannot happen if a human (main actor) is absent notwithstanding environment (Fig. 2(a)). On the other hand, when an action occurs, it does not necessarily signal that every actor in the scene has committed the action (Fig. 2(b)). Hence, the Actors Beholder first localizes all existing actors (humans) in a δ-frame snippet by an off-the-shelf object detector onto

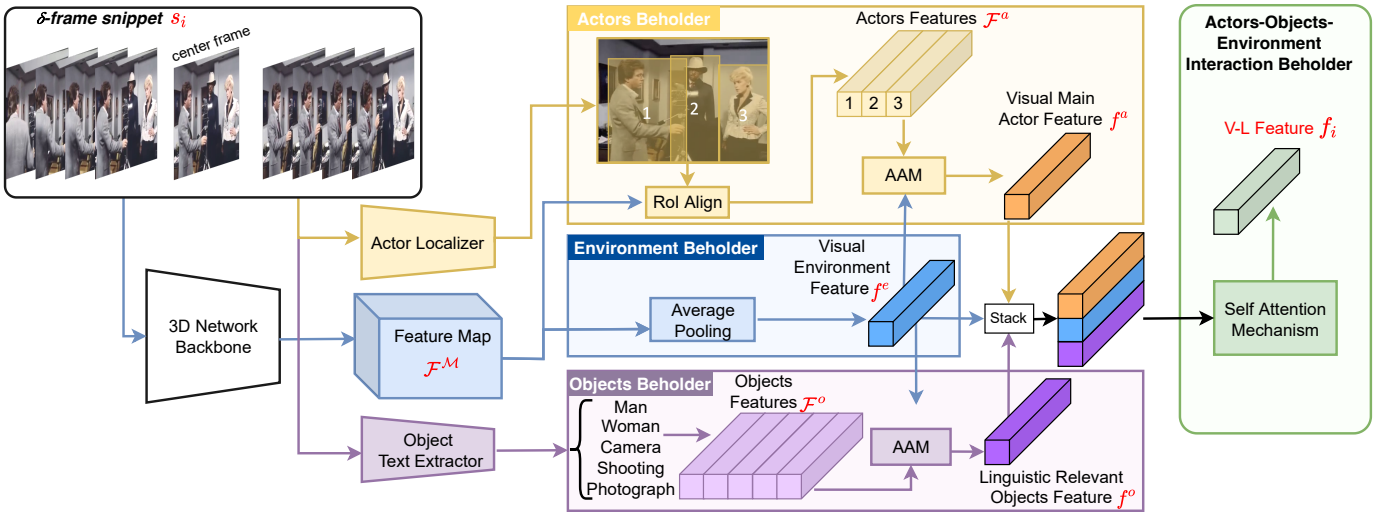


Fig. 3: The architecture of PMR. Given a δ-snippet s_i, the V-L feature is obtained by four modules: (i) actors beholder to extract local visual action feature f^a; (ii) environment beholder to extract global visual environment feature f^e; (iii) objects beholder to extract linguistic object feature f^o, and (iv) actors-objects-environment interaction beholder to model V-L feature as the interaction between actors, objects and the environment.

the middle frame assuming that the actors would not move fast enough to be mis-located with a small δ. We denote $\mathcal{B} = \{b_i\}_{i=1}^{N_B}$ as a set of detected human bounding boxes, where $N_B \geq 0$. Afterwards, each of the detected bounding boxes, b_i, is aligned onto feature map $\mathcal{F}^{\mathcal{M}}$ (obtained from Environment Beholder) using RoIAlign [36]. Then, each bounding box feature is average-pooled into a single feature vector f_i^a. Finally, we obtain a set of actor features $\mathcal{F}^a = \{f_i^a\}_{i=1}^{N_B}$.

To adaptively select an arbitrary number of main actors and extract their mutual relationships, we apply our proposed AAM, which is explained in Algorithm 1.

3) Objects Beholder: Different from the environment and actors, objects may appear very tiny, in the feature map $\mathcal{F}^{\mathcal{M}}$. Hence, in this objects beholder, we propose to use linguistic information from relevant objects, which is considerably more informative than visual information. We leverage CLIP [30] as a powerful pre-trained model to extract linguistic information.

As our task just focuses on human activities and their related objects, we utilize the corpus of ActivityNet Captioning annotations [23] to construct the object text vocabulary $\mathcal{T} = \{\mathcal{T}_i\}_{i=1}^{D}$.

ActivityNet Captioning dataset [23] annotates the same set of videos in ActivityNet-1.3 [21]. Video captions are composed by a vocabulary of up to 10,648 words. In order to create a vocabulary which majorly contains objects and human activities, we eliminate stop words, pronouns, numbers, and infrequent words (which appears 5 times or lower in the whole dataset). Afterwards, we remove words that do not present in the vocabulary of CLIP [30]. To this end, the final vocabulary for our objects beholder consists of $D = 3,544$ words.

Each word $\mathcal{T}_i \in \mathcal{T}$ is encoded by a Transformer network [28] into a text feature $\mathcal{T}_i^f$. Let W_t be a text projection matrix pre-trained by CLIP, the embedding text vocabulary is computed as $\mathcal{T}^e = W_t \cdot \mathcal{T}^f$, where $\mathcal{T}^f = \{\mathcal{T}_i^f\}_{i=1}^{D}$. Let W_i be an image projection matrix pre-trained by CLIP, a middle frame I of the δ-frame snippet is first encoded by Vision Transformer [37] to extract visual feature I^f, and then embedded by W_i, i.e., $I^e = W_i \cdot I^f$. The pairwise cosine similarities between embedded I^e and $\mathcal{T}^e$ is then computed. Top K similarity scores are chosen as output objects text represented by feature $\mathcal{F}^o = \{\mathcal{T}_i^f\}_{i=1}^{K}$. Similar to the actors beholder, we apply the proposed AAM (described in Algorithm 1) to select relevant objects from $\mathcal{F}^o$, then model the semantic relations among them, and finally obtain linguistic feature f^o.

4) Actors-Objects-Environment (AOE) Beholder:: AOE Beholder models the relations between global visual environment feature f^e, local visual of main actors features f^a, and linguistic relevant objects features f^o. Firstly, we stack three types of features together as $\mathcal{F}^{aoe} = [f^a, f^o, f^e]$. Then, we employ the self-attention model [28] followed by an average pooling layer to fuse the stack of features $\mathcal{F}^{aoe}$ into f_i. f_i is a visual-linguistic feature that represents the input snippet s_i through both visual (environment and actors modalities) and linguistic (objects modality) ways.

III. Contextual Explanation Representation in TAPG

To integrate our proposed PMR into TAPG task, we adopt the SOTA method of Boundary Matching Network (BMN) [8] as the action proposals generation module. BMN takes the V-L features sequence $\mathcal{F} = \{f_i\}_{i=1}^{T}$ from our PMR as its input. BMN contains three components: semantic modeling, temporal estimation (TE), and proposal estimation (PE). Semantic modeling captures temporal relations between snippets. The TE component evaluates the probabilities of each snippet $s_i \,|_{i=1}^{T}$ to be an action starting (P_i^S) or ending (P_i^E) boundaries. Finally, the PE component evaluates every interval $[i, j]$ in the video to estimate its actionness score $P_{i,d}^A$, where $d = j - i$. We refer readers to [8], [13] for a detailed description on the architecture of BMN.

A. Training Method

We follow [7], [8] to generate ground truth labels, including starting labels and ending labels for TE training, and duration labels for PE training.

The starting and ending labels are generated for every snippet of the input video, which are $L^S = \{l_n^s\}_{n=1}^T$ and $L^E = \{l_n^e\}_{n=1}^T$, respectively. A label l_n^s (or l_n^e) is set to 1 if its corresponding snippet s_i is the nearest one to any groundtruth starting boundary (or ending boundary).

The duration labels are $L^A \in \{0,1\}^{D\times T}$ where D is the maximum length of proposals being considered in number of snippets (we set $D = T$ in all of our experiments as suggested in [8]). With an element at position (t_i, t_j) stands for a proposal action $a_p = (t_s = \frac{t_j \cdot T}{t_v}, t_e = \frac{(t_j+t_i)\cdot T}{t_v})$, it will be assigned by 1 if its temporal Interaction-over-Union with any ground truth action in $\mathcal{A} = \{a_i\}_{i=1}^M$ reaches a local maximum, or 0 otherwise.

Three outputs of BMN, i.e., P^S, P^E, and P^A, are trained through three corresponding loss functions of $\mathcal{L}_s(P^S, L^S)$, $\mathcal{L}_e(P^E, L^E)$, and $\mathcal{L}_{act}(P^A, L^A)$. Where $\mathcal{L}_s$ and $\mathcal{L}_e$ are defined as weighted binary log-likelihood loss:

$$\mathcal{L}_{wb}(P, L) = \sum_{i=1}^{N} \left[\frac{L_i}{N^+} \log P_i + \frac{(1-L_i)}{N^-} \log(1-P_i) \right]$$

where N^+ and N^- are the number of positives and negatives in groundtruth labels, respectively. Conversely, $\mathcal{L}_{act}(P, L)$ is defined as follows:

$$\mathcal{L}_{act}(P, L) = \mathcal{L}_{wb}(P, L) + \lambda \mathcal{L}_2(P, L)$$

, where $\mathcal{L}_2$ is the mean squared error loss and λ is set to 10.

IV. Contextual Explanation Representation in VPC

Like TAPG task, in VPC [16], [17], [38], our PMR is employed to extract feature sequences that are served to the Paragraph Generation Module (PGM). PGM operates through each event of the video in the chronological order, then generates a caption describing the event. PGM not only has to maintain the consistency of every word in an event caption, but also need to model the coherency of all captions, to generate a smooth and sound paragraph that describes the input video.

Towards such requirement, we proposed a novel Transformer-in-Transformer (TinT) architecture, which includes (a) an inner Transformer Decoder [28] that generates caption of an event using its corresponding PMR features sequence, and (b) an outer Transformer that maintains the paragraph coherency via self-attention on a set of hidden states, each of which is produced after every event. We refer readers to [18] for a detailed description on the process of our TinT method.

A. Training Method

Given an event e_k and its groundtruth caption $C_k = \{c_i\}_{i=1}^{|C_k|}$, we employ the commonly used Kullback-Leibler (KL) divergence loss as our main training loss $\mathcal{L}_{cap.}$ to train our TinT model so that the predicted caption distribution becomes similar to groundtruth distribution. Besides, following [39] to additionally use a regularization term $\tau(C)$ that penalizes frequently predicted tokens, to reduce redundant phrases in the predicted paragraph. The optimization is illustrated as equations below:

$$\mathcal{L}_{cap.} = -\frac{1}{N}\sum_{i=1}^{N}(\log p_\theta(s_i|s_{<i}, \mathcal{V}_e)) + \lambda\tau(\mathbf{s})$$

$$\tau(C) = -\frac{1}{|C|}\sum_{i=1}^{|C|}\sum_{c\in\{c|C_{<i}\}} \log\left(1 - p_\theta(c|C_{<i}, \mathcal{E})\right)$$

where we set $\lambda = 0.1$ in all our VPC experiments.

Methods	Feature	AR@100	AUC(val)	AUC(test)
TCN [40]	2Stream	–	59.58	61.56
MSRA [41]	P3D	–	63.12	64.18
SSTAD [42]	C3D	73.01	64.40	64.80
CTAP [43]	2Stream	73.17	65.72	–
BSN [7]	2Stream	74.16	66.17	66.26
SRG [44]	2Stream	74.65	66.06	–
MGG [45]	I3D	74.54	66.43	66.47
BMN [8]	2Stream	75.01	67.10	67.19
DBG [9]	2Stream	76.65	68.23	68.57
BSN++ [27]	2Stream	76.52	68.26	–
TSI++ [31]	2Stream	76.31	68.35	68.85
MR [46]	I3D	75.27	66.51	–
SSTAP [47]	I3D	75.54	67.53	–
TCANet [48]	2Stream	76.08	68.08	–
Zheng, et.al. [49]	2Stream	74.93	65.20	–
AEN [10]	C3D	75.65	68.15	68.99
ABN [11]	C3D	76.72	69.16	69.26
AEI [12]	C3D	*77.24*	*69.47*	*70.09*
PMR + BMN [13]	C3D	**77.67**	**69.71**	**70.10**

TABLE I: **TAPG** comparisons on ActivityNet-1.3 [21] in terms of AR@100 and AUC on validation set and AUC on testing set. Methods in bottom section use the contextual explainable representation as stated in Sec. II.

Methods	Input	B4 ↑	M ↑	C ↑	R ↑	Div2 ↑	R4 ↓
Vanilla Trans. [50]	Res200/Flow	9.31	15.54	21.33	28.98†	77.29†	7.45
Trans.-XL [17]	Res200/Flow	10.25	14.91	21.71	30.25†	76.17†	8.79
Trans.-XLRG [16]	Res200/Flow	10.07	14.58	20.34	–	–	9.37
MART [16]	Res200/Flow	9.78	15.57	22.16	30.85†	75.69†	5.44
MART^{COOT} [51]	COOT	10.85	15.99	28.19	–	–	6.64
Memory Trans. [39]	I3D	11.74	15.64	26.55	–	**83.95**	**2.75**
PMR+TinT [18]	C3D/Ling	**14.50**	**17.97**	**31.13**	**36.56**	77.72	4.75

TABLE II: Performance comparison of PMR+TinT with other SOTA models on ActivityNet Captions *ae-test*. † denotes results obtained by ourselves.

V. Experiments

A. Datasets and Metrics

For both TAPG and VPC, we evaluate our proposed PMR with BMN [8] module on the popular dataset of ActivityNet-1.3, which includes 10,009 training videos, 4917 validation videos, and 5,044 testing videos.

On TAPG, each video is annotated with intervals containing one of 200 activities of interest. We evaluate and compare our method with SOTAs by two common metrics of AR@100 and AUC. AR@100 is the average recall (AR) calculated with an average of 100 proposals per video, while AUC is the area under the AR vs. AN curve score.

Exp	Setting					TAPG Performance				
	Act.	Env.	Obj.	AAM	Soft-Att	@50	@100	@200	@500	@1000
#1	√	×	×	×	√	25.96	35.14	43.48	52.37	58.47
#2	×	√	×	×	×	38.94	47.80	54.93	61.92	65.96
#3	×	×	√	×	√	18.06	26.68	37.14	49.28	56.99
#4	√	√	×	×	√	40.87	49.09	56.24	63.53	67.29
#5	√	√	√	×	√	42.60	49.86	56.87	63.76	67.60
#6	√	√	×	√	×	43.79	49.67	56.73	63.49	67.36
#7	√	√	√	√	×	44.56	50.26	57.30	64.32	68.19

TABLE III: TAPG comparisons on different network settings. Act., Env., Obj. denote actors, environment, objects beholders.

On VPC, each video is densely annotated with important events, each event is described by a single sentence. On average, there are 7.7 events per video. Besides, VPC task of ActivityNet-1.3 splits the validation set into ae-val subset with 2460 videos and ae-test subset with 2457 videos. We evaluate and compare our method wtih SOTAs by common metrics in image captioning and video captioning, i.e., BLEU-4 (B@4) [52], METEOR (M) [53], and CIDEr (C) [54]. To evaluate the diversity of generated captions, we use two diversity metrics of 2-gram diversity (Div@2) [55] and 4-gram repetition (R@4) [56].

B. Implementation Details

We employ the C3D [24] network pre-trained on Kinetics-400 [26] as the backbone network in all experiments on both tasks. Features extracted by C3D have 2048 dimensions.

For Objects Beholder, we adopt the powerful CLIP model [30] pre-trained on a large-scale dataset of 400M image-text pairs crawled from the Internet to extract object texts. In the Actors Beholder, to detect humans, we adopt Faster-RCNN model [57] pre-trained on the COCO dataset [58]. Adam optimizer was used in all experiments, and the initial learning rate is set to 1e-4 for both tasks.

C. Performance and comparison on TAPG

Table I presents the evaluation of our PMR on TAPG and comparisons with previous SOTAs on ActivityNet-1.3 [21]. The experimental results demonstrate that our proposed representation with BMN outperforms the existing methods in terms of AR@100 and AUC by an adequate margin. Notably, the performance on TAPG of our AOE-Net is competitive with AEI-B [12], which is followed closely by ABN [11], both of which also incorporate local actors and global environment. This experiment strongly supports our observation and motivation on using the human perception principle to analyze human actions in untrimmed videos.

D. Performance and comparison on VPC

We benchmark and compare our PMR and TinT modules on VPC task with the prior SOTAs on both ActivityNet Captions *ae-test* in Table II. Compared to SOTA approaches, i.e., MART [16], MART w/COOT [51], and PDVC [59], our approach outperforms with large margins on both accuracy and diversity metrics on ActivityNet Captions. For example, the accuracy gains 3.65%/1.98%/2.94%5.71% on B@4/M/C/R metrics whereas diversity increases 0.43% on Div@2 and reduces 0.67% on R@4 compared to the second-best performance. Qualitative comparisons on VPC are illustrated in Fig. 5.

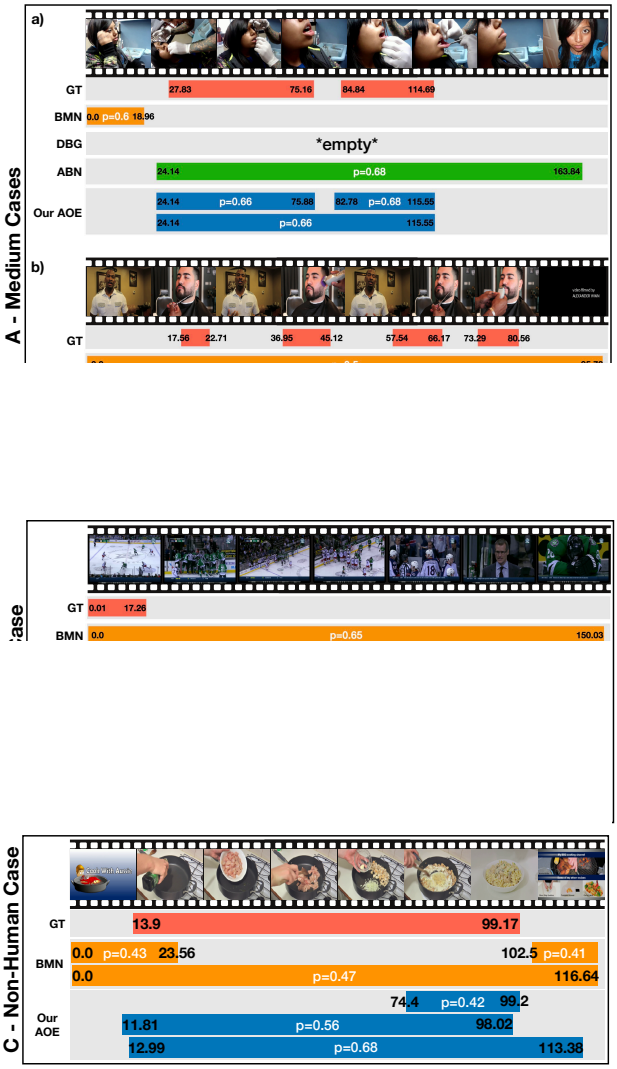


Fig. 4: Qualitative results in TAPG on ActivityNet-1.3 [21] dataset.

VI. Conclusion

In this paper, we present a novel video representation method, namely Perception-based Multi-modal Representation (PMR), which simulates the human perception process. Our PMR extracts the visual-linguistic representation of each snippet with four modules. Environment beholder and actors beholder capture global and local visual features of environment and main actors, respectively. Objects beholder extracts linguistic feature from relevant objects. The last beholder aims to model the relations between main actors, relevant objects and environment. To focus on an arbitrary number of main actor(s) or relevant objects, we introduced AAM.

We evaluate PMR on two untrimmed videos understanding tasks, i.e., temporal action proposals generation (TAPG) and video paragraph captioning (VPC). On TAPG, we employ the SOTA method of BMN [8] as the proposals generation module, while on VPC, we propose a novel Transformer-in-Transformer

Env.	Act.	Obj.	*at-test* split B@4 ↑	M ↑	C ↑	R ↑	Div@2↑	R@4 ↓	*ae-val* split B@4 ↑	M ↑	C ↑	R ↑	Div@2 ↑	R@4 ↓
√	×	×	13.62	17.41	29.09	35.96	76.14	5.97	14.02	17.58	30.31	36.20	76.11	6.08
×	√	×	11.83	16.22	21.39	33.97	79.20	4.16	12.13	16.57	24.98	34.36	79.18	4.24
×	×	√	13.38	17.69	30.30	35.63	**80.50**	**3.32**	14.00	17.88	31.64	35.95	**80.44**	**3.22**
√	√	×	13.77	17.52	30.05	35.93	77.78	4.69	14.12	17.78	31.15	36.12	78.02	4.56
√	×	√	**14.53**	17.79	30.83	**36.67**	76.47	5.60	14.84	17.97	31.86	36.80	76.41	5.67
√	√	√	14.50	**17.97**	**31.13**	36.56	77.72	4.75	**14.93**	**18.16**	**33.07**	**36.86**	77.72	4.87

TABLE IV: VPC comparisons on different network settings. Env., Act., and Obj. denote the global visual environment, local visual main agents, and linguistic relevant objects, respectively.

architecture [18] as the paragraph generator. On both tasks, we reported the quantitative and qualitative results, which suggest that our proposed PMR makes an adequate improvement to the selected SOTA modules.

However, we also observe several limitations, which shows some room for further research to improve our PMR. First, the Objects Beholder only represents objects as text features, however, the visual appearance and motions introduced to those objects may be good information to the representation. Second, the Actors Beholder assume humans as actors, but in a general scenario, actors can also be animals, therefore, it is more beneficial if Actors Beholder can learn to localize actors instead of relying on an off-the-shelf objects detector.

Acknowledgments: This material is based upon work supported by the National Science Foundation (NSF) under Award No OIA-1946391, NSF 1920920, NSF FAIN-2223793 and NIH 1R01CA277739.

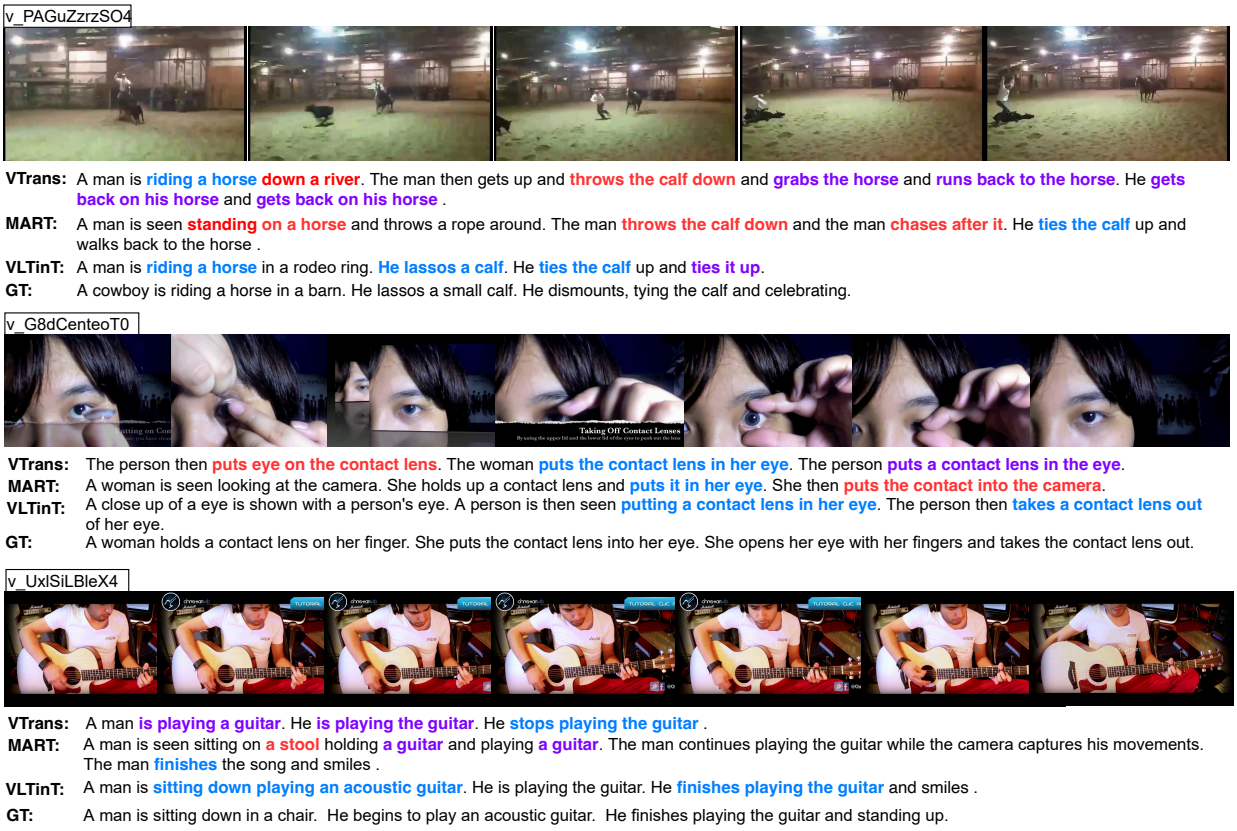


Fig. 5: Qualitative comparison on ActivityNet Captions *ae-test* split between our VLTinT and VTrans [50], MART [16]. At each video, captioning from VTrans is in the 1^{st} row, MART is in the 2^{nd} row, our VLTinT is in the 3^{rd} row, and groundtruth (GT) is in the 4^{th} row. Red text indicates the captioning mistakes, purple text indicates repetitive patterns, and blue text indicates some distinct expressions. We compared our model with Vanilla Transformer (VTrans) and MART as baselines. GT indicates the groundtruth captioning.

REFERENCES

[1] P. Pareek and A. Thakkar, "A survey on video-based human action recognition: recent updates, datasets, challenges, and applications," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 2259–2322, 2021.

[2] D.-Q. Vu, N. Le, and J.-C. Wang, "Teaching yourself: A self-knowledge distillation approach to action recognition," *IEEE Access*, vol. 9, pp. 105 711–105 723, 2021.

[3] L. Wang, Z. Tong, B. Ji, and G. Wu, "Tdn: Temporal difference networks for efficient action recognition," in *CVPR*, 2021, pp. 1895–1904.

[4] D. Q. Vu, N. T. Le, and J.-C. Wang, "Self-supervised learning via multi-transformation classification for action recognition," *arXiv preprint arXiv:2102.10378*, 2021.

[5] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang, and J. Liu, "Human action recognition from various data modalities: A review," *IEEE TPAMI*, 2022.

[6] D.-Q. Vu, N. T. Le, and J.-C. Wang, "(2+ 1) d distilled shufflenet: A lightweight unsupervised distillation network for human action recognition," in *ICPR*, 2022, pp. 3197–3203.

[7] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, "Bsn: Boundary sensitive network for temporal action proposal generation," in *ECCV*, 2018.

[8] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, "Bmn: Boundary-matching network for temporal action proposal generation," in *ICCV*, 2019.

[9] C. Lin, J. Li, Y. Wang, Y. Tai, D. Luo, Z. Cui, C. Wang, J. Li, F. Huang, and R. Ji, "Fast learning of temporal action proposal via dense boundary generator," *AAAI*, pp. 11 499–11 506, Apr. 2020.

[10] V.-K. Vo-Ho, N. Le, K. Kamazaki, A. Sugimoto, and M.-T. Tran, "Agent-environment network for temporal action proposal generation," in *ICASSP*, 2021, pp. 2160–2164.

[11] K. Vo, K. Yamazaki, S. Truong, M.-T. Tran, A. Sugimoto, and N. Le, "Abn: Agent-aware boundary networks for temporal action proposal generation," *IEEE Access*, vol. 9, pp. 126 431–126 445, 2021.

[12] K. Vo, H. Joo, K. Yamazaki, S. Truong, K. Kitani, M.-T. Tran, and N. Le, "Aei: Actors-environment interaction with adaptive attention for temporal action proposals generation," in *BMVC*, 2021.

[13] K. Vo, S. Truong, K. Yamazaki, B. Raj, M.-T. Tran, and N. Le, "Aoe-net: Entities interactions modeling with adaptive attention mechanism for temporal action proposals generation," *IJCV*, Oct 2022.

[14] M. Xu, C. Zhao, D. S. Rojas, A. Thabet, and B. Ghanem, "G-tad: Sub-graph localization for temporal action detection," in *CVPR*, 2020.

[15] R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, and C. Gan, "Graph convolutional networks for temporal action localization," in *ICCV*, 2019, pp. 7094–7103.

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

[17] Z. Dai, Z. Yang *et al.*, "Transformer-XL: Attentive language models beyond a fixed-length context," in *ACL*, 2019, pp. 2978–2988.

[18] K. Yamazaki, K. Vo, S. Truong, B. Raj, and N. Le, "VLTinT: Visual-Linguistic Transformer-in-Transformer for Coherent Video Paragraph Captioning," *arXiv e-prints*, p. arXiv:2211.15103, Nov. 2022.

[19] V. Gabeur, C. Sun, K. Alahari, and C. Schmid, "Multi-modal transformer for video retrieval," in *ECCV*. Springer, 2020, pp. 214–229.

[20] M. Wray, H. Doughty, and D. Damen, "On semantic similarity in video retrieval," in *CVPR*, 2021, pp. 3650–3660.

[21] B. G. Fabian Caba Heilbron, Victor Escorcia and J. C. Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *CVPR*, 2015, pp. 961–970.

[22] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar, "THUMOS challenge: Action recognition with a large number of classes," 2014.

[23] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles, "Dense-captioning events in videos," in *ICCV*, 2017, pp. 706–715.

[24] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE TPAMI*, vol. 35, pp. 221–231, 2013.

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

[26] W. Kay, J. Carreira, *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[27] H. Su, W. Gan, W. Wu, J. Yan, and Y. Qiao, "BSN++: complementary boundary regressor with scale-balanced relation modeling for temporal action proposal generation," in *ACCV*, 2020.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*. Curran Associates, Inc., 2017.

[29] M. Malinowski, C. Doersch, A. Santoro, and P. Battaglia, "Learning visual question answering by bootstrapping hard attention," in *ECCV*, 2018, pp. 3–20.

[30] A. Radford, J. W. Kim *et al.*, "Learning transferable visual models from natural language supervision," *arXiv preprint arXiv:2103.00020*, 2021.

[31] S. Liu, X. Zhao, H. Su, and Z. Hu, "Tsi: Temporal scale invariant network for action proposal generation," in *ACCV*, November 2020.

[32] Y. Bai, Y. Wang, Y. Tong, Y. Yang, Q. Liu, and J. Liu, "Boundary content graph neural network for temporal action proposal generation," in *ECCV*. Springer, 2020, pp. 121–137.

[33] J. Tan, J. Tang, L. Wang, and G. Wu, "Relaxed transformer decoders for direct action proposal generation," *ICCV*, 2021.

[34] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NIPS*, ser. NIPS'14, 2014, p. 568–576.

[35] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *ICCV*, October 2019.

[36] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

[37] A. Dosovitskiy, L. Beyer *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *CVPR*, 2021.

[38] K. Yamazaki, S. Truong, K. Vo, M. Kidd, C. Rainwater, K. Luu, and N. Le, "Vlcap: Vision-language with contrastive learning for coherent video paragraph captioning," in *ICIP*. IEEE, 2022, pp. 3656–3661.

[39] Y. Song, S. Chen, and Q. Jin, "Towards diverse paragraph captioning for untrimmed videos," in *CVPR*, 2021, pp. 11 245–11 254.

[40] X. Dai, B. Singh, G. Zhang, L. S. Davis, and Y. Qiu Chen, "Temporal context network for activity localization in videos," in *ICCV*, Oct 2017.

[41] T. Yao, Y. Li, Z. Qiu, F. Long, Y. Pan, D. Li, and T. Mei, "Msr asia msm at activitynet challenge 2017: Trimmed action recognition, temporal action proposals and densecaptioning events in videos," in *CVPRW*, 2017.

[42] S. Buch, V. Escorcia, B. Ghanem, L. Fei-Fei, and J. C. Niebles, "End-to-end, single-stream temporal action detection in untrimmed videos," in *BMVC*, 2017.

[43] J. Gao, K. Chen, and R. Nevatia, "Ctap: Complementary temporal action proposal generation," in *ECCV*, September 2018.

[44] H. Eun, S. Lee, J. Moon, J. Park, C. Jung, and C. Kim, "Srg: Snippet relatedness-based temporal action proposal generator," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2019.

[45] Y. Liu, L. Ma, Y. Zhang, W. Liu, and S.-F. Chang, "Multi-granularity generator for temporal action proposal," in *CVPR*, June 2019.

[46] P. Zhao, L. Xie, C. Ju, Y. Zhang, Y. Wang, and Q. Tian, "Bottom-up temporal action localization with mutual regularization," in *ECCV*. Springer, 2020, pp. 539–555.

[47] X. Wang, S. Zhang, Z. Qing, Y. Shao, C. Gao, and N. Sang, "Self-supervised learning for semi-supervised temporal action proposal," in *CVPR*, 2021, pp. 1905–1914.

[48] Z. Qing, H. Su, W. Gan, D. Wang, W. Wu, X. Wang, Y. Qiao, J. Yan, C. Gao, and N. Sang, "Temporal context aggregation network for temporal action proposal refinement," in *CVPR*, 2021, pp. 485–494.

[49] J. Zheng, D. Chen, and H. Hu, "Boundary adjusted network based on cosine similarity for temporal action proposal generation," *Neural Processing Letters*, pp. 1–16, 2021.

[50] L. Zhou, Y. Zhou *et al.*, "End-to-end dense video captioning with masked transformer," in *CVPR*, 2018, pp. 8739–8748.

[51] S. Ging, M. Zolfaghari *et al.*, "COOT: cooperative hierarchical transformer for video-text representation learning," in *NIPS*, 2020.

[52] K. Papineni, S. Roukos *et al.*, "Bleu: a method for automatic evaluation of machine translation," in *ACL*, 2002, pp. 311–318.

[53] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Workshop on Statistical Machine Translation*, 2014, pp. 376–380.

[54] R. Vedantam, C. L. Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *CVPR*, 2015, pp. 4566–4575.

[55] R. Shetty, M. Rohrbach *et al.*, "Speaking the same language: Matching machine to human captions by adversarial training," in *CVPR*, 2017.

[56] Y. Xiong, B. Dai, and D. Lin, "Move forward and tell: A progressive generator of video descriptions," in *ECCV*, 2018.

[57] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NeurIPS*, 2015.

[58] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, September 2014.

[59] T. Wang, R. Zhang *et al.*, "End-to-end dense video captioning with parallel decoding," in *ICCV*, 2021, pp. 6827–6837.