AAN : Attributes-Aware Network for Temporal Action Detection

Rui Dai, Srijan Das, Michael Ryoo, Francois Bremond

AAN: Attributes-Aware Network for Temporal Action Detection

Rui Dai*[1]
dairui01@hotmail.com
Srijan Das[2]
sdas24@charlotte.edu
Michael S. Ryoo[3]
mryoo@@cs.stonybrook.edu
François Brémond[1]
francois.bremond@inria.fr

[1] Inria, Université Côte d'Azur, France
[2] UNC Charlotte, USA
[3] Stony Brook University, USA

Abstract

The challenge of long-term video understanding remains constrained by the efficient extraction of object semantics and the modelling of their relationships for downstream tasks. Although OpenAI's CLIP visual features exhibit discriminative properties for various vision tasks, particularly in object encoding, they are suboptimal for long-term video understanding. To address this issue, we present the **Attributes-Aware Network** (AAN), which consists of two key components: the Attributes Extractor and a Graph Reasoning block. These components facilitate the extraction of object-centric attributes and the modelling of their relationships within the video. By leveraging CLIP features, AAN outperforms state-of-the-art approaches on two popular action detection datasets: **Charades** and **Toyota Smarthome Untrimmed** datasets.

1 Introduction

In video understanding, temporal action detection is one of the ultimate tasks that automatically detects human actions in videos along with classifying them. The deep learning revolution has been a huge driving force for the advancements in the video understanding domain. Despite the progress of action recognition algorithms in trimmed videos [2, 4, 11, 15, 38], the majority of real-world videos are lengthy and untrimmed with dense regions of interest [10, 35]. In these regions (temporal intervals), most actions involve human interaction with objects, such as *opening "fridge"*, *taking "ham"*, and *cutting "bread"*. These objects and their state changes are crucial attributes to understand human actions performed in videos. Thus, modelling these fine-grained object semantics for detecting actions is paramount in complex activities such as *making breakfast* or *furniture assembling*.

Many successful action detection models have been developed to process untrimmed videos in two stages [8, 9, 29, 30]. In the initial stage, frame-level features are extracted

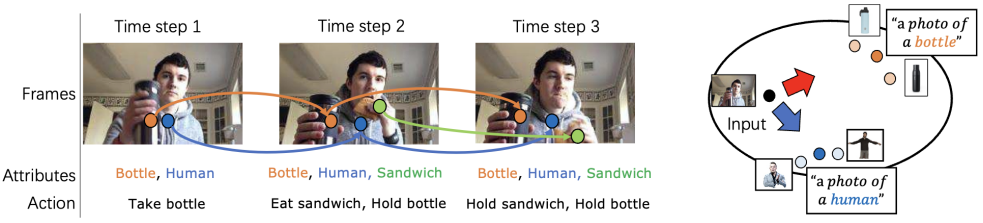
* This work was performed while Rui was at Inria as a student.

Figure 1: CLIP image classification for the video frames. A list of daily living attributes (i.e., objects) is used as labels, i.e, the inputs to the CLIP Text model. We show the top 5 highest similarity attributes for two example frames (below) along with the action label (top right). We find that the CLIP Image features effectively preserve the action-relevant object semantics of an image.

from visual input using a 3D convolutional network [4] that has been pre-trained on extensive video datasets like Kinetics [22]. The subsequent stage entails modeling temporal relationships among the frame features to detect activities. Nevertheless, these approaches do not explicitly encode object semantics. On one hand, limited by the absence of annotations of the relevant objects involved in action due to constrained labeling, the frame-level features may not preserve the object semantics relevant to the target actions in the first stage. On the other hand, the second stage is dedicated exclusively to temporal representation learning across the frame features. Some methods [14, 40] have attempted to enhance action understanding by employing object detectors and subsequently incorporating a reasoning module that operates on the extracted objects for action detection. While these frameworks are capable of efficiently extracting object semantics, the accuracy of action prediction is heavily dependent on the precision of object detection. Furthermore, the inclusion of an object detector introduces a trade-off. Object detectors are known for their large model complexity, often leading to increased computation costs during inference. Furthermore, methods using object detectors leverage region of interest (ROI) operations on intermediate 3D convolution features to optimize object detection [40]. Nevertheless, this technique typically operates on a restricted temporal data sequence, potentially restricting the model's capability to capture short-term relationships.

In pursuit of a dense understanding of scene, vision-language models, specifically OpenAI's CLIP [31], have demonstrated remarkable efficacy in pre-training image and text encoders for a variety of downstream tasks. Inspired by CLIP's success, numerous models have been pre-trained using large-scale open-vocabulary data comprising image-text pairs, resulting in a joint feature space for image and language [31, 33, 41]. As the pre-training process is not limited to a predefined set of object labels, the visual representations obtained are aligned with a more extensive range of "language" semantics [27]. Due to this configuration, CLIP features retain a richer object semantics (refer to Fig. 1). Consequently, in this paper, we explore the question: *How can we leverage CLIP features for fine-grained action detection?*

To this end, we propose the Attributes-Aware Network to address the challenge of multi-label action detection. This network is composed of two modules: Firstly, an attribute extractor that learns to extract attribute semantics from the frame-level features obtained from



(a) Fine-grained Action Video (b) CLIP feature space

Figure 2: On the left hand, there is an example of fine-grained actions in a video. Multiple actions can occur at the same time and actions can involve different objects (attributes). On the right hand, we show an example of the feature space of CLIP. In this space, the representation of the text sentence and image with the same semantics are close to each other.

the CLIP. Different from the existing object-centric video recognition methods [14, 17, 40], our method does not rely on the prior of the object detectors but leverages the joint visual and language space of CLIP features. The extracted attributes are relevant objects (e.g., knife) for an action (e.g., cutting) and the semantics of each attribute form nodes of a graph. Secondly, we introduce an attention-based graph reasoning module that models the inter-attribute relations and temporal relations of attributes within a video for frame-level action prediction.

To summarize, our contribution mainly lies in three folds. Firstly, we propose a module that extracts relevant object semantics of image frames using the CLIP. This module disentangles target information from the shared space between visual and semantic features. Secondly, we introduce an attention-based block to model the complex attribute relations within and across frames. Finally, our proposed method outperforms state-of-the-art approaches on two challenging multi-label action detection datasets. To our knowledge, we are the first to disentangle CLIP embeddings for the purpose of long-term video understanding.

2 Related Work

In this section, we survey existing approaches that model relations between different semantics for action understanding.

Lan et al. [25] propose a method that represents videos by a hierarchy of mid-level action elements in an unsupervised manner. Each action element corresponds to a spatio-temporal segment in the video and can represent actions at multiple spatio-temporal resolutions. Sigurdsson et al. [36] propose a fully-connected temporal CRF model for reasoning over the variant intent of videos, where the intent is defined as the clustering of similar activities (e.g., actions, objects) in a video. Although these approaches can structure the video using semantics, they do not explicitly learn the temporal structure nor are they learned in an end-to-end fashion.

In recent years, Wang et al. [40] proposed video as graphs where the nodes are based on object proposals. Likewise, Ghosh et al. [14] utilized the labels of the object bounding boxes to form fine-grained graphs of humans, scenes and objects for each image frame. Guermal et al. [17] extract object-specific feature descriptors for each object using an object

detector and learn action correlations using an attention mechanism. While these methods better characterize complex object-based actions in videos, they rely on object detectors pre-trained on a predefined set of object categories, which limits their ability to handle unseen objects and increases the computation complexity at both training and inference time.

More recently, several methods [3, 20, 28, 39] have used CLIP features for video understanding. However, these methods are designed to handle short temporal videos, and the challenge of handling actions over a long range of time for solving the task of action detection still persists. Towards long-term video understanding, Tirupattur et al. [37] introduced MLAD that can explore the action-temporal relations with a set of self-attention layers: an inter-class attention map for every time step and an inter-time attention map for every action class. Similarly, Dai [7] propose CTRN that can model the interaction relations via graph neural networks. However, both methods overlook object attributes, limiting their performance over object-dominated actions. In contrast, this paper proposes a method that learns relations among attributes (i.e., objects) extracted from CLIP features. To the best of our knowledge, this is the first method that leverages CLIP features for long-term video understanding while implicitly modelling object attributes.

Besides, model relations between different semantics. There are also some works using pure temporal modelling for temporal action detection [6, 9, 29]. TGM [29] is a temporal filter based on Gaussian distributions, which enables the learning of longer temporal structures with a limited number of parameters. PDAN [8] is a temporal convolutional network, with temporal kernels which are adaptive to the input data. MSTCT [9] uses convolutions in a token-based architecture to promote multiple temporal scales of tokens, and to blend neighbouring tokens imposing a temporal consistency with ease. Different from the above methods, besides the temporal modelling, our proposed method further models the object semantic relation for a better understanding of the video content.

3 Proposed Method

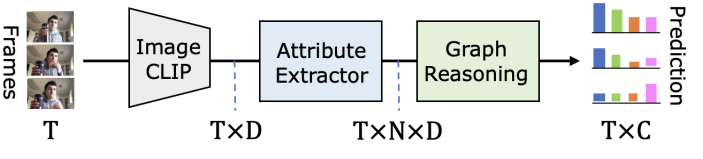


Figure 3: Overall framework for the proposed Attribute-Aware Network. This network is composed of 3 main components: CLIP encoder, attribute extractor and graph reasoning block for graph classification.

In this section, we present our **Attributes-Aware Network (AAN)**, which leverages CLIP features for the task of action detection. AAN is composed of three main components: a frozen CLIP encoder, an attributes extractor and a graph reasoning block (see Fig. 3). The attributes extractor extracts the attribute semantics from the frame-level CLIP features. Conversely, the graph reasoning block models the attribute relation and performs graph classification. These two components of AAN are trained end-to-end to optimize the attribute representation for the action detection task. We elaborate on these components in the following.

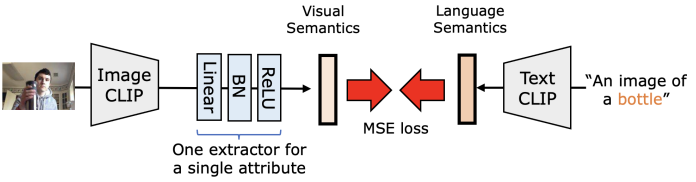


Figure 4: Attributes Extractor. For each attribute, it has a specific extractor to obtain its semantic representation from the input frame. The extractor for the same attribute is shared across the frames.

3.1 Attributes Extraction

The attributes extractor is used to extract the attributes from the frames. These attributes are object semantics encoded within the frame representations. Different from the previous works [14, 40] that extract the object semantics via object detector from the 3D convolutional feature map, in this work, the attributes extraction is based on the frame level representation obtained from CLIP encoder.

CLIP [31] is trained using a vast collection of image-text pairs in a contrastive fashion. Owing to the CLIP pre-training process, language and image features are projected into a shared embedding space, as depicted in Fig.2(b). This configuration enables the CLIP Image model to compute more generalized visual features, as demonstrated in [16, 34, 39] for a range of vision tasks. As illustrated in Fig. 1, CLIP Image features effectively retain the action-relevant object semantics within an image. However, CLIP Image features consist of a blend of various object representations. This raises the question: *how can we disentangle object semantics from the CLIP Image feature for long-term video understanding*?

In the joint embedding space of CLIP, when given a text prompt describing a specific object, such as "a photo of a *bottle*", the feature embedded by the Text CLIP encoder conveys the pure semantics of "bottle" in this space. This feature is closely related to images of "bottle" embedded by the Image CLIP model (refer to Fig. 2(b)). These prompts can be considered as anchors within the joint space, allowing the disentanglement of particular object semantics from the holistic image representation. Therefore, in this work, the semantics for a specific object are computed by minimizing the Euclidean distance between the visual frame feature and the text anchor representation in the shared semantic space.

In this work, our focus is on indoor environments. Initially, we pre-define N attributes associated with daily living actions. Since action labels typically consist of a noun and a verb, the attributes should encompass all nouns in the datasets. Next, we generate prompts based on these attributes and employ the CLIP Text encoder to extract text anchor features in the joint embedding space, represented by $\mathcal{T}^n$, where $n \in N$.

In the visual aspect, frame-level features are obtained using the CLIP Image encoder (refer to Fig. 4). These extracted features are stacked along the temporal axis (i.e., frames) to create a $T \times D_0$ video representation. This video representation is then fed to the attributes extractor, which comprises N filters. Each filter corresponds to a specific attribute and includes a linear layer, batch normalization, and ReLU activation, as follows:

$$I_t^n = ReLU(BN(W^n F_t)) \tag{1}$$

Here, F_t represents the CLIP feature of the frame at time step t, and W^n denotes the linear

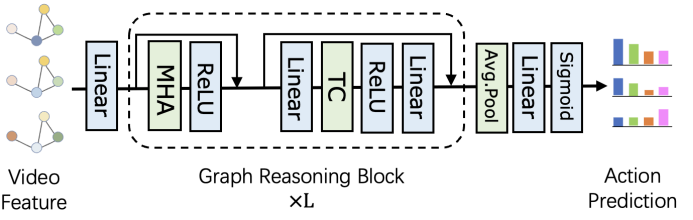


Figure 5: Graph Classification.

layer for attribute n. ReLU and BN refer to ReLU non-linear activation and batch normalization, respectively. Note that each filter signifies an object semantics in the latent space. We minimize the L2 distance [1] between the output feature (I_t^n) of the attributes extractor and its corresponding text anchor feature ($\mathcal{T}^n$). This minimization objective encourages the extractor to extract the object-specific attributes in an image. The formulation of this objective is as:

$$\mathcal{L}_{attributes} = \frac{1}{NT} \sum_{t \in T} \sum_{n \in N} ||I_t^n - \mathcal{T}^n||^2 \tag{2}$$

Therefore, the linear layers situated above the frozen CLIP visual encoder, optimized with the aforementioned loss, facilitate the extraction of relevant attributes (object-related information) pertaining to the scene for video understanding. Note that, it is possible to utilize both MSE loss and contrastive loss [18] for semantic extraction. In our work, we have explored the use of contrastive loss but found that MSE loss (L2) performs better for our specific task. This observation aligns with our intuition, which suggests that contrastive learning is more suitable for large-scale pre-training rather than fine-tuning downstream tasks. Balancing embedding learning with downstream task classification can be challenging, making MSE loss a more effective choice for our purposes.

3.2 Graph Reasoning Block

Following the attribute extraction, each frame can be represented as a graph, where the extracted attributes function as the graph nodes. The graph edges are initialized based on the statistics of attribute occurrences in the training distribution. An action label consists of a noun and a verb. The occurrence of attributes is determined by the co-occurring probability of the nouns. G_{ij} represents the concurrent instances for attribute classes n_i and n_j. The conditional probability matrix $P_{ij} = P(n_j|n_i)$ is then calculated as $P_{ij} = G_{ij}/G_i$, where G_i denotes the frequency of n_i in the training distribution, and $P_{ij} \in \mathbb{R}^{N \times N}$ indicates the probability of class n_j given the simultaneous occurrence of n_i. The computed P represents the initial graph edges. All graphs (i.e., frames) within the video share the same graph initialization settings. The action prediction for each frame can be interpreted as a graph classification task.

In this work, our graph classification mechanism is close to Self-Attention Graph Pooling (SAGPool) [26]. However, in contrast to [26], we perform graph reasoning on a series of frames, which includes modelling both attribute-attribute and attribute-temporal relations. As shown in Fig. 5, this block comprises L attention-based graph reasoning layers followed by a pooling layer to aggregate the graph-level information. Each graph reasoning block

contains a multi-head graph convolution layer, a Temporal Convolution layer, and a linear layer with non-linear activations and residual links.

The input feature is first fed into a linear layer (i.e., bottleneck layer) to squeeze the feature dimension from D_0 to D_1. Then the features are fed to the graph reasoning blocks. In each block, self-attention-based graph convolution is applied on each frame. The graph convolution is used for modeling the relationship among the attributes at each frame. In practice, for block i at frame t, the input features is $X_t^i \in \mathbb{R}^{D_1 \times 1 \times N}$. Then we utilize a self-attention module to learn the inter-attributes relation. Moreover, the computed relation is then summed up with the initial graph edge based on the co-occurring attributes probability matrix P. The obtained graph adjacency matrix is formulated as follows:

$$A_t^i = softmax((W_1^i X_t^i)^\top W_2^i X_t^i) + P \tag{3}$$

where W_1^i and $W_2^i \in \mathbb{R}^{D_1 \times D_1}$ are the weights of two linear layers. Each value in A_t can be seen as a composite edge between two vertices. P represents the global co-occurrence statistics from the training set and the self-attention mask represents the relation that is adaptive to different frames. Finally, the graph convolutional operation is performed using the formulation from [24]:

$$X_t'^i = ReLU(A_t^i X_t^i W_3^i) + X_t^i \tag{4}$$

where $W_3^i \in \mathbb{R}^{D_1 \times D_1}$ is the weight of the linear layer. The output feature of the graph convolution is then fed into a linear layer, followed by a temporal convolution. The linear layer is employed for channel mixing prior to applying the temporal convolution. Temporal convolution is performed for the same node across multiple frames to model temporal information. This is followed by a ReLu activation and an additional linear layer. These operations can be expressed as:

$$X^{i+1} = W_5^i(ReLU(TC(W_4^i X'^i))) + X'^i \tag{5}$$

where W_4^i and $W_5^i \in \mathbb{R}^{D_1 \times D_1}$ are the weights of two linear layers, while TC denotes the temporal convolution operation. The output X^{i+1} is input to the subsequent graph reasoning blocks.

In the final step, graph classification is carried out by classifying the aggregated graph representation for each frame. The graph aggregation is performed using an average pooling layer, while the classification of the graph employs a linear layer with sigmoid activation. The Binary Cross Entropy (BCE) loss, $\mathcal{L}_{action}$, is computed in comparison to the ground-truth labels for multi-label action classification of each frame, as demonstrated in [5]. Consequently, the total loss for training AAN is given by:

$$\mathcal{L}_{total} = \mathcal{L}_{attributes} + \mathcal{L}_{action} \tag{6}$$

4 Experiment

4.1 Dataset

In this work, we evaluate our method on two challenging action detection datasets which involve actions with fine-grained object details. **Charades** [35] is a large untrimmed dataset with 9848 videos of daily living actions. The dataset contains 157 action classes with more

than 30 objects shared across multiple action classes. We utilize the "action localization" setting for this dataset, which aims at detecting actions for different frames [36]. We also evaluate our method on **Toyota Smarthome Untrimmed (TSU)** [10] (Cross-subject protocol). Similar to the Charades, TSU is recorded in an indoor environment. There are up to 5 actions that can occur at the same time in a given frame. Different from Charades, the TSU involves long-term videos and composite activities. For evaluation, we compute the per-frame mAP by default on these two datasets following [43].

4.2 Implementation details

In the proposed network, we employ ViT/14 [12] based CLIP visual encoder. The CLIP encoding feature size D_0 is 768. For light-weighting the network, we then map the D_0 to intermediate channel size D_1, which is 256. We set the number of attributes N to 38 to fit the requirement of general daily living video understanding. There are $L = 5$ graph reasoning blocks used, and the kernel size for the temporal convolution within the graph reasoning block is 3. The number of heads for the multi-head attention is set to 4. AAN is trained using two RTX 6000 GPUs with a batch size of 32. The Adam optimizer [23] is utilized with an initial learning rate of 0.0001, which is scaled by a factor of 0.5 with a patience of 8 epochs.

Prompt: In our work, we perform feature extraction on a per-frame setting. As a result, we utilize a standard image prompt for image classification. Specifically, during the training phase, we define a list of prompts including (1) "*A photo of* **xx**", (2) "*There is a* **xx**", (3) "*An image of* **xx**", and (4) "*A photo with a* **xx**", where **xx** represents the object label. To enhance the robustness of attribute representation, we randomly select one of these prompts for each video during training. During inference, we use prompt (1) to extract attributes.

Attributes: The predefined attributes in our study are derived from the object and action labels (e.g. "*book*" in "*reading book*") provided by the Charades and TSU datasets. Additionally, both datasets include a list of objects present in their respective datasets, and we leverage this information to compile our attribute list.

4.3 Comparison to the State-of-the-Art

In this section, we compare AAN with state-of-the-art methods on two large indoor datasets, Charades [35] and TSU [10], as shown in Table 1. Note that, similar to our method, we compared only the RGB only result. Both datasets feature complex actions with varying object interactions. We compare our approach with leading methods for these two datasets, including techniques utilizing TCN [8], Transformer [32, 37], graph convolution [7], and ConvTransformer [9]. We observe that our method significantly outperforms state-of-the-art approaches (e.g., +3.2% on Charades and +7.4% on TSU compared to MS-TCT [9]). This marks the first time a method achieves 30% in the localization task on the Charades dataset and 40% on the TSU dataset. In our model analysis, we demonstrate that this substantial improvement in action detection performance is not solely due to the use of the CLIP visual encoder in comparison to I3D [4] or X3D [13] encoders employed in state-of-the-art methods. Rather, it is attributed to each component of AAN, which plays a crucial role in leveraging the CLIP features for the action detection task.

Additionally, we assess the performance of our method using the action dependency metrics [37] on the Charades dataset. As depicted in Table 2, our method surpasses MLAD

Eval in per-frame mAP (%)	Charades	TSU
R-C3D [44]	12.7	8.7
Super-event [30]	18.6	17.2
TGM [25]	20.6	26.7
PDAN [8]	23.7	32.7
Coarse-Fine [45]	25.1	-
MLAD [53]	18.4	-
CTRN [9]	25.3	33.5
MS-TCT [9]	25.4	33.7
Coarse-fine + SSDet [24]	26.9	-
ViVit-L + TTM [3]	28.8	-
Attribute-Aware Network	**32.0**	**41.3**

Table 1: Comparison with the state-of-the-art methods.

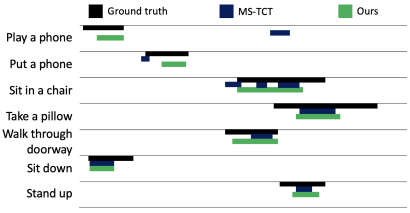


Figure 6: Visualization of action detection.

	$\tau = 0$			$\tau = 20$			$\tau = 40$		
	P_{AC}	$F1_{AC}$	mAP_{AC}	P_{AC}	$F1_{AC}$	mAP_{AC}	P_{AC}	$F1_{AC}$	mAP_{AC}
I3D [4]	14.3	2.1	15.2	12.7	2.9	21.4	14.9	3.1	20.3
MLAD [53]	19.3	8.9	28.9	18.9	10.5	35.7	19.6	10.8	34.8
MS-TCT [9]	26.3	19.5	30.7	27.6	22.1	37.6	27.9	22.1	36.4
Attribute-Aware Network	**31.4**	**20.4**	**35.4**	**30.4**	**22.3**	**41.8**	**32.5**	**22.2**	**40.8**

Table 2: Evaluation on the Charades dataset using the action-conditional metrics [37].

and MS-TCT across all metrics for both co-occurring action detection ($\tau = 0$) and distant action detection ($\tau = 40$). This demonstrates the robustness of our proposed approach.

4.4 Further Analysis

4.4.1 Ablation

Model Analysis. In this section, we examine the effectiveness of each component in AAN on the Charades dataset. Table 3 demonstrates our analysis of the necessity of the Attribute Extractor and the Graph Reasoning block. Note that the graph reasoning block can not operate without the attribute extractor. As seen in the table, merely extracting object attributes from the CLIP feature does not significantly enhance performance (+1.4%). However, performing reasoning across these extracted attributes leads to better modelling of complex videos (+14.0%).

Furthermore, we assess the importance of components within the Graph Classification block: Multi-Head Attention (MHA) and Temporal Convolution. These results are computed based on the presence of the Attribute Extractor. Table 4 reveals that adding MHA or Temporal Convolution both improve performance compared to the vanilla Attribute Extractor (+7.0% and +8.3%, respectively). By incorporating all components, AAN achieves a substantial improvement.

Backbone. As our method is built on top of the CLIP backbone (i.e., pre-trained ViT model [12]) rather than the conventional I3D model [4]. We thus further analyse if the performance boost is principal because of changing the backbone network. As shown in Table 5, we first evaluate the state-of-the-art method PDAN [8] and MS-TCT [9] with the pre-trained ViT backbone (i.e., CLIP Image model [51]). We find that while using the same backbone as our method, the performance of PDAN and MS-TCT can be improved. However, as PDAN and MS-TCT do not have a specific design for leveraging the object-related

Attribute Extractor	Graph Reasoning Blocks	mAP (%)
		18.1
✓		19.4
✓	✓	32.0

Table 3: Ablation on the Proposed Modules.

Multi-Head Attention	Temporal Conv.	mAP (%)
✓		25.1
	✓	26.4
✓	✓	32.0

Table 4: Ablation inside graph classification module.

	MLP	PDAN [8]	MS-TCT [2]	Ours
I3D [4]	15.6	23.7	25.4	-
ViT [31]	19.0	26.0	29.7	32.0

Table 5: Ablation on the visual backbone.

feature, our method can still perform better (+2.3%).

We further compared our method and MS-TCT in terms of per-action class precision on Charades. We observe that for 22.9% action classes, our method outperforms MS-TCT for more than 5%. The top-5 actions that outperform MS-TCT are: Closing a window (+37.2%), Sitting on a chair (+26.4%), Taking a broom (+23.6%), Closing a fridge (+22.0%), Putting a laptop (+18.9%). All the classes are relevant to objects.

4.4.2 Qualitative analysis

As shown in Fig. 6, we visualize the predictions of our method and the state-of-the-art method MS-TCT on a sample video from the Charades dataset. Both methods employ the same backbone network. We observe that, in comparison to MS-TCT, our method is more proficient in predicting object-related actions, such as *play a phone*, *put a phone*, and *sit in a chair*.

5 Conclusion

In this paper, we introduced the Attributes-Aware Network (AAN), which utilizes CLIP features for action detection tasks. AAN comprises two essential components: the Attributes Extractor and the Graph Reasoning block, which are vital for learning object semantics and modelling their relationships in videos. AAN surpasses previous state-of-the-art methods on two widely-used Activities of Daily Living datasets, establishing a new benchmark. Future research will focus on rethinking various vision tasks using CLIP features and AAN-style frameworks.

Acknowledgement

This work has been supported by the French government, through the 3IA Cote d'Azur Investments in the Future project managed by the National Research Agency (ANR) with the reference number ANR-19-P3IA-0002. This work was also supported in part by the National Science Foundation (IIS-2245652). The authors are also grateful to the OPAL infrastructure from Université Côte d'Azur for providing resources and support.

References

[1] David M Allen. Mean square error of prediction as a criterion for selecting variables. *Technometrics*, 13(3):469–475, 1971.

[2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691*, 2021.

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. A clip-hitchhiker's guide to long video retrieval, 2022.

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733. IEEE, 2017.

[5] Tianshui Chen, Muxin Xu, Xiaolu Hui, Hefeng Wu, and Liang Lin. Learning semantic-specific graph representation for multi-label image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 522–531, 2019.

[6] Rui Dai, Luca Minciullo, Lorenzo Garattoni, Gianpiero Francesca, and François Bremond. Self-attention temporal convolutional network for long-term daily living activity detection. In *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–7. IEEE, 2019.

[7] Rui Dai, Srijan Das, and Francois F Bremond. CTRN: Class Temporal Relational Network For Action Detection. In *BMVC 2021 - The British Machine Vision Conference*, Virtual, United Kingdom, November 2021.

[8] Rui Dai, Srijan Das, Luca Minciullo, Lorenzo Garattoni, Gianpiero Francesca, and Francois Bremond. Pdan: Pyramid dilated attention network for action detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2970–2979, 2021.

[9] Rui Dai, Srijan Das, Kumara Kahatapitiya, Michael Ryoo, and Francois Bremond. Ms-tct: Multi-scale temporal convtransformer for action detection. In *CVPR*, 2022.

[10] Rui Dai, Srijan Das, Saurav Sharma, Luca Minciullo, Lorenzo Garattoni, Francois Bremond, and Gianpiero Francesca. Toyota smarthome untrimmed: Real-world untrimmed videos for activity detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2022. doi: 10.1109/TPAMI.2022.3169976.

[11] Srijan Das, Rui Dai, Di Yang, and Francois Bremond. Vpn++: Rethinking video-pose embeddings for understanding activities of daily living. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[13] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition, 2020.

[14] Pallabi Ghosh, Yi Yao, Larry S Davis, and Ajay Divakaran. Stacked spatio-temporal graph convolutional networks for action segmentation. *arXiv preprint arXiv:1811.10575*, 2018.

[15] Rohit Girdhar, João Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. *CoRR*, abs/1812.02707, 2018. URL http://arxiv.org/abs/1812.02707.

[16] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021.

[17] Mohammed Guermal, Rui Dai, and François Brémond. THORN: Temporal Human-Object Relation Network for Action Recognition. *ICPR*, 2022.

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[19] Kumara Kahatapitiya and Michael S Ryoo. Coarse-fine networks for temporal activity detection in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8385–8394, 2021.

[20] Kumara Kahatapitiya, Anurag Arnab, Arsha Nagrani, and Michael S. Ryoo. Victr: Video-conditioned text representations for activity recognition, 2023.

[21] Kumara Kahatapitiya, Zhou Ren, Haoxiang Li, Zhenyu Wu, Michael S Ryoo, and Gang Hua. Weakly-guided self-supervised pretraining for temporal activity detection. In *AAAI Conference on Artificial Intelligence*, 2023.

[22] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[23] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. URL http://arxiv.org/abs/1412.6980.

[24] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[25] Tian Lan, Yuke Zhu, Amir Roshan Zamir, and Silvio Savarese. Action recognition by hierarchical mid-level action elements. In *Proceedings of the IEEE international conference on computer vision*, pages 4552–4560, 2015.

[26] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *International conference on machine learning*, pages 3734–3743. PMLR, 2019.

[27] Ziyi Lin, Shijie Geng, Renrui Zhang, Peng Gao, Gerard de Melo, Xiaogang Wang, Jifeng Dai, Yu Qiao, and Hongsheng Li. Frozen clip models are efficient video learners. In *European Conference on Computer Vision*, pages 388–404. Springer, 2022.

[28] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. CLIP4Clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*, 2021.

[29] A. Piergiovanni and M. S. Ryoo. Temporal gaussian mixture layer for videos. In *International Conference on Machine Learning (ICML)*, 2019.

[30] AJ Piergiovanni and Michael S Ryoo. Learning latent super-events to detect multiple activities in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[32] Michael S. Ryoo, Keerthana Gopalakrishnan, Kumara Kahatapitiya, Ted Xiao, Kanishka Rao, Austin Stone, Yao Lu, Julian Ibarz, and Anurag Arnab. Token turing machines, 2023.

[33] Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can clip benefit vision-and-language tasks? *arXiv preprint arXiv:2107.06383*, 2021.

[34] Hengcan Shi, Munawar Hayat, Yicheng Wu, and Jianfei Cai. Proposalclip: Unsupervised open-category object proposal generation via exploiting clip cues. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9611–9620, 2022.

[35] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision(ECCV)*, 2016.

[36] Gunnar A Sigurdsson, Santosh Divvala, Ali Farhadi, and Abhinav Gupta. Asynchronous temporal fields for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 585–594, 2017.

[37] Praveen Tirupattur, Kevin Duarte, Yogesh Rawat, and Mubarak Shah. Modeling multi-label action dependencies for temporal action localization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[38] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pages 4489–4497, Washington, DC, USA, 2015. IEEE Computer Society. ISBN 978-1-4673-8391-2. doi: 10.1109/ICCV.2015.510. URL http://dx.doi.org/10.1109/ICCV.2015.510.

[39] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021.

[40] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *Proceedings of the European conference on computer vision (ECCV)*, pages 399–417, 2018.

[41] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. Vlm: Task-agnostic video-language model pre-training for video understanding. *arXiv preprint arXiv:2105.09996*, 2021.

[42] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE international conference on computer vision*, pages 5783–5792, 2017.

[43] Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. Every moment counts: Dense detailed labeling of actions in complex videos. *International Journal of Computer Vision*, 126(2-4):375–389, 2018.