AGENT-ENVIRONMENT NETWORK FOR TEMPORAL ACTION PROPOSAL GENERATION

Viet-Khoa Vo-Ho[⋆†] *Ngan Le*[†] *Kashu Yamazaki*[†] *Akihiro Sugimoto*[§] *Minh-Triet Tran*[⋆]

[⋆] Faculty of Information Technology, University of Science, VNU-HCM, Vietnam
Vietnam National University, Ho Chi Minh City, Vietnam
[†] Department of Computer Science, University of Arkansas, Fayetteville, USA
[§]National Institute of Informatics, Japan

ABSTRACT

Temporal action proposal generation is an essential and challenging task that aims at localizing temporal intervals containing human actions in untrimmed videos. Most of existing approaches are unable to follow the human cognitive process of understanding the video context due to lack of attention mechanism to express the concept of an action or an agent who performs the action or the interaction between the agent and the environment. Based on the action definition that a human, known as an agent, interacts with the environment and performs an action that affects the environment, we propose a **contextual Agent-Environment Network**. Our proposed contextual AEN involves (i) **agent pathway**, operating at a local level to tell about which humans/agents are acting and (ii) **environment pathway** operating at a global level to tell about how the agents interact with the environment. Comprehensive evaluations on 20-action THUMOS-14 and 200-action ActivityNet-1.3 datasets with different backbone networks, i.e C3D and SlowFast, show that our method robustly exhibits outperformance against state-of-the-art methods regardless of the employed backbone network.

Index Terms— Action Proposal Generation, Contextual Agent-Environment Network

1 Introduction

Temporal action proposal generation (TAPG) aims at proposing video temporal intervals that likely contain an action in an untrimmed video with both action categories and temporal boundaries. This task has promising applications, such as action recognition [1], summarization [2, 3], captioning [4, 5], and video recommendation [6]. A robust TAPG method should be able to (i) generate temporal proposals with boundaries covering action instances precisely and exhaustively, (ii) cover multi-duration actions, and (iii) generate reliable confidence scores to retrieve proposals properly. Despite many recent endeavors, TAPG remains an open problem, especially when facing real-world complications such as action duration variability, activity complexity, camera motion, and viewpoint changes.

The limitations of the existing TAPG can be summarized as follows:

- Most of existing work [7], [8, 9], [1] extracts video visual representation by applying a backbone model into whole spatial dimensions of video frames. This tends predictions over-biased towards the environment rather than agents committing actions because the agents together with their actions usually occupy a small region compared to the entire frame.
- Existing approaches treat everything in a video frame in the same manner and does not pay attention to the difference among three key entities, i.e., agent, action, and environment, for temporal action proposal. Attention mechanism that enables us to capture such different key entities as well as to express the relationship between them is missing.
- Most of the existing approaches are unable to follow the human cognitive process of understanding the video content. In the human cognitive process, a person focuses on deciding what an agent is doing through the observation of agent activities and the environment around the agent. Nevertheless, such a process is not taken into account at all. Instead, existing work just applies a backbone network into entire spatial dimensions of video snippets of frames (8-frame snippets or 16-frame snippets, etc.).

To address the above drawbacks, we propose a novel **contextual AEN** to semantically extract video representation. Our proposed AEN contains two semantic pathways corresponding to (i) **agent pathway** and (ii) **environment pathway**. The contribution of **contextual AEN** is two-fold:

- AEN includes (i) Agent-Environment representation network (AERN) to extract rich features sequence from an untrimmed video and (ii) boundary matching networks to evaluate confidence scores of densely distributed proposals generated from the extracted feature.
- A novel video contextual Agent-Environment (AE) visual representation is introduced. Our semantic AE visual representation involves two parallel pathways to represent every snippet of the video: (i) agent pathway, operating at a local level to tell what the agents in the snippet are doing and which agents deserve to be concentrated more on; (ii) environment pathway, operating at a global level to express the relationship

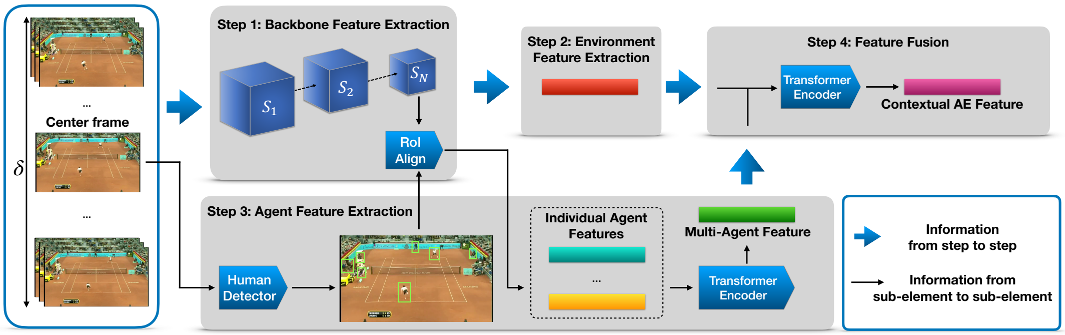
Center frame
δ
...
...
Step 1: Backbone Feature Extraction
S_1
S_2
S_N
RoI Align
Step 2: Environment Feature Extraction
Step 4: Feature Fusion
Transformer Encoder
Contextual AE Feature
Step 3: Agent Feature Extraction
Human Detector
Individual Agent Features
...
Multi-Agent Feature
Transformer Encoder
Information from step to step
Information from sub-element to sub-element

Fig. 1. The architecture of our proposed contextual Agent-Environment (AE) representation network (AERN).

between the agents and the environment. These two pathways are fused together by our attention mechanism for the video representation where a feature may focus more on either local or global levels entirely depending on the context of its corresponding snippet.

2 Related Work

TAPG [8, 1, 10, 11, 12, 13, 14, 15, 16, 17] aims at proposing intervals so that each of them contains an action instance with its associated temporal boundaries and confidence score in untrimmed videos. There are two main approaches in TAPG: anchor-based and boundary-based. The anchor-based methods [10, 11, 12, 13, 14] are inspired by anchor-based object detectors in still images like Faster R-CNN [18], RetinaNet [19], or YOLO [20]. These methods deal with the proposal task as a classification task where multiple predefined anchors with different lengths are regarded as classes and a class that best fits the ground truth action length is used as ground truth true class for training. Although this approach helps to save computational costs, it lacks the flexibility of action duration. The boundary-based methods [15, 16, 17], on the other hand, break every action intervals into starting and ending points and learn to predict them. In the inference phase, starting and ending probabilities at every timestamp in the given video are predicted. Then, points with local peak in probability are chosen as potential boundaries. The potential starting points are paired with potential ending points for a potential action interval when their interval fits in the predefined upper and lower threshold, along with a confidence score being a multiplication of the starting and ending probabilities. As one of the first boundary-based methods, [15] defined actionness scores by grouping continuous high-score regions as a proposal. Later, [16] proposed a two-stage strategy where boundaries and actionness scores at every temporal point are predicted in the first stage and fused together, filtered by Soft-NMS to get the final proposals at the second stage. [17] improved [16] by generating a boundary-matching matrix instead of actionness scores to capture an action-duration score for more descriptive final scores.

3 Proposed Method

Given an untrimmed video $\mathcal{V} = \{x_l\}_{l=1}^{L}$ with L frames, our goal is to generate a set of temporal segments, each of which possibly and tightly contain an action. Let us denote F as the visual representation of video $\mathcal{V}$, which is firstly divided into $T = \left\lfloor \frac{L}{\delta} \right\rfloor$ non-overlapping δ-frame snippets. Let ϕ be a feature extraction function which is applied to each δ-frame snippet, the visual representation F is then defined as follows:

$$F = \{f_i\}_{i=1}^{T} = \{\phi(x_{\delta\cdot(i-1)+1}, ..., x_{\delta\cdot i})\}_{i=1}^{T} \qquad (1)$$

In the next two subsections, we discuss how we devise Agent-Environment Representation Network (AERN) as a function ϕ and how we integrate it with an action proposal generation module, respectively.

3.1 AE Representation Network(AERN)

Our proposed AERN extracts contextual AE visual representation of a δ-frame snippet at both global and local levels, which plays a key role in temporal action proposals generation. Considering our goal is extracting features for a δ-frame snippet from frame t to frame $t + \delta$, the AERN is illustrated in Fig.1(a) and consists of following steps:

Step 1: Backbone Feature Extraction: In action recognition, a 3D convolutional backbone network is usually used to encode global semantic information of a δ-frame snippet. In this work, we employed C3D [7] and SlowFast [1] pre-trained on Kinetics-400 [21] as our backbone feature extractor. In order to capture enough semantic information of the snippet while keeping enough resolution in spatial domains, we discard the last fully connected layers to use the feature map S_N from the last convolutional block, which is crucial in Step 3.

Step 2: Environment Feature Extraction: To extract the environment feature, feature map S_N is passed through average pooling and several fully connected layers until the softmax layer, outputting a vector containing semantic information of the overall scene, namely, environment feature ϕ_e. This pathway captures the information at the global level of the scene, however, it may not capture small details like the motions of humans.

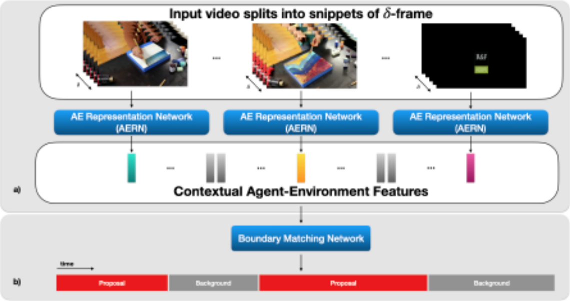
Input video splits into snippets of δ-frame
AE Representation Network (AERN)
AE Representation Network (AERN)
AE Representation Network (AERN)
a)
Contextual Agent-Environment Features
Boundary Matching Network
time
b)
Proposal
Background
Proposal
Background

Fig. 2. An overview architecture of our proposed AEN for action proposal generation where AE Representation Network is in Fig.1 and described in section 3.1

Step 3: Multi-Agent Feature Extraction: An agent in a δ-frame snippet is denoted as a human appears in it. To detect all agents existing in a δ-frame snippet, we start with the center frame by applying a human detector. These bounding boxes around agents are then used to guide the RoIAlign [22] to extract local features from S_N. Each local feature corresponds to an agent and all local features from multiple agents are fused into a single multi-agent feature ϕ_a via an attention model based on Transformer Encoder [23]. Thus, we obtain an agent ϕ_a from multiple agents of δ-frame snippet.

In this step, we adopt a Faster R-CNN [18] pre-trained on MS-COCO [24] as our human detector because of its good performance and popularity.

Step 4: Feature Fusion: In our proposed AEN, environment feature ϕ_e plays at the global level while multi-agent feature ϕ_a plays at the local level. After simultaneously extracting these features, Transformer Encoder [23] is employed to re-weight them by a proper ratio, which helps the overall model to know which information to consider while reasoning the action proposals, i.e. deciding whether to emphasize on detailed information of agents or overall information of the scene.

3.2 Deployment with Action Proposal Network

Our proposed AEN is easily deployed and incorporated with any TAPG network in an end-to-end framework as shown in Fig. 2. In this paper, Boundary-Matching Network (BMN) is employed because of its impressive performance. BMN is a fully convolutional network with 3 modules, namely, Base Module (BM), Temporal Evaluation Module (TEM), and Proposal Evaluation Module (PEM). BM processes the input features through several 1D convolutional layers, producing output features that are fed into TEM and PEM simultaneously. TEM aims to produce the probabilities for every temporal point in the features set being a starting or ending boundaries. Meanwhile, PEM produces two matrices, each of which densely contains the confidence scores of every possible duration at every starting temporal point, but are trained by two different types of loss functions as suggested by [25].

3.3 Training Phase

Label Generation: We follow [25, 16] to generate the ground truth labels for the training process including starting labels, ending labels for TEM training and duration labels for PEM training. The starting and ending labels are generated for every snippet of the video, which are called $L_S = \{l^s_n\}^T_{n=1}$ and $L_E = \{l^e_n\}^T_{n=1}$, respectively. A label point l^s_n (or l^e_n) is set to 1 if its corresponding timestamp in the video is the nearest to any ground truth starting (ending) timestamp.

The duration labels for a video are gathered into a matrix $L_D \in [0,1]^{D\times T}$ where D is the maximum length of proposals being considered in number of snippets, as suggested in [25], we set $D = T$ and $D = T/2$ for experiments on ActivityNet-1.3 [31] and THUMOS-14 [32], respectively. 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 is assigned by 1 if its Interaction-over-Union with any ground truth action in $\mathcal{A} = \{a_i\}^M_{i=1}$ reach a local maximum, or 0 otherwise.

Loss function: As mentioned in section 3.2, TEM generates probabilities vectors of starting and ending boundaries (P_S and P_E), while PEM generates two actionness scores matrices P^{cc}_D and P^{cr}_D. These four outputs are trained simultaneously by different loss functions as following:

$$\mathcal{L}_{TEM} = \mathcal{L}_{binary}(P_S, L_S) + \mathcal{L}_{binary}(P_E, L_E) \quad (2)$$

$$\mathcal{L}_{PEM} = \mathcal{L}_{binary}(P^{cc}_D, L_D) + \lambda_{reg} \cdot \mathcal{L}_2(P^{cr}_D, L_D) \quad (3)$$

$$\mathcal{L} = \lambda_1 \cdot \mathcal{L}_{TEM} + \lambda_2 \cdot \mathcal{L}_{PEM} \quad (4)$$

We follow [25, 16] and set $\lambda_{reg} = 10$ and $\lambda_1 = \lambda_2 = 1$. Furthermore, $\mathcal{L}_{binary}$ is a weighted binary log-likelihood function to deal with imbalanced number of negative and positive examples in groundtruth labels:

$$\mathcal{L}_{binary} = \frac{1}{N}\sum_{i=1}^{N} \alpha^+ \cdot l_i \cdot \log p_i + \alpha^- \cdot (1 - l_i) \cdot \log p_i, \quad (5)$$

where l_i and p_i are label and probability of the output, respectively. $\alpha^+ = \frac{N}{N^+}$ and $\alpha^- = \frac{N}{N^-}$, with N, N^- and N^+ are total number of examples and total number of positive and negative examples, respectively.

3.4 Inference Phase:

During inference, four outputs are generated by BMN model [25] from features set extracted by our AEN, including P_S, P_E, P^{cc}_D, and P^{cr}_D. Peaking probabilities of starting and ending boundaries from P_S and P_E, which are local maximums, are selected to form initial proposals by pairing every peak starting point with peak ending points behind them and within a pre-defined range. For a proposal formed by t_s and t_e boundaries with duration $d_p = t_e - t_s$, its score $score_p$ are computed by the following formula as proposed in [25]:

$$score_p = P_S[t_s] \cdot P_E[t_e] \cdot \sqrt{P^{cc}_D[d_p, t_s] \cdot P^{cr}_D[d_p, t_s]} \quad (6)$$

Then, with a list of proposals and their scores, we apply a Soft-NMS [33] to eliminate highly overlapped proposals before outputting the final list of proposals.

Table 1. Comparison in terms of AR@AN and AUC on validation set and test set of ActivityNet-1.3 dataset

	TCN [26]	MSRA [27]	Prop-SSAD [28]	CTAP [29]	BSN [16]	SRG [30]	MGG [17]	BMN [25]	Our AEN	
									SlowFast	C3D
AR@100 (val)	-	-	73.01	73.17	74.16	74.65	74.54	75.01	75.62	**75.65**
AUC (val)	59.58	63.12	64.40	65.72	66.17	66.06	66.43	67.10	67.78	**68.15**
AUC (test)	61.56	64.18	64.80	-	66.26	-	66.47	67.19	68.45	**68.99**

4 Experiments

4.1 Datasets

ActivityNet-1.3 [31] is a large scale dataset for human activity understanding, containing roughly 20K untrimmed videos which are divided into training, validation and test sets with the ratio of 0.5, 0.25 and 0.25, respectively.

THUMOS-14 [32] is primarily a dataset for action recognition, yet, it also opens the action localization track, which is held on a portion of its validation set for training and another portion of test set for testing, with each comprised of 200 and 214 videos, respectively; and captures 20 different actions.

For comparability purposes, we follow the same settings as it was in [25] for both datasets.

4.2 Implementation Details

For ActivityNet-1.3, we benchmark our proposed AEN with both C3D [7] and SlowFast [1] backbone, whereas, for THUMOS-14, we only benchmark our method on C3D backbone. All backbones are pre-trained on Kinetics-400 [21]. Following [25, 16], we trained our proposed network with Adam update rule is employed with the initial learning rate of 0.0001 and 0.001 for ActivityNet-1.3 and THUMOS-14, respectively.

4.3 Experimental Results

Table 1 shows the comparison in terms of AR@AN (AN = 100) and AUC between our AEN against other SOTA methods on both validation and test sets of ActivityNet-1.3 dataset. Compared to SOTA approaches, our AEN obtains better performance with large margins on both AR@AN and AUC metrics regardless of the backbone networks. Likewise, our AEN also gives a superior performance on THUMOS-14 in Table 3 when compared to SOTA approaches on this dataset.

Generalization is also an important aspect to be evaluated in TAPG. We conduct experiments on ActivityNet-1.3 [31] to evaluate this property, in which videos in two non-overlapped action class subsets of ”Sports, Excercise, and Recreation” and ”Socializing, Relaxing, and Leisure” are collected into *Seen* and *Unseen* subsets, respectively. Table 2 delivers two training settings, results evaluated on Unseen subset does not drop significantly when training only on *Seen* subset comparing to training on *Seen+Unseen* sets, which implies that our AEN achieves high generalizability in generating proposals.

Table 2. Generalization evaluation on ActivityNet 1.3.

	Seen		Unseen	
Training Data	AR@100	AUC	AR@100	AUC
Seen+Unseen	74.58	66.96	75.25	67.49
Seen	74.40	66.69	**73.66**	**65.92**

Table 3. Comparison on THUMOS-14 test set (AR@AN).

Methods	@50	@100	@200	@500	@1000
SCNN-prop [13]	17.22	26.17	37.01	51.57	58.20
SST [34]	19.90	28.36	37.90	51.58	60.27
TURN [14]	19.63	27.96	38.34	53.52	60.75
MGG [17]	29.11	36.31	44.32	54.95	60.98
BSN [16]	29.58	37.38	45.55	54.67	59.48
BMN [25]	32.73	40.68	47.86	56.42	60.44
Our AEN	**33.36**	**42.93**	**50.34**	**59.10**	**64.03**

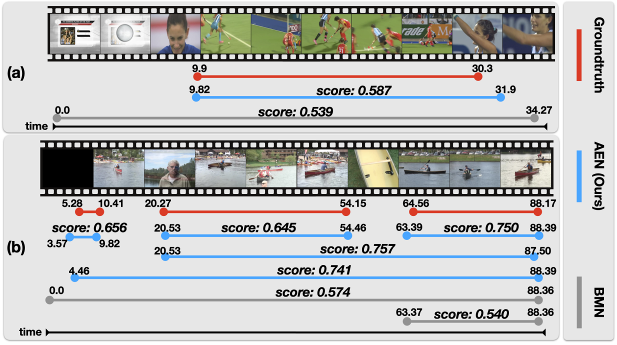

Fig. 3. Qualitative results by BMN [25] and our proposed AEN on ActivityNet-1.3 [31] on C3D backbone network.

Conclusion

This paper proposed a novel AEN for the TAPG problem. Different from existing work applying a backbone network into an entire video frame, AEN involves two parallel pathways in the video visual representation: (i) the agent pathway, which plays at the local level and tells about where agents are and what the agents are doing; (ii) the environment pathway, which plays at the global level and tells about how the environment affects after receiving the actions from the agents as well the relationship between the agents, actions, and the environment. Our experiments demonstrated that AEN outperforms the SOTA methods with C3D backbone on THUMOS-14 and with both C3D and SlowFast backbones on ActivityNet-1.3.

Acknowledgment: This material is based upon work supported by the National Science Foundation under Award No. OIA-1946391. Research is partially supported by Vingroup Innovation Foundation (VINIF) in project code VINIF.2019. DA19, and the Domestic Master/PhD Scholarship of VINIF.

Disclaimer: Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

5 References

[1] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He, "Slowfast networks for video recognition," in *ICCV*. 2019, pp. 6201–6210, IEEE.

[2] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, "Describing videos by exploiting temporal structure," in *ICCV*, 2015, pp. 4507–4515.

[3] T. Yao, T. Mei, and Y. Rui, "Highlight detection with pairwise deep ranking for first-person video summarization," in *The 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 982–990.

[4] Haifeng Hu Weixuan Wang, Zhihong Chen, "Hierarchical attention network for image captioning," in *AAAI*, 2019, pp. 8957–8964.

[5] S. Liu, Z. Ren, and J. Yuan, "Sibnet: Sibling convolutional encoder for video captioning," *ITPAMI*, pp. 1–1, 2020.

[6] Joonseok Lee and Sami Abu-El-Haija, "Large-scale content-only video recommendation," in *ICCV-W*, Oct 2017.

[7] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu, "3d convolutional neural networks for human action recognition," *TPAMI*, vol. 35, no. 1, pp. 221–231, Jan 2013.

[8] Karen Simonyan and Andrew Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NIPS*, 2014, pp. 568–576.

[9] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman, "Convolutional two-stream network fusion for video action recognition," in *CVPR*, 6 2016, pp. 1933–1941.

[10] A. Richard and J. Gall, "Temporal action detection using a statistical language model," in *CVPR*, 2016, p. 3131–3140.

[11] Y. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar, "Rethinking the faster r-cnn architecture for temporal action localization," in *CVPR*, 2018, pp. 1130–1139.

[12] Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem, "Fast temporal activity proposals for efficient detection of human actions in untrimmed videos," in *CVPR*, June 2016.

[13] Zheng Shou, Dongang Wang, and Shih-Fu Chang, "Temporal action localization in untrimmed videos via multi-stage cnns," in *CVPR*, 2016, pp. 1049–1058.

[14] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia, "Turn tap: Temporal unit regression network for temporal action proposals," in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[15] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin, "Temporal action detection with structured segment networks," in *ICCV*, 2017, pp. 2933–2942.

[16] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang, "Bsn: Boundary sensitive network for temporal action proposal generation," *ECCV*, pp. 3–21, 2018.

[17] Yuan Liu, Lin Ma, Yifeng Zhang, Wei Liu, and Shih-Fu Chang, "Multi-granularity generator for temporal action proposal," in *CVPR*, 2019, pp. 3604–3613.

[18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NIPS*, Cambridge, MA, USA, 2015, NIPS'15, p. 91–99, MIT Press.

[19] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017, pp. 2999–3007.

[20] Joseph Redmon and Ali Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.

[21] 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.

[22] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick, "Mask r-cnn," in *ICCV*, Oct 2017.

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Larry Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[25] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen, "Bmn: Boundary-matching network for temporal action proposal generation," in *ICCV*, October 2019.

[26] X. Dai, B. Singh, G. Zhang, L. S. Davis, and Y. Q. Chen, "Temporal context network for activity localization in videos," in *ICCV*, 2017, p. 5727–5736.

[27] 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.

[28] T. Lin, X. Zhao, and Z. Shou, "Temporal convolution based action proposal: Submission to activitynet 2017," in *arXiv preprint arXiv:1707.06750*, 2017.

[29] Jiyang Gao, Kan Chen, and Ram Nevatia, "Ctap: Complementary temporal action proposal generation," in *ECCV*, 2018.

[30] 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.

[31] Bernard Ghanem Fabian Caba Heilbron, Victor Escorcia and Juan Carlos Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *CVPR*, 2015, pp. 961–970.

[32] 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," `http://crcv.ucf.edu/THUMOS14/`, 2014.

[33] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis, "Soft-nms – improving object detection with one line of code," in *ICCV*, Oct 2017.

[34] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles, "Sst: Single-stream temporal action proposals," in *CVPR*, July 2017.