UCF-System for TRECVID-2021 ActEV challenge

Zacchaeus Scheffer*, **Ishan Dave***, **Akash Kumar***, **Praveen Tirupattur***,
Yogesh Rawat†, Mubarak Shah†
Center for Research in Computer Vision
University of Central Florida, Orlando, Florida, USA
*{zaccy, ishandave, akash_k , praveentirupattur}@knights.ucf.edu
†{yogesh, shah}@crcv.ucf.edu

Abstract

Activity detection has wide-reaching applications in video surveillance, sports, and behavior analysis. The existing literature in activity detection has mainly focused on benchmarks like AVA, AVA-Kinetics, UCF101-24, and JHMDB-21. However, these datasets fail to address all issues of real-world surveillance camera videos like untrimmed nature, tiny actor bounding boxes, multi-label nature of the actions, etc. In this work, we propose a real-time, online, action detection system which can generalize robustly on any unknown facility surveillance videos. Our real-time system mainly consists of tracklet generation, tracklet activity classification, and prediction refinement using the proposed post-processing algorithm. We tackle the challenging nature of action classification problem in various aspects like handling the class-imbalance training using PLM method and learning multi-label action correlations using LSEP loss. In order to improve the computational efficiency of the system, we utilize knowledge distillation. Our approach gets second place in TRECVID 2021 ActEV challenge. Project Webpage: www.crcv.ucf.edu/research/projects/gabriellav2/

1 Introduction

The problem of video understanding has wide-reaching applications like action recognition [1–8], action detection [9–13], temporal action localization [14, 15], and video synthesis [16, 17].

The task of spatio-temporal activity localization involves detecting the actions present in the videos, and generating a spatial bounding box that tracks the activities over time. The main two problem statements involving videos are: *Can we recognize the action in the video?* and *If so, can we say where the activity is happening?* The first problem is termed as video classification, which involves labeling single or multiple simultaneous activities present in a video. The second problem targets annotating *where* the activity is happening. This is referred as the task of spatio-temporal activity localization.

The majority of works [18–22] on action detection focus on benchmark datasets like AVA [23], AVA-Kinetics [24], UCF101-24 [25] or J-HMDB [26]. These approaches are not suitable for real-world surveillance video due to several reasons: (1) actor size of the surveillance camera is tiny compared to the actor-centric videos of the benchmarks, (2) surveillance videos are untrimmed, unlike the 3 second trimmed videos of AVA [23] and AVA-Kinetics [24], and (3) real-time and online approach is required for the video surveillance.

Prior works [10, 13, 27–35] present approaches for action detection in surveillance video. One of the best performing systems from the prior works is our prior system, Gabriella [10], which is a real-time, online, action detection approach. Gabriella adopts an end-to-end approach by first detecting the action proposal using a pixel-wise localization module which is followed by action classification and post-processing. Although this system outperforms most of the concurrent systems, it has two main limitations: (1) it merges overlapping actor bounding boxes, which results in huge regions for indoor scene and degrades performance of action classification stage, and (2) localization network does not generalize well on the unknown scene/facility camera, which results in a high probability of missing actions.

In this work, we build upon our previous system, Gabriella [10], to improve the system overall performance and generalization capability in unknown facility cameras. Firstly, in order to avoid merging in crowded scenes we replace the pixel-wise localization network with the object detector and tracker to get actor-centric trackelets. Secondly, we strengthen the action classification unit by utilizing state-of-the-art multi-label class-imbalance training, partial label masking (PLM), and learning class-correlation through log-sum-exp pairwise (LSEP) loss. We also utilize knowledge distillation to make the action classification component more computationally efficient. Our system places second in VIRAT TRECVID ActEV 2021 challenge.

2 Related Works

Spatio-Temporal Activity Localization: The task of recognizing and localizing actions across frames in videos is termed as spatio-temporal activity localization. Primitive works took inspiration from images and 2D models and extended such approaches to frames. With the introduction of 3D convolutions, most of the works shifted from 2D-CNN backbones [36–38] to 3D-CNN [39–41]. The main limitation of the prior works is that they have been trained and tested mostly on trimmed datasets such as UCF101-24 [42], JHMDB-21 [26] or AVA [23]. In the real-world, we deal with untrimmed videos. In the literature, only a few large-scale datasets have been created to tackle this problem [43–45]. ActEV UF-Full and TrecVID utilize the MEVA dataset and VIRAT [46] datasets respectively to develop more works on untrimmed videos for the spatio-temporal localization task. What makes these datasets challenging, is the average length of videos, which is 20 to 30 times that of previously proposed datasets. The mains problem solved on untrimmed datasets is to approximate where the activity is happening in the temporal dimension and detect the type of action being localized. Also, the solutions are not always real-time, which is a critical aspect for security surveillance videos. In our work, we develop a real-time spatio-temporal localization framework to detect actions in these long untrimmed videos.

Post-processing: In general, raw output of object detection algorithm can't be used as a finalized localization map. It contains a lot of false positives indicating multiple instances of a single object. These multiple instances needs to be suppressed to generate a single instance per object detected. There have been works [47–49] to tackle this issue utilizing Non-Maximum threshold in parallel to object detection approaches. T-CNN [50] imposes high confidence score based on contextual information. [47], [48] and [49] uses temporal overlap scores of bounding box across frames. This approaches are mostly limited to ImageNetVID [51] dataset. Since, most of the datasets are trimmed, the problem of false alarms have mostly been looked over spatially across frames. On the other hand, in an untrimmed video, multiple actions have an abrupt starting and ending time. Thus, we extend these approaches to spatio-temporal dimension. We target multiple detection on a frame (spatially), and, extend those detections across multiple frames (temporal) suppressing the false alarm detections. However, we use tracking ids of proposals instead of object detections per frame. We also monitor the classification score of detections over time. This procedure not only helps us to link detections efficiently, it also suppresses the contrastive fine-grained activities such as *person standing up* versus *person sitting down*.

3 Method

3.1 Overview

The proposed system takes in a video clip as input and detects all activities in the form of tracklets. The system first operates on entire clip to spatio-temporally localize actor tracklets. Once we extract potential tracklets, our classification system identifies all possible activities occurring within each tracklet. These action predictions are then fed into our TMAS system, which simultaneously filters and combines them into accurate and consistent action tubes. As an end result, we obtain spatio-temporal action detections over long untrimmed videos in an online real-time process. The following sections describe the different components of our system.

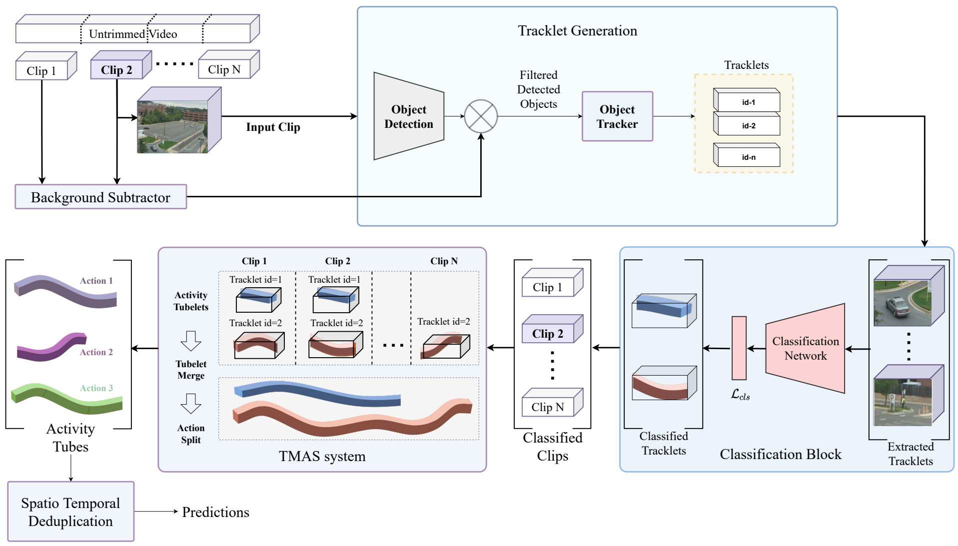

Figure 1: Schematic Diagram for UCF DIVA system: Firstly, an untrimmed video is divided into fixed temporal sized clips, which are then passed to the object detector to detect the actors frame-wise. The actor bounding boxes in different frames of the clip are then joined using a tracker to get tracklets. The action classifier predicts actions classes on each tracklet, which are then post-processed through the proposed post-processing algorithm.

3.2 Tracklet Generation

Tracklet Identification To identify tracklets in a clip, we first send every fourth frame in that clip to an object detector. The object detector gives a pixelwise probability mask which is thresholded into a binary mask, with positive regions connected into objects using connected component analysis, and the resulting components coverted into bounding boxes. These candidate object bounding boxes are sent through a background subtractor which filters out objects which are sufficiently stationary. This is fine because we only care about objects which are performing an action. Finally, these filtered bounding boxes are sent to an object tracker which assigns an object id to each detection such that the same object gets the same id in subsequent frames. Then, for each object id, all corresponding bounding boxes are merged into the smallest-bounding bounding box. The cuboid defined by this merged bounding box that spans the entire clip temporally, along with the associated object id is a tracklet.

Tracklet Extraction To extract a tracklet, we crop the clip according to the tracklet's cuboid, and linearly interpolate that crop into a consistent resolution for our classifier. This cropped and resized clip with the associated object id is an extracted tracklet.

3.3 Tracklet Classification

The next step in our proposed system is tracklet classification. Our action classification network is a multi-label prediction network, which classifies the actions present within each tracklet. We treat this as a multi-label classification problem because actors can perform multiple activities simultaneously. For example, an actor can perform the actions *Riding* and *activity_carrying* at the same time. We use a 3D-Convolution based deep learning model [1] initialized with pre-trained weights on Kinetics [52] dataset for action classification. We modify the final layer of the model to have a $C + 1$ dimensional output, where C is the number of action classes and the additional output is for the background class. A sigmoid activation is used in the final layer in place of a softmax as this is a multi-label classifier. We use BCE loss to train the classifier which is defined as,

$$\mathcal{L}_{cls}(\hat{y}, y) = -\frac{1}{C+1}\sum_{i=0}^{C}[y_i log(\hat{y}_i) + (1-y_i)log(1-\hat{y}_i)] \tag{1}$$

where $\hat{y}_i$ is the prediction and y_i is the ground truth label.

3.4 TMAS Algorithm

To merge the tracklets and obtain the final action tubes, we propose the tracklet-Merge Action-Split algorithm (TMAS). Each tracklet t_i is described as follows: $\left(f_1^i, f_2^i, \mathbf{b}^i, \mathbf{a}_c^i\right)$ where f_1^i is the start time, f_2^i is the end time, $\mathbf{b}^i$ are the bounding boxes for each frame of the tracklet, and $\mathbf{a}_c^i$ are the frame-level action probability scores for each action class $c \in \{0, 1, ...C\}$, where 0 is background. First, we merge the tracklets into action-agnostic tubes of varying length; then we split these action-agnostic tubes into a set of action-specific tubes which contain the spatio-temporal localizations for the various activities in the video.

Algorithm 1 The Tubelet-Merge algorithm which merges tracklets into action-agnostic tubes. The CheckEnd function determines if a candidate tube becomes a final tube or is merged with another candidate.

Input: A stream of tracklets, **S**, from the classifier
Output: A set of action-agnostic spatio-temporal tubes, T_{done}
Notation: $\text{Inter}_{\text{temp}}$ calculates temporal overlap between tracklets.
$|\mathbf{M}[(t_c, *)]|$ returns the cardinality of the set $\{t : \mathbf{M}[(t_c, t)] > 0\}$.

procedure Tubelet-Merge(**S**)
  $T_{prev}, T_{done} \leftarrow \{\}$ ▷ Initialize candidate and final tubes
  M ← initialize hash table
  while t_c in **S** **do** ▷ Continue until the stream of tracklets ends
    for all t_p in T_{prev} **do**
      if $\text{Inter}_{\text{temp}}(t_p, t_c) > 0$ **then**
        $\mathbf{M}[(t_p, t_c)] \leftarrow IoU(t_p, t_c)$
      else
        CheckEnd(t_p, T_{prev}, **M**)
    append t_c to T_{prev} ▷ Tubelet becomes a candidate tube
  while T_{prev} is not empty **do** ▷ Deals with remaining candidates
    $t_p \leftarrow T_{prev}[0]$
    CheckEnd(t_p, T_{prev}, **M**)
  return T_{done}

function CheckEnd(t_p, T_{prev}, **M**)
  if $|\mathbf{M}[(t_p, *)]| == 0$ **then**
    Move(t_p, T_{prev}, T_{done}) ▷ Moves t_p from T_{prev} to T_{done}
  else if $|\mathbf{M}[(t_p, *)]| == 1$ **then**
    $t_i \leftarrow \max_{t_i} \mathbf{M}[(t_p, t_i)]$
    if $|\mathbf{M}[(*, t_i)]| == 1$ **then**
      Merge(t_p, t_i, T_{prev}, **M**)
    else
      Move(t_p, T_{prev}, T_{done})
  else
    $t_i \leftarrow \max_{t_i} \mathbf{M}[(t_p, t_i)]$
    Merge(t_p, t_i, T_{prev}, **M**)

function Merge(t_1, t_2, T_{prev}, **M**) ▷ Merges two candidate tubes
  $t_1 \leftarrow (f_1^1, f_2^2, \{\mathbf{b}^1, \mathbf{b}^2\}, \{\mathbf{a}^1, \mathbf{a}^2\})$ ▷ $\{\}$ is concatenation
  remove t_2 from T_{prev}
  $\mathbf{M}[t_1, t_i] \leftarrow \mathbf{M}[t_2, t_i]$ ▷ Done for all t_i where $\mathbf{M}[t_2, t_i] \geq 0$

Tracklet-Merge The procedure to merge tracklets into action-agnostic tubes is described in Algorithm 1. The temporally sequential stream of tracklets coming from the classification network are passed to the Tubelet-Merge procedure as input. The set of candidate tubes is initialized with the first tracklet. For each subsequent tracklet, we look for spatio-temporal overlap with the existing candidate tubes. This results in four possible outcomes: 1) If there is no overlap, the tracklet itself becomes a new candidate tube, 2) If there is a unique match found between a candidate tube and the tracklet, they are merged and become a new candidate tube, 3) if the tublet has an overlap with multiple candidates, then the tracklet becomes a new candidate, 4) if multiple tublets have an overlap with a single candidate tube, then the tracklet with the highest overlap is merged with that candidate

Algorithm 2 The Action-Split algorithm which converts the action-agnostic tubes into action-specific predictions.

Input: A set of action-agnostic tubes, T, and a set of actions, C
Output: A set of spatio-temporal action-specific tubes, A_G
Notation: The hyperparameters $\tau, \alpha, \beta,$ and γ are described in the supplementary materials. $a_c^i[f]$ and $t_i[f]$ contain the action prediction scores and tube information at frame f, respectively.

procedure ACTION-SPLIT(T)
    $A_G \leftarrow \{\}$ ▷ Initializes the action-specific tubes
    for all t_i in T **do**
        $t_{smooth} \leftarrow$ SMOOTH(t_i)
        for all c in $1 : C$ **do** ▷ Loop through each action class
            $a_L \leftarrow$ EXTRACT(t_{smooth}, c)
            append a_L to A_G
    return A_G

function SMOOTH(t_i)
    for all f in $f_1^i : f_2^i$ **do**
        $a_c^i\,[f] \leftarrow \frac{1}{2\tau+1} \sum_{k=-\tau}^{\tau} a_c^i\,[f+k]$
    return t_i

function EXTRACT(t_i, c) ▷ Extracts tubes of a specific class
    $A_L, a_l \leftarrow \{\}$ ▷ Initialize extracted action tubes and a placeholder
    $count \leftarrow 0$
    for all f in $f_1^i : f_2^i$ **do**
        if $a_c^i[f] > \alpha$ **then** ▷ Continue current action tube
            append $t_i[f]$ to a_l
            $count \leftarrow 0$
        else
            $count \leftarrow count + 1$
        if $count > \beta$ **then** ▷ Current action tube is finished
            append a_l to A_L
            $a_l \leftarrow \{\}, count \leftarrow 0$
    remove tubes shorter than γ from A_L
    return A_L

and the other tracklets become separate candidate tubes. Once all tracklets are checked, the candidate tubes become the final action-agnostic tubes.

Action-Split From the action-agnostic tubes we obtain action-specific spatio-temporal localizations using the Action-Split procedure described in Algorithm 2. We start by smoothing out per-frame action confidence scores; which accounts for fragmentation caused by action miss-classifications. Then we build the action-specific tubes by checking for continuous occurrences of each action class; this allows several occurrences of the same activity to occur within a single tube. For instance, a person *walking* might stop and *stand* for several seconds and start walking again; this entire sequence will be contained in a single spatio-temporal tube, but the Action-Split procedure will correctly generate two separate instances of *activity_walking* and one instance of *activity_standing*. To be robust to classification errors, action tubes with the same action label that are within a limited temporal neighborhood are combined together to form a single continuous action prediction.

Runtime Complexity The worst-case runtime of our TMAS algorithm is $\mathcal{O}\left(n^2\right)$, where n is the total number of candidate tubes at any given time. However, we sequentially process our tracklets and constantly shift the candidate tubes which can not have any possible future match to the set of final tubes. Therefore, the set of candidate tubes at any particular time is reasonably small and our TMAS algorithm contributes negligible overhead to our system's overall computation time.

4 Experiments

Classification Network: We experiment with multiple classification models to determine the best network architecture for our system. For a fair comparison, all models are initialized with pre-trained weights on the Kinetics [52] and are trained with the same settings. A comparison of their performance on the VIRAT validation set is shown in Table 1. We use the average F1-Score as a metric for comparison and observe that R(2+1)D model [53] outperforms the other models.

Architecture	Precision	Recall	F1-Score
I3D [54]	0.36	0.31	0.33
P3D [55]	0.43	0.41	0.41
3D-ResNet [1]	0.46	**0.43**	0.44
R(2+1)D [53]	**0.50**	**0.43**	**0.45**

Table 1: Ablation experiments for different classification network architectures. Precision, Recall, and F1-scores are averaged over all classes on the VIRAT validation set.

Rank	team_name	team_abbrev	nAUDC@tfa0.2	p_miss@tfa0.15
1	BUPT-MCPRL	BUPT-MC_26542	0.4085	0.3249
2	UCF	UCF_26546	0.4306	0.3408
3	INF	INF_26532	0.4444	0.3508
4	M4D_2021	M4D_202_26467	0.8466	0.7941
5	TokyoTech_AIST	TOKYOTE_26508	0.8516	0.8197
6	Team UEC	TEAMUE_26530	0.9640	0.9503

Table 2: Official results for TRECVID 2021 ActEV challenge. Best and second best scores are highlighted.

4.1 Comparison with other teams

As shown in Table 2, we placed second in the competition overall with an nAUDC of 0.4306 and a pmiss of 0.3408, lagging behind first by only 0.0221 and 0.0159 respectively.

5 Acknowledgement

Authors would like to acknowledge support from the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA R&D Contract No. D17PC00345. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Authors would also like to thank Jonathan Fiscus (NIST) for providing useful tools and data for system evaluation and comparison.

References

[1] K. Hara, H. Kataoka, and Y. Satoh. Towards good practice for action recognition with spatiotemporal 3d convolutions. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 2516–2521, 2018.

[2] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 203–213, 2020.

[3] Ugur Demir, Yogesh S Rawat, and Mubarak Shah. Tinyvirat: low-resolution video action recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 7387–7394. IEEE, 2021.

[4] Ishan Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. Tclr: Temporal contrastive learning for video representation. *arXiv preprint arXiv:2101.07974*, 2021.

[5] Ishan Dave, Naman Biyani, Brandon Clark, Rohit Gupta, Yogesh Rawat, and Mubarak Shah. " knights": First place submission for vipriors21 action recognition challenge at iccv 2021. *arXiv preprint arXiv:2110.07758*, 2021.

[6] Ishan Dave, Chen Chen, and Mubarak Shah. Spact: Self-supervised privacy preservation for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[7] Mohit Sharma, Raj Aaryaman Patra, Harshal Desai, Shruti Vyas, Yogesh Rawat, and Rajiv Ratn Shah. Noisyactions2m: A multimedia dataset for video understanding from noisy labels. In *ACM Multimedia Asia*, pages 1–5. 2021.

[8] Alec Kerrigan, Kevin Duarte, Yogesh Rawat, and Mubarak Shah. Reformulating zero-shot action recognition for multi-label actions. *Advances in Neural Information Processing Systems*, 34, 2021.

[9] Kevin Duarte, Yogesh Rawat, and Mubarak Shah. Videocapsulenet: A simplified network for action detection. In *Advances in Neural Information Processing Systems*, pages 7610–7619, 2018.

[10] Mamshad Nayeem Rizve, Ugur Demir, Praveen Tirupattur, Aayush Jung Rana, Kevin Duarte, Ishan R Dave, Yogesh S Rawat, and Mubarak Shah. Gabriella: An online system for real-time activity detection in untrimmed security videos. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 4237–4244. IEEE, 2021.

[11] Aayush J. Rana and Yogesh S. Rawat. We don't need thousand proposals: Single shot actor-action detection in videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2960–2969, January 2021.

[12] Ishan Dave, Zacchaeus Scheffer, Praveen Tirupattur, Yogesh Rawat, and Mubarak Shah. Ucf-system: Activity detection in untrimmed videos. 2020.

[13] Wenhe Liu, Guoliang Kang, Po-Yao Huang, Xiaojun Chang, Yijun Qian, Junwei Liang, Liangke Gui, Jing Wen, and Peng Chen. Argus: Efficient activity detection system for extended video analysis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, March 2020.

[14] Praveen Tirupattur, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. Modeling multi-label action dependencies for temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1460–1470, June 2021.

[15] Sirnam Swetha, Hilde Kuehne, Yogesh S Rawat, and Mubarak Shah. Unsupervised discriminative embedding for sub-action learning in complex activities. *arXiv preprint arXiv:2105.00067*, 2021.

[16] Naman Biyani, Aayush J Rana, Shruti Vyas, and Yogesh S Rawat. Larnet: Latent action representation for human action synthesis, 2021.

[17] Sarah Shiraz, Krishna Regmi, Shruti Vyas, Yogesh S. Rawat, and Mubarak Shah. Novel view video prediction using a dual representation, 2021.

[18] Junting Pan, Siyu Chen, Mike Zheng Shou, Yu Liu, Jing Shao, and Hongsheng Li. Actor-context-actor relation network for spatio-temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 464–474, 2021.

[19] Shoufa Chen, Peize Sun, Enze Xie, Chongjian Ge, Jiannan Wu, Lan Ma, Jiajun Shen, and Ping Luo. Watch only once: An end-to-end video action detection framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8178–8187, 2021.

[20] Yuanzhong Liu, Zhigang Tu, Liyu Lin, Xing Xie, and Qianqing Qin. Real-time spatio-temporal action localization via learning motion representation. In *ACCV Workshops*, pages 184–198, 2020.

[21] Bo Chen and Klara Nahrstedt. Escalation: a framework for efficient and scalable spatio-temporal action localization. In *Proceedings of the 12th ACM Multimedia Systems Conference*, pages 146–158, 2021.

[22] Okan Köpüklü, Xiangyu Wei, and Gerhard Rigoll. You only watch once: A unified cnn architecture for real-time spatiotemporal action localization. *arXiv preprint arXiv:1911.06644*, 2019.

[23] Chunhui Gu, Chen Sun, Sudheendra Vijayanarasimhan, Caroline Pantofaru, David A. Ross, George Toderici, Yeqing Li, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. AVA: A video dataset of spatio-temporally localized atomic visual actions. *CoRR*, abs/1705.08421, 2017.

[24] Ang Li, Meghana Thotakuri, David A Ross, João Carreira, Alexander Vostrikov, and Andrew Zisserman. The ava-kinetics localized human actions video dataset. *arXiv preprint arXiv:2005.00214*, 2020.

[25] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017.

[26] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *International Conf. on Computer Vision (ICCV)*, pages 3192–3199, December 2013.

[27] Ishan Dave, Zacchaeus Scheffer, Akash Kumar, Sarah Shiraz, Yogesh Singh Rawat, and Mubarak Shah. Gabriellav2: Towards better generalization in surveillance videos for action detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 122–132, January 2022.

[28] Ya Li, Guanyu Chen, Xiangqian Cheng, Chong Chen, Shaoqiang Xu, Xinyu Li, Xuanlu Xiang, Yanyun Zhao, Zhicheng Zhao, and Fei Su. Bupt-mcprl at trecvid 2019: Actev and ins. In *TRECVID*, 2019.

[29] Takashi Hosono, Kiyohito Sawada, Yongqing Sun, Kazuya Hayase, and Jun Shimamura. Activity normalization for activity detection in surveillance videos. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1386–1390. IEEE, 2020.

[30] Zhijian Hou, Yingwei Pan, Ting Yao, and Chong-Wah Ngo. Vireojd-mm@ trecvid 2019: Activities in extended video (actev). In *TRECVID*, 2019.

[31] Yongqing Sun, Xu Chen, Chaoyu Li, Kiyohito Sawada, Takashi Hosono, Jun Zhu, Chengjuan Xie, Sixiang Huang, Lan Wang, Kai Hu, et al. Ntt_cqupt@ trecvid2019 actev: Activities in extended video. In *TRECVID*, 2019.

[32] Junwei Liang, Lu Jiang, Juan Carlos Niebles, Alexander G Hauptmann, and Li Fei-Fei. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5725–5734, 2019.

[33] Wenhe Liu, Guoliang Kang, Po-Yao Huang, Xiaojun Chang, Yijun Qian, Junwei Liang, Liangke Gui, Jing Wen, and Peng Chen. Argus: Efficient activity detection system for extended video analysis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops*, pages 126–133, 2020.

[34] Konstantinos Gkountakos, Despoina Touska, Konstantinos Ioannidis, Theodora Tsikrika, Stefanos Vrochidis, and Ioannis Kompatsiaris. Spatio-temporal activity detection and recognition in untrimmed surveillance videos. In *Proceedings of the 2021 International Conference on Multimedia Retrieval*, pages 451–455, 2021.

[35] Shuo Chen, Pascal Mettes, Tao Hu, and Cees GM Snoek. Interactivity proposals for surveillance videos. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pages 108–116, 2020.

[36] Yixuan Li, Zixu Wang, Limin Wang, and Gangshan Wu. Actions as moving points. In *arXiv preprint arXiv:2001.04608*, 2020.

[37] Lin Song, Shiwei Zhang, Gang Yu, and Hongbin Sun. Tacnet: Transition-aware context network for spatio-temporal action detection. *CoRR*, abs/1905.13417, 2019.

[38] Jiaojiao Zhao and Cees G. M. Snoek. Dance with flow: Two-in-one stream action detection. *CoRR*, abs/1904.00696, 2019.

[39] Junting Pan, Siyu Chen, Mike Zheng Shou, Yu Liu, Jing Shao, and Hongsheng Li. Actor-context-actor relation network for spatio-temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 464–474, 2021.

[40] Xitong Yang, Xiaodong Yang, Ming-Yu Liu, Fanyi Xiao, Larry S Davis, and Jan Kautz. Step: Spatio-temporal progressive learning for video action detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[41] Rui Hou, Chen Chen, and Mubarak Shah. Tube convolutional neural network (T-CNN) for action detection in videos. *CoRR*, abs/1703.10664, 2017.

[42] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

[43] Bernard Ghanem Fabian Caba Heilbron, Victor Escorcia and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970, 2015.

[44] Gunnar A. Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Charades-ego: A large-scale dataset of paired third and first person videos. *CoRR*, abs/1804.09626, 2018.

[45] 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. *CoRR*, abs/1507.05738, 2015.

[46] Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, J. K. Aggarwal, Hyungtae Lee, Larry Davis, Eran Swears, Xioyang Wang, Qiang Ji, Kishore Reddy, Mubarak Shah, Carl Vondrick, Hamed Pirsiavash, Deva Ramanan, Jenny Yuen, Antonio Torralba, Bi Song, Anesco Fong, Amit Roy-Chowdhury, and Mita Desai. A large-scale benchmark dataset for event recognition in surveillance video. In *CVPR 2011*, pages 3153–3160, 2011.

[47] Hatem Belhassen, Heng Zhang, Virginie Fresse, and El-Bay Bourennane. Improving video object detection by seq-bbox matching. In *VISIGRAPP*, 2019.

[48] Wei Han, Pooya Khorrami, Tom Le Paine, Prajit Ramachandran, Mohammad Babaeizadeh, Humphrey Shi, Jianan Li, Shuicheng Yan, and Thomas S. Huang. Seq-nms for video object detection. *ArXiv*, abs/1602.08465, 2016.

[49] Alberto Sabater, Luis Montesano, and Ana C. Murillo. Robust and efficient post-processing for video object detection. *CoRR*, abs/2009.11050, 2020.

[50] Kai Kang, Hongsheng Li, Junjie Yan, Xingyu Zeng, Bin Yang, Tong Xiao, Cong Zhang, Zhe Wang, Ruohui Wang, Xiaogang Wang, and Wanli Ouyang. T-cnn: Tubelets with convolutional neural networks for object detection from videos. *IEEE Trans. Cir. and Sys. for Video Technol.*, 28(10):2896–2907, October 2018.

[51] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[52] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset, 2017.

[53] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.

[54] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[55] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541, 2017.