COMPLEX PAIRWISE ACTIVITY ANALYSIS VIA INSTANCE LEVEL EVOLUTION REASONING

Sudipta Paul[⋆] *Carlos Torres*[†] *Shivkumar Chandrasekaran*[‡] *Amit K. Roy-Chowdhury*[⋆]

[⋆] University of California, Riverside, CA [†] TwoSixLabs, LLC [‡] Mayachitra, Inc
spaul007@ucr.edu, carlos.torres@twosixlabs.com, shiv@mayachitra.com, amitrc@ece.ucr.edu

ABSTRACT

Video activity analysis systems are often trained on large datasets. Activities and events in the real world do not occur in isolation, instead, they occur as interactions between related objects. This work introduces a novel method that jointly exploits relational information between pairs of objects and temporal dynamics of each object. The proposed method effectively leverages a new simple architecture that is flexible and easily trained to detect relational activities and events using small datasets (hundreds of samples). The solution is constructed and tested using synthetic videos of car-collision events. The annotated datasets in this work will be made available online to the research community. Experimental results demonstrate the efficacy of the network to perform complex activity analysis.

Index Terms— activity recognition, pairwise activity, relational reasoning, temporal reasoning

1. INTRODUCTION

Recognizing a complex activity that typically involves multiple objects, requires an understanding of the interaction among the objects in space [1]. The dynamics of the relevant objects over time give an important cue for inferring the activity category [2]. The performance of a complex activity analysis system can be improved if the system is able to reason about the relation of the interacting objects in space. Also, the system can use the reasoning to capture the short term and long term evolution of the dynamics of objects with time. For example, in a video of car collision, the collision is most likely to occur when two cars are approaching each other at high velocity. Reasoning over the relative distance and the direction of motion of the two cars in space will boost the performance of a collision detection system. However, if we consider a collision event and a near-miss event (safe event), both of them share the same spatial entity: close proximity of two interacting objects and high velocity. Hence, the system will confuse a near-miss event (safe event) as collision unless it accounts for the change in dynamics of the interacting objects with time. So, reasoning about the spatial interaction as well as the dynamics of an instance over time is critical for recognizing a complex activity.

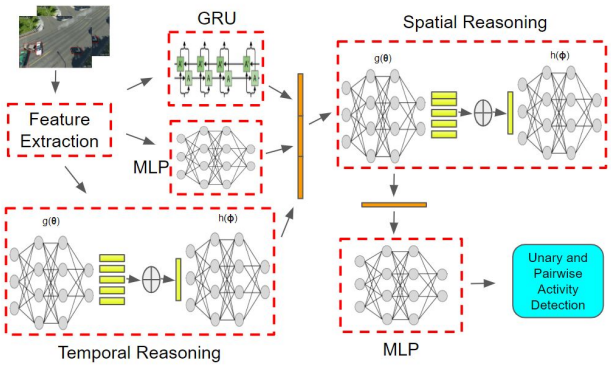

Fig. 1. Overview of the proposed architecture for the collision activity detection system. It will first compute feature embedding for each object appearance of each frame by concatenating the output of bidirectional GRU, two-layer MLP, and temporal reasoning module. This embedding encodes temporal evolution information. Then we use the spatial reasoning module to learn the pairwise interaction. Finally, we compute the activity detection confidence for each pair of objects as well as individual objects of a frame.

Recent works in activity recognition try to model the long term temporal relationship information using RNN [3, 4, 5, 6] and 3D convolution [7, 8, 9, 10]. But these methods extract features from the whole scene and fail to capture the region-based relationships. As observed by [11], most of the actions are classified based on background information instead of capturing key region information. There is work on action recognition that model temporal dynamics and spatial object-object interactions [1, 12, 13, 14]. However, these investigations address the problem of video representation for activity recognition of the entire video, which differs from our target task. Instead of doing activity recognition at the video level, our goal is to detect the involvement and interaction evolution of objects and object pairs in an activity.

In this work, we demonstrate how pairwise as well as

This project was supported in part by the Office Of Naval Research (ONR) N00014-15-C-5113 and Navy (NavAir) N-6833518-C-0199. The content of the information does not necessarily reflect the position or the policy of the Department of Defense or the U.S. Government and no official endorsement should be inferred. Work done in part during Paul and Torres tenure at Mayachitra, Inc.

unary involvement of objects in a complex activity like collision can be detected by enabling spatial and temporal reasoning of the object interactions over time. Detection of objects involved in a collision is important for an autonomous driving system to safely react with proper driving assistance. Regular traffic scenarios are mostly comprised of the natural flow of safe events, whereas anomaly events like collision are typically rare, resulting in a lack of event training data and imbalance in data classes. Again, available collision event data may not exhibit variation across a sufficiently wide range of appearances. This inherent imbalance of data classes, lack of collision event data and insufficient appearance variation between safe events and collision events make the collision detection problem extremely challenging. To account for the wide range appearance invariance between activity classes and lack of event data, we do not use deep feature representation of appearance and dense optical flow. Instead, we work with a low dimensional, simpler and informative representation of interacting objects. Moreover, the chosen representation of the interacting objects is invariant across different scenarios, e.g., background, illumination, viewpoint. Consequently, the activity detection module is robust to a wide range of scene conditions. The main contributions include:

- We propose a novel method that reasons about the relation of the interacting objects and captures the temporal dynamics of objects to boost activity recognition performance.
- We address a novel problem of activity recognition for pairs of objects. We introduce a new dataset **CarBump**, containing synthetic videos of car collision events with pairwise activity annotation.
- We empirically show the activity recognition performance of the proposed method on the CarBump dataset.

2. METHODOLOGY

Consider a training set of n_v videos $\{v_i\}_{i=1}^{n_v}$, where a video v_i consists of frames $\{f_i\}_{i=1}^{n_f}$ and contains objects $\{o_i\}_{i=1}^{n_o}$ and the tracks of the objects are known. For a frame f, containing objects $\{o_i\}_{i=1}^{k}$ $(k \leq n_o)$, we have activity label set $\{a_{ij}\}_{i=1,j=1}^{k,k}$. Here a_{ij} is the activity label of an object pair (o_i, o_j) in frame f. During test time, given for a video v and tracks for objects $\{o_i\}_{i=1}^{n_o}$, our task is to infer the activity label a_{ij} for each object pair of each frame. Inspired by the recent success of [15] for relational reasoning, we device a strategy to encode spatial relation as well as temporal evolution in instance level to accomplish the task of pairwise activity recognition. The overview of our model is visualized in Figure 1.

Feature Extraction. Suppose, for an object instance o in frame f_i, we have the information $\{x_i, y_i, h_i, w_i, \mathcal{M}_i, \rho_i\}$ where (x_i, y_i) is the centroid, h_i, w_i are height and width of the bounding box containing the object, $\mathcal{M}_i$ is the mask and ρ_i is the confidence score of the object class. For the same instance o in frame f_{i-1}, we have $\{x_{i-1}, y_{i-1}, h_{i-1}, w_{i-1}, \mathcal{M}_{i-1}, \rho_{i-1}\}$. We compute the change of position in x and y direction, $\Delta x = x_i - x_{i-1}$ and $\Delta y = y_i - y_{i-1}$, change of bounding box length, $\Delta h = h_i - h_{i-1}$ and $\Delta w = w_i - w_{i-1}$, second order moment of area $\mu_{20}, \mu_{02}, \mu_{21}, \mu_{12}$ from the mask $\mathcal{M}_i$. Feature representation of object o in frame f_i consists of centroid (x_i, y_i), bounding box length $\{h_i, w_i\}$, change of bounding box length $\{\Delta h, \Delta w\}$, change of speed in eight direction $\{\Delta x, \Delta y, -\Delta x, -\Delta y, \frac{1}{\sqrt{2}}(\Delta x + \Delta y), \frac{1}{\sqrt{2}}(\Delta x - \Delta y), \frac{1}{\sqrt{2}}(\Delta y - \Delta x), -\frac{1}{\sqrt{2}}(\Delta x + \Delta y)\}$, second order moment of area $\{\mu_{20}, \mu_{02}, \mu_{21}, \mu_{12}\}$, and confidence score ρ_i. So, the resultant feature embedding for an object instance o in frame f_i is $\boldsymbol{x}_i^o \in \mathcal{R}^{19}$. Contrast of the feature representation during collision event and safe event is evident from Figure 2.

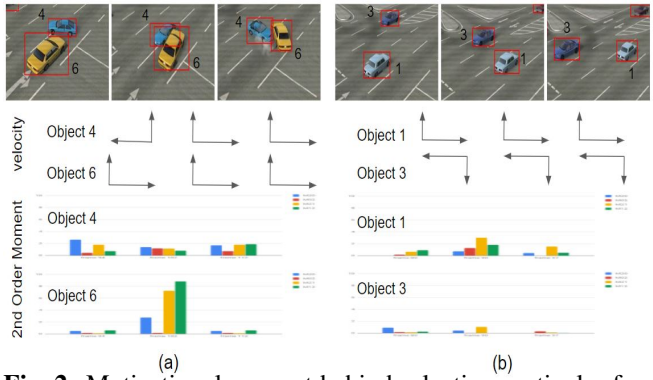

Fig. 2. Motivational concept behind selecting particular features. The scenarios indicated here are: (a) collision event and (b) safe event. For each event, we show three frames from a small segment of the corresponding video clips which are approximately nine frames apart. In this example scenario, during the collision event (a), there is a change of direction of movement for object 4. Again, compared to the safe event scenario, the change of second order moment is higher for collision event scenario.

Bidirectional GRU. Bidirectional GRU is used to encode information of previous and subsequent frames in the feature embeddings. We sequentially input object feature $\boldsymbol{x}^o$ from each frame and obtain embedding $\boldsymbol{x}^{GRU} \in \mathcal{R}^{64}$.

Two-layer MLP. We also use two-layer MLP (Multilayer Perceptron) to compute projection of $\boldsymbol{x}^o$ of each object appearance from each frame to obtain new embedding $\boldsymbol{x}^{MLP} \in \mathcal{R}^{64}$. The two layers consist of 32 and 64 units consecutively.

Temporal Reasoning. In [16], to reason on different frames, the pairwise temporal relation is defined as a composite function,

$$TR(\mathcal{V}) = h_\phi\Big(\sum_{i<j} g_\theta(\boldsymbol{x}_i^f, \boldsymbol{x}_j^f)\Big) \tag{1}$$

Here, input to the temporal relational (TR) network is the video $\mathcal{V}$ with n_f selected ordered frames $\{f_1, f_2, \ldots, f_{n_f}\}$ and $\boldsymbol{x}_i^f$ is the representation of i^{th} frame f_i. Function h_ϕ and g_θ are fused features of different ordered frames. Inspired by their work, instead of using the frame descriptor to reason

on temporal relations for the entire video, we reason on the temporal evolution of each object instance. We use a four-frame temporal relation composite function to represent an object from each frame by,

$$TR(\mathcal{O}) = h_{\phi_t}(\sum_{i<j<k<l} g_{\theta_t}(\boldsymbol{x}_i^o, \boldsymbol{x}_j^o, \boldsymbol{x}_k^o, \boldsymbol{x}_l^o)). \quad (2)$$

Here, the input is $\mathcal{O}$; a set of appearances of object o in m selected frames and $\{\boldsymbol{x}_i^o\}_{i=1}^m$ is the feature representations of m appearances of that object. Instead of considering all combinations of four-frame relationship, we strategically sample five combinations of frames to obtain the reasoning. Consider, we want to compute the temporal relation for an object o^t, which is present in the t^{th} frame. To compute the temporal relationship, we sample features of the same object from the set of features $\mathcal{O}^t = \{\boldsymbol{x}_{t-9}^o, \boldsymbol{x}_{t-6}^o, \boldsymbol{x}_{t-3}^o, \boldsymbol{x}_t^o, \boldsymbol{x}_{t+3}^o, \boldsymbol{x}_{t+6}^o, \boldsymbol{x}_{t+9}^o\}$. By using five combinations of four-frame features, the resulted composite function is,

$$\begin{aligned} TR(\mathcal{O}^t) = \; & h_{\phi_t}(\sum_{i<j<k<l} g_{\theta_t}(\boldsymbol{x}_i^o, \boldsymbol{x}_j^o, \boldsymbol{x}_k^o, \boldsymbol{x}_l^o)) \\ \approx \; & h_{\phi_t}\big[g_{\theta_t}(\boldsymbol{x}_{t-9}^o, \boldsymbol{x}_{t-3}^o, \boldsymbol{x}_{t+3}^o, \boldsymbol{x}_{t+9}^o) \\ & + g_{\theta_t}(\boldsymbol{x}_{t-9}^o, \boldsymbol{x}_{t-6}^o, \boldsymbol{x}_{t-3}^o, \boldsymbol{x}_t^o) + g_{\theta_t}(\boldsymbol{x}_{t-6}^o, \boldsymbol{x}_{t-3}^o, \boldsymbol{x}_t^o, \boldsymbol{x}_{t+3}^o) \\ & + g_{\theta_t}(\boldsymbol{x}_{t-3}^o, \boldsymbol{x}_t^o, \boldsymbol{x}_{t+3}^o, \boldsymbol{x}_{t+6}^o) + g_{\theta_t}(\boldsymbol{x}_t^o, \boldsymbol{x}_{t+3}^o, \boldsymbol{x}_{t+6}^o, \boldsymbol{x}_{t+9}^o)\big]. \end{aligned}$$

Here, the function g_{θ_t} represents a three-layer MLP, parameterized by θ_t and the function h_{ϕ_t} represents a two-layer MLP, parameterized by ϕ_t. Each layer of g_{θ_t} and h_{ϕ_t} has 128 units. The temporal relational network results in a feature embedding $\boldsymbol{x}^{TR} \in \mathcal{R}^{128}$ for an object instance of a frame.

Spatial Reasoning. We concatenate $\boldsymbol{x}^{GRU}$, $\boldsymbol{x}^{MLP}$, and $\boldsymbol{x}^{TR}$ to obtain new representation $\boldsymbol{x}^T \in \mathcal{R}^{256}$ for an object appearance in a frame, ($\odot$ is concatenation operator)

$$\boldsymbol{x}^T = \boldsymbol{x}^{GRU} \odot \boldsymbol{x}^{MLP} \odot \boldsymbol{x}^{TR}. \quad (3)$$

The spatial relational reasoning for each object in a frame is obtained by,

$$SR(\mathcal{F}) = h_{\phi_s}(\sum_{i,j} g_{\theta_s}(\boldsymbol{x}_i^T, \boldsymbol{x}_j^T)) \quad (4)$$

Here, the input to the spatial relational (SR) network is $\mathcal{F}$; set of feature representations $\{\boldsymbol{x}_i^T\}_{i=1}^{n_o}$ for n_o objects $\{o_i\}_{i=1}^{n_o}$ in a frame. The function g_{θ_s} represents a two-layer MLP, parameterized by θ_s and the function h_{ϕ_s} represents a two-layer MLP, parameterized by ϕ_s. The two-layer MLP of function g_{θ_s} consists of 512 and 256 units consecutively. Each layer of h_{ϕ_s} has 256 units. The SR network results in a feature embedding $\boldsymbol{x}^{SR} \in \mathcal{R}^{256}$ for an object instance of a frame.

Pairwise and Unary Activity. To detect the pairwise as well as unary activity, we compute,

$$\boldsymbol{y}_{ij} = g_{\theta_a}(\boldsymbol{x}_i^{SR}, \boldsymbol{x}_j^{SR}). \quad (5)$$

We concatenate each pair of object feature embeddings $(\boldsymbol{x}_i^{SR}, \boldsymbol{x}_j^{SR})$ of a frame and train the two-layer MLP g_{θ_a}, parameterized by θ_a. It generates the logit corresponding to pairwise collision activity. The two layers of g_{θ_a} consists of 128 and 2 units sequentially. We perform softmax classification on the logit $\boldsymbol{y}_{ij}$ to obtain the confidence score a_{ij} for activity classification. Here to be noted that the activity score a_{ij}, where $i = j$, represents the unary activity.

Loss Function. We use the weighted cross entropy loss over each frame. For each batch during training, we calculate the number of positive samples (p) containing collision events and negative samples (n) without collision events and assign weights $n/(p+n)$ and $p/(p+n)$ on the positive and negative samples respectively to account for the class imbalance.

3. DATASET

The CarBump dataset (χ) includes synthetic video game recordings of car collision from oblique traffic-CCTV views. The clips are processed by automated detector (Mask-RCNN [17]) and tracker modules (modified Deep-Sort [18]) to produce tubelets (mask tracks over time). The collision events are manually inspected and annotated with specific frames containing the collision (time) and the instances of the objects involved in the activity of interest (tubelet ID numbers).

Collision Videos. The dataset contains 141 videos containing a total, at least, one collision each (141 collisions). Each video file is, at least, 150 frames (i.e., 5 seconds at 30 fps).

Collision Annotations. The annotations include detections, tubelets and, activity information for each frame. These were manually inspected and edited to only contain traffic object classes that include: bus, car, motorcycle, person, stop sign, traffic light, and truck object labels. The class IDs are consistent with COCO-2014. The annotations are provided as pickles (".pkl"), where each file contains the following data:

- obj_ids: object identification number as int([N,])
- obj_classes: int([N, 80]) (for COCO), if one-hot; else float
- indexed, shape [N,1] or [N,]
- obj_bboxes: float ([N, 4 or 5]); if 5 then 5-th element is confidence (else assumed 1). The first 4 are (col_0, row_0, col_1, row_1)
- obj_events: can be None or [] if no events; otherwise for a collision it contains a list of length N, where each element is an object's list of activities (for that frame; an object may be involved in multiple simultaneous events). Each event (in the list) is a dictionary with {event_id: int, event_type: int, other_objs: list_of_ints} items.
- frame_img_shape: (height, width) or (height, width, L)
- frame_num: int, frame number

4. EXPERIMENTS AND RESULTS

We experimentally evaluate the performance of the proposed activity detection module and discuss the implementation details and the performance metrics used in the evaluation.

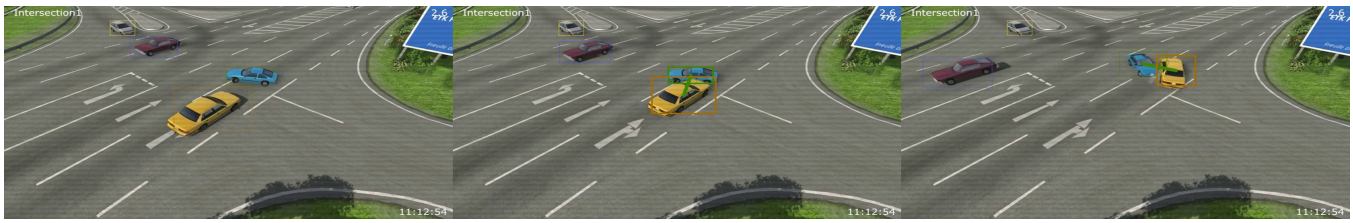

Fig. 3. An example illustration of the performance of the activity detection module. During collision, the pairwise activity detection is marked by green line. Thick bounding box around the object represents the unary activity detection.

Implementation Details We randomly select 70% of the annotated video clips as the training set (98 video clips), 20% of the video clips as the validation set (28 video clips) and the rest of the video clips (15 video clips) as the testing set. The networks are implemented using PyTorch. We use a stratified sampling scheme to sample w length segments of temporally consistent frames. For a video with n_f frames, there can be $n_f - w + 1$ samplable windows and each window is assigned with weight $1/(n_f - w + 1)$. Normalized weights of windows over all the videos are considered as the probability of sampling that window. During training, we use minibatch of four segments, each containing consecutive 100 frames. The initial learning rate is set to $.01/128$ and we drop the learning rate depending on the number of iterations and learning rate scale factor. We use a TITAN X GPU for training the network. The model is trained using ADAM optimizer [19].

Performance Metrics. Mean average precision (mAP) is widely used in activity recognition tasks to quantify the performance. Consider, for a video v with n_f frames and n_o objects throughout the video, there can be $n_f \times n_o$ possible detected regions and $n_f \times n_o \times n_o$ possible pairs, which can be involved in a collision. The activity recognition module perform classification on all of the $n_f \times n_o \times n_o$ pairs.

Quantitative Results. Before presenting and analysing the results, we define all the short notations denoting different experimental setups that we use hereafter.

- ◇ **SPATIAL+TEMPORAL**: Represents the proposed approach described in Section 2.
- ◇ **SPATIAL**: In this approach, temporal reasoning unit is excluded from the setup of SPATIAL+TEMPORAL.
- ◇ **RGB+FLOW**: Adopted approach proposed by [20]. In their proposed model, RPN (Region Proposal Network) is used to generate region proposals and activity classification is done on the proposed regions. Instead of classifying on the proposed regions using RPN, we utilize the available track informations and classify on the known regions.
- ◇ **RGB**: Same architecture as the RGB+FLOW. Instead of using both RGB and Flow information, we only utilize the RGB information.

We compare the performance of our proposed method with three baseline methods. To analyze the performance of the proposed method, we divide the activity recognition task into unary activity recognition task and pairwise+unary activity recognition task. In the unary activity recognition task, the model predicts the activity score for individual instances. In pairwise+unary activity recognition task, the model predicts the activity score for both object pairs and individual objects. We present the result in Table 4. As can be observed from Table 4, our proposed model outperforms other baselines both in unary and pairwise+unary classification tasks with mAP score **0.45** and **0.42** consecutively. From Table 4, it is evident that the performance of baseline methods RGB and RGB+FLOW drops significantly for pairwise activity classification, proving the inadequacy of the methods to capture relational information. Again, the poor performance of SPATIAL demonstrates the importance of incorporating temporal information. Here to be noted that, in the RGB and RGB+FLOW approach, we perform classification on 16 frame segments of each instance considering $IOU = 0.375$.

Table 1. Activity detection performance comparison, where the † symbol indicates baseline implemented by author.

Approach	Unary (mAP)	Pairwise+Unary (mAP)
RGB †	0.30	0.007
RGB+FLOW † [20]	0.18	0.058
SPATIAL	0.092	0.099
SPATIAL+TEMPORAL	**0.45**	**0.42**

Qualitative Results. In Figure 3, we present an example illustration of collision activity detection. In the second and the third frame, the yellow and the blue car colliding with each other are detected by the system. The system was able to detect individual involvement (marked by thick bounding box) as well as pairwise involvement of the two cars in the same accident event (marked by green line).

5. CONCLUSION

In this work, we present a novel approach to encode instance-level spatial and temporal reasoning to boost activity recognition performance. The experimental result suggests that the proposed method is able to distinguish between insufficient appearance variance to detect collision events. Furthermore, the introduced CarBump dataset will promote complex activity analysis using relational reasoning. **Future Work** includes applications to natural videos and extensions to multiple activities (car-person collisions, near-misses, running stop signs and red lights, etc.). In addition, potential directions include developing a network that is more reflective of scene context to optimize spatio-temporal search space and performance.

6. REFERENCES

[1] Fabien Baradel, Natalia Neverova, Christian Wolf, Julien Mille, and Greg Mori, "Object level visual reasoning in videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 105–121.

[2] Xiaolong Wang and Abhinav Gupta, "Videos as space-time region graphs," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 399–417.

[3] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici, "Beyond short snippets: Deep networks for video classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4694–4702.

[4] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[5] Fu Li, Chuang Gan, Xiao Liu, Yunlong Bian, Xiang Long, Yandong Li, Zhichao Li, Jie Zhou, and Shilei Wen, "Temporal modeling approaches for large-scale youtube-8m video understanding," *arXiv preprint arXiv:1707.04555*, 2017.

[6] Antoine Miech, Ivan Laptev, and Josef Sivic, "Learnable pooling with context gating for video classification," *arXiv preprint arXiv:1706.06905*, 2017.

[7] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[8] 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*, 2017, pp. 6299–6308.

[9] 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*, 2018, pp. 6450–6459.

[10] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 305–321.

[11] Gunnar A Sigurdsson, Olga Russakovsky, and Abhinav Gupta, "What actions are needed for understanding human actions in videos?," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2137–2146.

[12] Xiaolong Wang and Abhinav Gupta, "Videos as space-time region graphs," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[13] Ashesh Jain, Amir R. Zamir, Silvio Savarese, and Ashutosh Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[14] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid, "Actor-centric relation network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 318–334.

[15] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap, "A simple neural network module for relational reasoning," in *Advances in neural information processing systems*, 2017, pp. 4967–4976.

[16] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba, "Temporal relational reasoning in videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 803–818.

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask r-cnn," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2980–2988.

[18] Nicolai Wojke, Alex Bewley, and Dietrich Paulus, "Simple online and realtime tracking with a deep association metric," *arXiv preprint arXiv:1703.07402*, 2017.

[19] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[20] Rohit Girdhar, João Carreira, Carl Doersch, and Andrew Zisserman, "A better baseline for ava," *arXiv preprint arXiv:1807.10066*, 2018.