Actor-centered Representations for Action Localization in Streaming Videos

Sathyanarayanan Aakur[1] and Sudeep Sarkar[2]

[1] Oklahoma State University, Stillwater, OK 74074 saakur@okstate.edu
[2] University of South Florida, Tampa, FL, 33620 sarkar@usf.edu

Abstract. Event perception tasks such as recognizing and localizing actions in streaming videos are essential for scaling to real-world application contexts. We tackle the problem of learning *actor-centered* representations through the notion of *continual hierarchical predictive learning* to *localize* actions in streaming videos *without* the need for training labels and outlines for the objects in the video. We propose a framework driven by the notion of hierarchical predictive learning to construct *actor-centered* features by attention-based contextualization. The key idea is that predictable features or objects do not attract attention and hence do not contribute to the action of interest. Experiments on three benchmark datasets show that the approach can learn robust representations for localizing actions *using only one epoch of training*, i.e., a single pass through the streaming video. We show that the proposed approach outperforms unsupervised and weakly supervised baselines while offering competitive performance to fully supervised approaches. Additionally, we extend the model to multi-actor settings to recognize group activities while localizing the multiple, plausible actors. We also show that it generalizes to out-of-domain data with limited performance degradation.

1 Introduction

Understanding events in videos requires understanding beyond recognition, such as localizing the actor, understanding their future behavior from current and past observations, and building robust representations at the event and actor levels. While many recent works have focused on action recognition [1,18,21] and action localization [6,20,8], significant progress has primarily been driven by the use of large-scale, annotated training data. While self-supervised learning [7,42] has reduced the need for labeled data for *recognition*, there is still a dependency on large amounts of manual annotations for *localization*.

We consider the problem of learning *actor-centered* representations to localize actions in *streaming videos* i.e., needing a single-pass through video for training (single epoch) and without training labels and outlines. We do not need multiple training epochs to build the representations. We define an actor-centered representation as a compositional structure of the scene that encodes the properties (location, geometry, and relational cues) of the *dominant* actor contributing to the action of interest. For example, in Figure 1, there are many actors (three

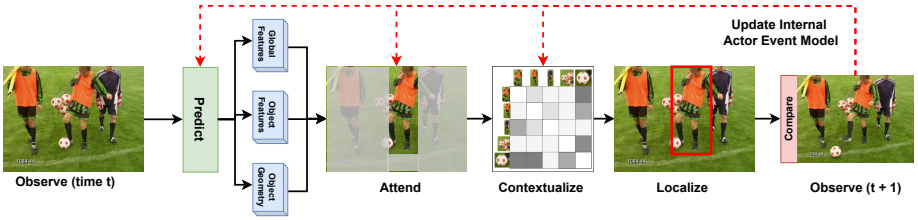

Fig. 1. Our goal is to learn actor-centered representations for actor localization in streaming videos *without explicit annotations*. Given a frame at time t, we follow the sequence of: observe, predict, attend based on prediction error, contextualize actor representations, and localize. The internal event models are constantly updated based on the prediction errors with the observation at time $t + 1$.

players, a soccer ball, etc.) in the scene, but only one *dominant actor* (the player in the middle) is involved in the action "*kicking ball*". Hence, an actor-centered representation would encode the appearance and geometry of the player in the middle and *contextualize* their features concerning the other objects in the scene. Such representations allow us to capture action-specific contextual cues in a generalizable representation.

We build actor-centered event representations by contextualizing the actor's features with environment-level (or scene) at both a perceptual level (such as color, texture, and movement) and a conceptual level (such as actor-environment interactions and action goals). Computationally, we model this process by following a sequence of operations given by *observe, predict, compare, attend, contextualize*, and *localize*. Figure 1 illustrates this process. The *key idea* is that predictable features or objects do not attract attention and hence do not contribute to the action of interest. We introduce the idea of hierarchical prediction that enables the framework to select objects of interest and maintain context in prediction to localize the action by navigating spurious motion patterns such as camera motion and background clutter. This hierarchical prediction differs from prior versions of predictive learning for action recognition [2,3], which do not consider the actor-centered features such as appearance, geometry, and their evolution with respect to the scene.

The **contributions** are four-fold: (i) we introduce the idea of *hierarchical* predictive learning to learn actor-centered representations to localize actions in *streaming videos* in an *unsupervised manner*, (ii) introduce a novel, attention-driven formulation for learning robust, actor-centered event features for action localization and recognition, (iii) demonstrate that the proposed approach can be trivially extended to multi-actor group activity recognition and localization, and (iv) show that the use of actor-centered feature representations helps learn robust features that can generalize to data from outside the training domain *without finetuning for both localization and group activity recognition.*

2 Related Work

Action localization has largely been tackled through *supervised* learning approaches [9,12,14,35,36,39,41,43,45], which aim to simultaneously generate bounding box proposals and labels learned from annotated training data. The common pipeline uses convolutional neural networks (both 2D and 3D [40]) to extract features from RGB images, optionally the optical flow images, and generate bounding box proposals to localize objects in the video sequence. A linking algorithm (Viterbi or actor linking [6]) is used to extract action tubes from the generated bounding boxes. Annotated training data is used to train recognition and bounding box regression modules.

Weakly supervised [6,21,32] reduce the dependency on training data by negating the need for spatial-temporal annotations and using either attention-based pooling [21,32] or appearance-based linking from generic object detection-based proposals [6]. They typically require video-level label annotations that are used to learn representations for recognition and use object-level labels and characteristics to select bounding box proposals from pre-trained object detection models. Hence, they may be constrained to localizing actions specific to classes from the detection models.

Unsupervised approaches [3,37] do not require annotations for labels or bounding boxes. Soomro *et al.* [37] use pre-trained object detection models to generate proposals and score each with a "humanness" score that ranks the likelihood of belonging to an action class and uses a knapsack-based algorithm to discover action classes to self-label videos. Aakur *et al* [3] use a predictive learning-based approach (PredLearn for brevity) to create spatial-temporal attention maps which are used to localize objects of interest. Closely related to our approach, PredLearn anticipates the future spatial feature using a motion-weighted loss function at the feature level. However, it does not enforce consistency in actor-specific features such as geometry or contextualized representations to help reject the background clutter and maintain context in prediction. Additionally, we localize multiple actors together with learning robust features.

3 Actor-centered Action Localization

Problem Formulation. In our setup, we consider the problem of localizing the *dominant* action a_i at each time instant t in a *streaming video*. Each video can contain multiple objects in the scene with one *dominant* action performed by one or more *actors*. The key challenge is ignore clutter and identify the object(s) of interest (i.e., the *actors*) *without any supervision* while building robust representations that capture the motion and relational dynamics of the event. Figure 2 illustrates the proposed action localization framework. We begin with perceptual features extracted from a convolutional neural network [34] and progressively refine these scene-level features with context from event-level dynamics (Section 3.2) and actor-centered context (Section 3.3) through the notion of hierarchical predictive learning (Section 3.4), to jointly model both the evolution of the action and the actors in a unified framework.

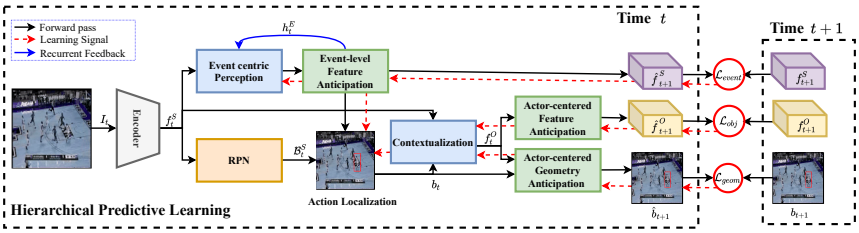

Fig. 2. Overview of our approach. Given a sequence of frames in *streaming fashion*, our model constructs an actor-centered representation using the notion of hierarchical predictive learning. A prediction-driven attention map is used to localize the action.

3.1 Extracting Perceptual Features

First, we extract a global, scene-level representation of the given visual sequence. This representation includes both perceptual features and identifying regions of likely interest representing objects. While our approach is general enough to handle different object proposal approaches (see Section 4), we use a pre-trained convolutional neural network to extract the scene-level representation (f_t^S) and use a Single-Short Object Detector (SSD) [23] layer to generate region-proposals ($\mathcal{B}_t^S$), where objects are likely to exist. Following prior work in [3], we make the SSD class-agnostic by considering *all* bounding boxes returned (at an "objectness" threshold of 0.01) regardless of the predicted class and their corresponding confidence scores. This allows us to remove biases towards certain actors, such as human actors, and help handle any visual variations (such as pose and occlusion) that can cause missed detection.

3.2 Event-centric Perception

The second step in the proposed model is to construct a feature representation of the current scene (at time t) influenced by the observed event's spatial-temporal dynamics. While CNN features provide an efficient spatial representation, it does not consider the contextual knowledge provided by temporal transitions and spatial interactions among the scene's entities. This process requires modeling a stable event representation h_t^E and an attention mechanism α_t^S that uses this global representation to jointly perceive and anticipate the spatial and temporal dynamics of the event. Formally, we define the event-centric perception model as a prediction function that maximizes the probability $P(\hat{f}_{t+1}^S | W_p, \alpha_t^S, h_t^E, f_t^S)$, where $\hat{f}_{t+1}^S$ is the anticipated features at time $t+1$ conditioned on an internal event representation h_t^E, a temporally-weighted, spatial attention function α_t^S and the current observed features f_t^S. W_p is the set of learnable parameters in this module. There are two steps in constructing the event-centric perceptual features - (i) learning an efficient global, event-level representation and (ii) using the learned event representation to drive the perception in a recurrent manner. First, we first create the event-centric scene representation by weighting the CNN feature f_t^S of the frame at time t by an attention vector (α_t^S), influenced

by the spatial temporal dynamics of the current event. Hence, the event-centric representation is given by $f_t^E{=}\alpha_t^S \odot f_t^S$, where $\alpha_t^S{=}f_a(f_t^S, h_{t-1}^E)$ and $f_a(\cdot)$ is a learned attention function [4]. Second, we use a hierarchical stack of Long Short Term Memory (LSTM) networks [10] to construct the internal event representation. We take a continual predictive learning approach, inspired by [2,3], to learn an efficient global representation of the event that captures the relevant spatial-temporal patterns to provide context for event-based perception. The hierarchical LSTM stack is used as a spatial-temporal decoder network. It takes a sequence of event-centric image features as input and propagates its prediction up the stack. The output of the top-most LSTM is taken as the anticipated features ($\hat{f}_{t+1}^S$) at the next time step. Hence, the hierarchical LSTM stack acts as a generative model that learns and uses a stable event representation to anticipate the scene's spatial and temporal evolution. Formally, this is represented as

$$\hat{f}_{t+1}^{\ell}, h_t^{\ell} = LSTM(\hat{f}_{t+1}^{\ell-1}, W_{\ell}, h_{t-1}^{\ell}) \tag{1}$$

$$\hat{f}_{t+1}^{0}, h_t^{0} = LSTM(f_t^E, W_0, h_{t-1}^{0}) \tag{2}$$

where $\hat{f}_{t+1}^{\ell-1}$ refers to the predicted features at the ℓ^{th} LSTM in the stack and W^{ℓ} refers to the weights associated with the LSTM at the ℓ^{th} layer; Equation 2 shows the initialization for the bottom-most 0^{th}-level LSTM.

Note that $\hat{f}_{t+1}^{\ell}$ for the top most LSTM network is taken as the prediction for time $t+1$ and the corresponding hidden state is taken as the event representation such that $h_t^E{=}h_t^{\ell}$. The memory is not shared within the stack and hence allows each level of the stack to model the spatial-temporal dynamics at different granularity, with the ℓ^{th}-level LSTM influenced by the lower-level LSTMs.

Event-level prediction. The event-centric perception module is trained in a predictive learning approach with the training objective given by

$$\mathcal{L}_{event} = \|f_{t+1}^S - f_t^S\|_2 \odot \|f_{t+1}^S - \hat{f}_{t+1}^S\|_2 \tag{3}$$

where the first term represents the weighted difference between the *features* at consecutive time steps t and $t+1$ and the resulting value $\mathcal{L}_{event}$ represents a weighted $L-2$ norm of the predicted and expected value that penalizes incorrect predictions at spatial locations with maximal change at the *feature level*. Hence, $\mathcal{L}_{event}$ is the prediction-based drive, a measure of the effectiveness of the learned event representation at a coarse spatial quantization. This predictive learning process forms the bottom level of the hierarchy and helps learn event-level dynamics for modeling the action in the scene. While effective, as shown in PredLearn [3], it is not enough to handle complex scenes and multiple actors in the scene. There is a need to model the actor-environment interactions for effective visual understanding.

3.3 Contextualization: Actor-centered Features

The next step is to construct actor-centered representations that contextualize the event-level dynamics with the actor-environment interactions. We consider

a feature representation of a scene to be *actor-centered* if the resulting representation can (i) reject clutter in the scene, (ii) reduce the impact of background or spurious motion patterns, and (iii) *contextualize* the actor's motion dynamics with the rest of the scene or environment. This representation is analogous to a posterior-weighted spatial representation that highlights areas of interest while suppressing spatially irrelevant features. In our framework, the posterior is obtained by updating the prior, captured by the prediction-based error from Equation 3, with the current observation (f_t^S). The contextualized representation is obtained by computing the dot-product attention [25] between the posterior-weighted representation and the actual representation and is defined as

$$f_t^O = GAP(softmax(f_t^S \odot \mathcal{F}_t^S) \odot \mathcal{F}_t^S) \tag{4}$$

where GAP refers to the Global Average Pooling function [22] and $\mathcal{F}_t^S$ is a contextualized feature representation conditioned by the posterior probability provided by the spatial-temporal prediction loss $\mathcal{L}_{Event}$. We compute this function as $\mathcal{F}_t^S$=$softmax(\mathcal{L}_{Event}) \odot f_t^S$, which intuitively provides a representation that rejects clutter by scaling down the spatial regions that do not contribute to the prediction uncertainty. These areas typically involve background scenes or actors whose actions are more predictable and less likely to be of interest. This formulation helps preserve the scene's spatial-temporal structure by summing out any trivial motion-based changes, making it more robust to spurious motion patterns in the input, such as those induced by background noise and small camera motion. Empirically, this formulation results in a more robust video-level representation that can generalize across domains (see Section 4.2).

3.4 Hierarchical Predictive Learning

We use the notion of *continual, hierarchical predictive* learning to train the model end-to-end without needing labels and outlines of the objects in the video. This approach aims to model the dynamics of the observed event at different levels of granularity, moving beyond just scene-level dynamics [3] or temporal dynamics [2]. To this end, we create a hierarchy of predictions that are performed at every time step that models the event-level and actor-level dynamics in the event. At the lowest level is the event-level prediction (Section 3.2). At the next level is the prediction of the actor's dynamics within the scene's context. At the top level is the prediction of the actor's visual properties. The goal is to anticipate the actor's location and geometry in the context of event-level and actor-level dynamics. Each level of the stack influences the prediction of the upper level and hence forms a hierarchy of predictions that capture the inherent dynamics within the event. The actor-level predictions (levels 2 and 3) are conditioned on the event-level representation by constructing a global representation given by $\hat{f}_t^E = GAP(\alpha_t^S \odot \hat{f}_t^S)$, where $\hat{f}_t^S$ refers to the anticipated spatial features (from the previous prediction step) and α_t^S refers to the spatial attention constructed (conditioned on the current observation) at time t. This formulation of the global

representation forces the model to learn spatially relevant features that are important *across time steps* and hence helps ensure that the event representation is robust by acting as temporal smoothing.

Computationally, we learn two LSTM-based prediction models that use this global representation ($\hat{f}_t^E$) to anticipate the actors' dynamics in terms of contextualized features and geometry. One LSTM anticipates the *changes* in the actor's geometry rather than directly predicting the BB location, which allows the predictor to focus on the *evolution* of geometry. The other LSTM anticipates the actor-centered representations ($\hat{f}_{t+1}^O$) at time $t+1$. Hence, the goal of these two LSTMs is to minimize the actor-centered prediction errors defined as

$$\mathcal{L}_{object} = \|f_{t+1}^O - \hat{f}_{t+1}^O\|_2 + \mathcal{D}_{bb}(b_{t+1}, \hat{b}_{t+1}) + \mathcal{D}_g(b_{t+1}, \hat{b}_{t+1}) \tag{5}$$

where $\mathcal{D}_{bb}(b_{t+1}, \hat{b}_{t+1})$ is the distance between the predicted bounding box and the actual observed bounding centers; $\mathcal{D}_g(b_{t+1}, \hat{b}_{t+1}) = (\sqrt{w} - \sqrt{\hat{w}})^2 + (\sqrt{h} - \sqrt{\hat{h}})^2$, where ($\hat{h}$ and $\hat{w}$) and (h, w) are the predicted and actual height and widths of the bounding box bb_{t+1}, respectively. Hence, the entire framework is trained end-to-end using the overall objective function given by

$$\mathcal{L}_{total} = \lambda_1 \frac{1}{n_f} \sum_{i=1}^{w_f} \sum_{j=1}^{h_f} \mathcal{L}_{event} + \lambda_2 \mathcal{L}_{object} \tag{6}$$

where λ_1 and λ_2 are modulating factors to balance the trade-off between predicting the event-level and object-level prediction errors. Both losses directly penalize the event-level representation (h_t^E), the spatial attention (α_t^S) and the contextualization module ($\mathcal{F}_t^S$. Hence it adds an implicit regularization to prevent overfitting since the model's parameters are updated *continuously per frame*. The resulting spatial-temporal loss $\mathcal{L}_{event}$ can then considered to be reflective of the *predictability* of both the actor *and* scene. Hence, spatial locations with a higher error indicate the location of the actor [3,11]. Note that the entire process is unsupervised, there are no labels or bounding box annotations needed for training since the predictions at time t are compared to observations at time $t+1$ to provide supervision to progressively refine the representations.

3.5 Attention-based Action Localization

The final step in the proposed approach is using attention to localize the actor (the object of interest) in the given video. We create an attention-like representation using the prediction-based error $\mathcal{L}_{event}$ to identify areas of interest. The input to the localization process consists of (i) initial regions of interests generated based on spatial features $\mathcal{B}_t^S$ (from Section 3.1), (ii) the spatial-temporal prediction error $\mathcal{L}_{event}$ (from Section 3.2), (iii) number of attention "grids" to consider K, and (iv) the total number of bounding box predictions per frame t. We first construct an attention-like representation by running the spatial-temporal prediction error through a *softmax* function to produce an attention

map of shape $c_x \times c_y$ where c_x and c_y are spatial dimensions of the observed feature maps, with each point corresponding to a "grid" in the frame (following notation from YOLO [29]). The softmax operation magnifies areas of high errors while suppressing areas of low prediction errors.

We consider areas of high prediction error to be regions of interest. However, we allow the attention map to be split between multiple objects and consider the top K grids (sorted based on prediction error) to select bounding box localization. Following the notation from YOLO-based object detection models [29], we define a binary function $\mathbb{1}(\cdot)$ that returns *True* if a bounding box proposal's center falls within the "grid" $e_{i,j}$ and *False* otherwise. This allows us to select objects that are most likely to contribute to the grid's prediction error. Note that this is different from [3], where each bounding box is assigned an energy term based on distance from a prior position and the magnitude of the prediction error, which does not allow them to attend to multiple objects. This is further explored in Section 4, where the use of hierarchical prediction allows the model to attend to multiple objects simultaneously for multi-actor localization.

Implementation Details. In our experiments, we use a VGG-16 network [34], pre-trained on ImageNet [31], as the backbone network for training a Single Shot Multbox Detector (SSD) [23] to extract frame-level representations and generate localization proposals. The SSD is trained on MS-COCO with input re-sizes to 512×512. We use the output of the max-pooling layer after the fifth convolutional layer as f_t^S. We use the SSD as a class-agnostic region proposal network by taking the bounding box proposals without any predicted classes or associated probabilities. The number of layers ℓ in the hierarchical prediction network (in Section 3.2) as 3 and set the dimensions of the hidden state at each layer to 512. We set the number of attention grids $K = 5$ and the number of localization per frame $N = 10$ (Section 3.5). We train with adaptive learning [2], with an initial learning rate of 1×10^{-10} and scaling factors $\Delta_t^- = 0.1$ and $\Delta_t^+ = 0.01$.

4 Experimental Evaluation

4.1 Data, Metrics and Baselines

We use three standard benchmark datasets (UCF Sports [30], JHMDB [15], and THUMOS'13[17]) to evaluate the proposed approach for action localization. We also evaluate on the Collective Activity dataset [5] to demonstrate and evaluate our approach on multi-actor action localization. **UCF Sports** [30] contains 10 classes characterizing sports-based actions such as weight-lifting and diving. We use the official splits containing 103 videos for training and 47 videos for testing, as defined in [19] for evaluation. **JHMDB** [15] has 21 action classes from 928 trimmed videos, each annotated with human joints and bounding box for every frame. It offers several significant challenges for unsupervised action localization, such as camera motion that causes significant occlusions and background objects that act as distractions. We report all results as the average across all three splits. **THUMOS'13** [17] (or the *UCF-101-24* dataset) is a subset of the UCF-101 [38] dataset, consisting of 24 classes and 3,207 videos. It is one of the

Approach	Supervision		UCF Sports		JHMDB		THUMOS'13	
	Spatial	Label	σ=0.2	σ=0.5	σ=0.2	σ=0.5	σ=0.2	σ=0.5
Tube CNN [12]	✓	✓	0.47	-	-	0.77	0.47	0.41
Action Tubelets [14]	✓	✓	0.53	0.27	-	-	0.48	-
Action Tubes [9]	✓	✓	0.56	0.49	0.55	0.45	-	-
MRSTL [48]	✓	✓	-	-	-	0.37	-	0.68
MENET [24]	✓	✓	-	-	-	**0.82**	-	**0.84**
HISAN [27]	✓	✓	-	-	-	0.77	-	0.73
ACAR-Net [26]	✓	✓	-	-	-	-	-	0.84
ALSTM [32]	✗	✓	-	-	-	-	0.06	-
VideoLSTM [21]	✗	✓	-	-	-	-	0.37	-
Actor Supervision [6]	✗	✓	-	**0.48**	-	**0.36**	**0.46**	-
Soomro *et al* [37]	✗	✗	0.46*	0.30*	**0.43***	**0.22***	0.21*	0.06*
PredLearn [3] (k=k_{gt})	✗	✗	0.55	0.32	0.30	0.10	0.31	0.10
AC-HPL (Ours, k=k_{gt})	✗	✗	**0.70**	**0.59**	**0.43**	0.15	**0.38**	**0.20**

Table 1. Comparison with state-of-the-art approaches on three common benchmark datasets - UCF Sports, JHMDB and THUMOS'13. We report the video-level mAP at different overlap thresholds. * refers to the use of class-specific object proposals.

most challenging action localization datasets with complex motion, background clutter and high intra-class variability. Following prior works [21,37], we report results on the first split. **Collective Activities** [5] is a group activity dataset where the goal is to recognize the activity performed by multiple actors such as talking, queueing, and walking. It comprises 44 short video sequences with 5 group activities, with every 10 frames annotated with bounding boxes of actors involved in the group activity. This dataset offers a unique challenge in localizing all actors (>= 1) involved in the action while learning robust features that can capture the dynamics of each actor in the context of their collective activity. We follow prior works [46,13,44,28,8] and use 1/3 of the video sequences for testing and the rest for training. We report results for both recognition and localization.

Label Prediction and Metrics. Due to the unsupervised nature of learning representations, we use *k-means* clustering to obtain class labels. The frame-level features are max-pooled to obtain video-level features. Following prior work [3,16,47], we use the Hungarian method to map from predicted clusters to the ground-truth labels. We set the number of clusters to the number of classes in the ground-truth for comparison with state-of-the-art. For *action localization*, we report the mean average precision (mAP) metric at different overlap thresholds for a fair comparison with prior works [21,37].

Baselines. We compare against several fully supervised baselines such as MRSTL [48], MENET [24], HISAN [27], ACAR-Net [26], tube convolution networks [12], motion-based action tublets [14] and action tubes [9] and weakly supervised such as ALSTM [32], VideoLSTM [21] and Actor Supervision [6]. We also evaluate our approach against unsupervised action localization approaches such as Soomro *et al.* [37] and the closely related predictive learning approach [3], which we term as PredLearn. Note, we compare against PredLearn when the number of clusters is set to the ground-truth clusters ($k = k_{gt}$).

Test Data → / Train Data ↓	UCF Sports		JHMDB		THUMOS'13	
	AC-HPL	PredLearn	AC-HPL	PredLearn	AC-HPL	PredLearn
	σ=0.5		σ=0.2		σ=0.2	
UCF Sports	**0.59**	0.32	**0.39**	0.19	**0.38**	0.20
JHMDB	**0.48**	0.23	**0.43**	0.30	**0.35**	0.26
THUMOS'13	**0.50**	0.27	**0.40**	0.24)	**0.38**	0.31

Table 2. Generalization capability when evaluated on out-of-domain test samples *without finetuning*. PredLearn refers to [3] and AC-HPL refers to our approach.

Approach	Supervision?		Acc.
	Label	Box	
LRCN [28]	✓	✓	64.0
VGG-16 [28]	✓	✗	68.3
VGG-16 [28]	✓	✓	71.2
Hierarchical LSTM [13]	✓	✓	81.1
CERN [33]	✓	✓	84.8
stagNET [28]	✓	✓	89.1
ARG [46]	✓	✓	91.0
Action Transformer [8]	✓	✓	92.8
GroupFormer [20]	✓	✓	96.3
AC-HPL (k-means)*	✗	✗	72.2
AC-HPL (Finetuned)*	✓	✗	**80.2**

(a)

Approach	Avg. IOU	Recall	mAP	
			0.2	0.5
PredLearn	0.18	0.172	0.378	0.011
AC-HPL (K=1)	0.205	0.181	0.442	0.017
AC-HPL (K=5)	0.271	0.284	0.545	0.068
AC-HPL (K=10)	0.342	0.396	0.551	0.14
AC-HPL (K=25)	0.472	0.634	0.723	0.449

* denotes features are trained for only 1 epoch on the target dataset.

(b)

Table 3. Evaluation of our approach on the **Collective Activities** dataset for (a) multi-actor group activity recognition and (b) multi-actor group activity localization.

4.2 Quantitative Analysis

We first present the quantitative results of the proposed in Table 1, where we compare against different baseline approaches. We report the mean average precision (mAP) scores over the most commonly reported overlap thresholds of 0.2 and 0.5. The approaches are ordered by the amount of supervision required for training. The models at the top require *strong* supervision in terms of spatial annotations such as bounding boxes *and* video-level labels to localize and classify the action. The models in the middle are *weakly* supervised and hence only require video-level labels for training. The approaches at the bottom require no training annotations. It can be seen that our approach outperforms all baselines, including fully supervised models, on the *UCF Sports* dataset, even at higher thresholds. Interestingly, we significantly outperform the closely related PredLearn by a significant margin ($\approx 15\%$ in absolute mAP).

On datasets with significantly higher complexity, such as JHMDB and THUMOS'13, we see consistent improvements over the other unsupervised models such as PredLearn and Soomro *et al.*'s action discovery approach, that use bounding box proposals from *class-specific* proposals and hence are restricted to objects (humans) that are present in the pre-trained object detection models. On the other hand, we use *class-agnostic* proposals and are not restricted to any object class. Also, it is interesting to note that hierarchical predictive learning and actor-centered feature representations help overcome the challenges posed by occlusions and clutter, as indicated by the significant gains over PredLearn at higher thresholds on JHMDB and THUMOS'13.

Generalization to Novel Domains. In addition to evaluating the proposed approach in traditional settings, we also assess its ability to generalize to

novel domains. To be specific, we check its generalization capability by training on one dataset and testing its performance on a different dataset *without finetuning*. We begin by evaluating the approach on the generalization task by training the model on the training data from one of the three standard benchmarks (UCF Sports, JHMDB - Split 1, and THUMOS'13) and evaluating on the others. While the three datasets have similar actions, they have varying amounts of data, camera motion, and occlusions, which provide a challenging benchmark for evaluating generalization performance. We also report the closely related PredLearn approach's performance, which does not use hierarchical prediction and actor-centered representations. Table 2 summarizes the results. It can be seen that the proposed approach generalizes well across datasets, *regardless of the training data size*. For example, UCF Sports has a very small number of training data (103) and classes (10). However, the model can transfer well to other datasets with more classes (21 for JHMDB and 24 for THUMOS'13). It is to be noted that the model is *not finetuned* on any data in the target domain yet performs as well as weakly supervised models such as VideoLSTM (0.37 mAP@0.2 on THUMOS'13), *which was trained on the data.* Similarly, the use of actor-centered representations allows for better generalization compared to PredLearn, which has a poorer recognition performance due to the lack of *contextualized* feature representations.

4.3 Multi-Actor Group Activity Localization

Our approach can be naturally extended to multi-actor group activity recognition and localization using the Collective Activities dataset [5]. The goal is to recognize the collective or group activity performed by the majority of the actors in the scene. While majority of the prior works [28,33,46,8,20] have focused on *recognition*, there have not been efforts for multi-actor *localization*. Using the prediction errors outlined in Section 3.5, we can increase the number of attention *grids* (K_{attn}) and use the resulting attention points to localize multiple actors in a given scene. We show that our approach can attend to multiple actors and learn robust representations for simultaneous localization and recognition in multi-actor videos. Table 3(a) summarizes the performance for *recognition*, while Table 3(b) summarizes the performance for *localization*. For recognition, we evaluate two versions of the proposed approach to generate the labels - a completely unsupervised version with k-means for prediction and a fine-tuned version where the features are categorized into classes using a 2-layer feedforward neural network. As can be seen, without extensive training and annotations such as bounding boxes, we can achieve a recognition of 80.2%, which is remarkable considering that other state-of-the-art approaches require large amounts of training annotations and epochs. We only need one epoch of training to finetune the features to the multi-actor setting and do not need any annotations for learning. For localization, we report the average IOU of all bounding boxes produced by the approach and the mAP at 0.2 IOU and 0.5 IOU. To evaluate the upper bound of the approach, we also compute recall by considering groundtruth bounding boxes with at least one attention point as a true positive (TP). As can be seen

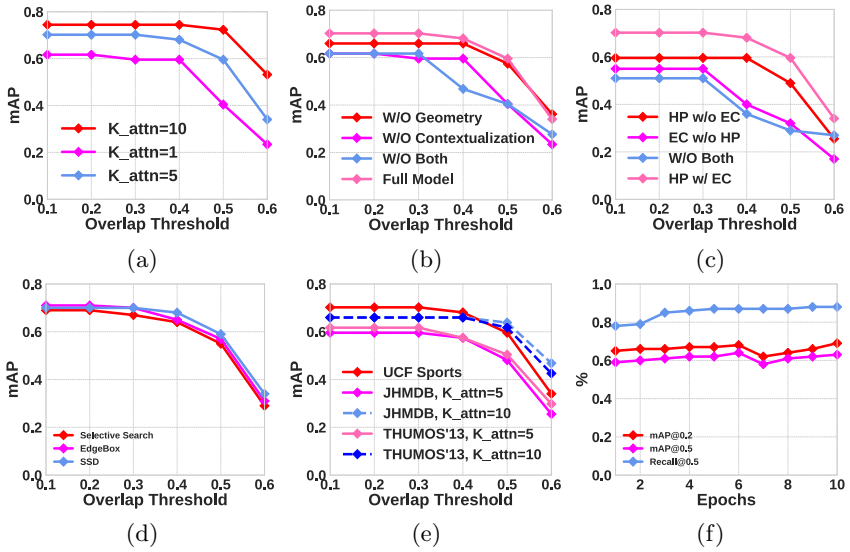

Fig. 3. Ablation experiments on UCF Sports to evaluate the effect of (a) number of attention "grids", (b) actor-centered prediction, and (c) event-centric perception, (d) choice of region proposal, (e) out-of-domain data, and (e) number of training epochs.

from Table 3(b), as the number of attention points (K_{attn}) increases, we achieve a better IOU, recall, and mAP. Note that the closely related PredLearn is not able to handle multiple actor localization since their attention focuses only on the dominant *actor* and considers other actors as clutter. Nevertheless, these results and the qualitative visualizations in Figure 4 show that the approach can perform multi-actor localization without any bells and whistles while not being explicitly trained for the task.

4.4 Ablation Studies

In Figure 3(a) we show the effect of the changing the number of attention "*grids*" (K_{attn}) (Section 3.5). As the K_{attn} increases, the localization performance also increases and allows the model to keep track of the object of interest even if there are other potential actors. We also evaluate the effect of the different terms in the actor-centered prediction loss (Equation 5). As can be seen from Figure 4(b), the use of both geometry prediction and contextualized feature prediction help improve the performance significantly, with the use of contextualized prediction providing a greater jump in performance. In Figure 4(c), we present the effect of using event-centric perceptual features (Section 3.2) on the framework with and without hierarchical prediction. It can be that the use of both improves the performance, especially at higher thresholds, indicating that the use of hierarchical prediction with event-centric features helps attend to areas of interest.

Effect of Region Proposal Methods. To evaluate the effect of object detection modules on the proposed approach, we try other bounding box proposals from

untrained approaches such as EdgeBox and Selective Search. We present the results on UCF Sports below in Figure 4(d). As can be seen, our approach is not dependent on SSD as the object proposal mechanism and is able to use any region proposal mechanism as its input. We do not use any class labels from SSD and make it class-agnostic to ensure that we do not have any assumptions about the actor or domain semantics, unlike supervised or weakly-supervised approaches that use object detectors as part of the action proposals.

Effect of out-of-domain data. We also evaluated the localization performance of the approach when trained with out-of-domain data. Figure 3(e) shows that the real performance drop is at higher overlap thresholds when testing on data outside of the training domain. However, increasing K_{attn} helps alleviate this issue and even outperforms models trained in the same domain.

Effect of multiple training epochs. Although our approach is designed to work with one epoch of training, we also evaluate the impact of multi-epoch training on the UCF Sports dataset and present the results in Figure 3(f). It can be seen that increasing the number of epochs allows the model to learn better features for recognition while the localization is improved as well.

4.5 Qualitative Analysis

We qualitatively analyze our approach and visualize some interesting instances in Figure 4. We show two specific groups of examples - (i) a comparison with PredLearn [3] to highlight the importance of using actor-centered features and hierarchical prediction beyond numbers presented in Section 4, and (ii) some failure modes of the approach to identify possible ways to mitigate them. In rows 1 and 4, it can be seen that although there are other objects in the scene whose motion is unpredictable, the use of multiple attention grids and actor-centered prediction helps the model to maintain focus on the actor. Row 3 shows that the model can overcome the challenges posed by camera motion *and* object deformation to maintain context in prediction, whereas PredLearn (without hierarchical prediction) is influenced by the camera motion and loses track of the object. We also visualize some of the failure modes of the proposed model in the final row in Figure 4. In particular, we would like to highlight two areas that lead to failure. First, consider the sequence on the left. The model's attention is initially on the wrong player in the scene and continues to attend to areas of the same player, which we attribute to the actor-centered prediction. Although the object is well localized, it is not the *labeled* object of interest to which the model does not have access. The second failure mode, highlighted in the sequence on the right, is bounding box selection. Although the attention is on the object for most frames, *class-agnostic* proposals returns poorer bounding box fit. The mAP score does not take these factors into account.

5 Conclusion

We showed that we can learn actor-centered representations to localize actions with just a single-pass through video for training (single epoch) and without

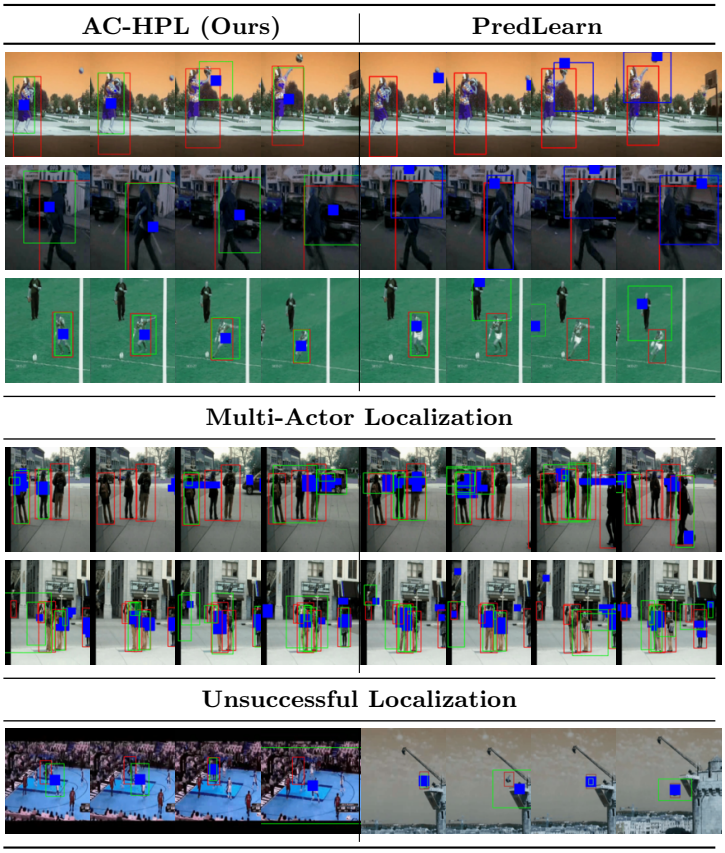

Fig. 4. Qualitative Examples. *Top:* handling camera motion and background motion to maintain context in localization. *Middle*: multi-actor localizations from AC-HPL. *Bottom*: Unsuccessful localizations from AC-HPL. *Visualization Legend:* Red BB: Grountruth, Green BB: Predictions, Blue Squares: Attention Locations.

training labels and outlines. We do not need multiple training epochs to build the representations. This makes the approach useful for many real-world context where storing the video raises privacy or high storage cost concerns. Our solution was a hierarchical predictive learning framework that continuously predicts and learns from errors at different granularities. The resulting spatial-temporal error localized the action. The model leverages a novel actor-centered representation to learn robust features that mitigate the effect of camera motion and background clutter. We showed that we can beat SOTA on unsupervised action localization and multi-actor group activity localization while generalizing to novel domains *without finetuning*.

Acknowledgements. This research was supported in part by the US National Science Foundation grants CNS 1513126, IIS 1956050, IIS 2143150, and IIS 1955230.

References

1. Aakur, S., de Souza, F.D., Sarkar, S.: Going deeper with semantics: Exploiting semantic contextualization for interpretation of human activity in videos. In: IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE (2019)
2. Aakur, S.N., Sarkar, S.: A perceptual prediction framework for self supervised event segmentation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
3. Aakur, S.N., Sarkar, S.: Action localization through continual predictive learning. arXiv preprint arXiv:2003.12185 (2020)
4. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
5. Choi, W., Shahid, K., Savarese, S.: What are they doing?: Collective activity classification using spatio-temporal relationship among people. In: 2009 IEEE 12th International Conference on Computer Vision Workshops. pp. 1282–1289. IEEE (2009)
6. Escorcia, V., Dao, C.D., Jain, M., Ghanem, B., Snoek, C.: Guess where? actor-supervision for spatiotemporal action localization. Computer Vision and Image Understanding **192**, 102886 (2020)
7. Gan, C., Gong, B., Liu, K., Su, H., Guibas, L.J.: Geometry guided convolutional neural networks for self-supervised video representation learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5589–5597 (2018)
8. Gavrilyuk, K., Sanford, R., Javan, M., Snoek, C.G.: Actor-transformers for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 839–848 (2020)
9. Gkioxari, G., Malik, J.: Finding action tubes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 759–768 (2015)
10. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
11. Horstmann, G., Herwig, A.: Surprise attracts the eyes and binds the gaze. Psychonomic Bulletin & Review **22**(3), 743–749 (2015)
12. Hou, R., Chen, C., Shah, M.: Tube convolutional neural network (t-cnn) for action detection in videos. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 5822–5831 (2017)
13. Ibrahim, M.S., Muralidharan, S., Deng, Z., Vahdat, A., Mori, G.: A hierarchical deep temporal model for group activity recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1971–1980 (2016)
14. Jain, M., Van Gemert, J., Jégou, H., Bouthemy, P., Snoek, C.G.: Tubelets: Unsupervised action proposals from spatiotemporal super-voxels. International Journal of Computer Vision **124**(3), 287–311 (2017)
15. Jhuang, H., Gall, J., Zuffi, S., Schmid, C., Black, M.J.: Towards understanding action recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3192–3199 (2013)
16. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9865–9874 (2019)
17. Jiang, Y.G., Liu, J., Zamir, A.R., Toderici, G., Laptev, I., Shah, M., Sukthankar, R.: Thumos challenge: Action recognition with a large number of classes (2014)

18. Kuehne, H., Arslan, A., Serre, T.: The language of actions: Recovering the syntax and semantics of goal-directed human activities. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 780–787 (2014)
19. Lan, T., Wang, Y., Mori, G.: Discriminative figure-centric models for joint action localization and recognition. In: 2011 International Conference on Computer Vision. pp. 2003–2010. IEEE (2011)
20. Li, S., Cao, Q., Liu, L., Yang, K., Liu, S., Hou, J., Yi, S.: Groupformer: Group activity recognition with clustered spatial-temporal transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13668–13677 (2021)
21. Li, Z., Gavrilyuk, K., Gavves, E., Jain, M., Snoek, C.G.: Videolstm convolves, attends and flows for action recognition. Computer Vision and Image Understanding **166**, 41–50 (2018)
22. Lin, M., Chen, Q., Yan, S.: Network in network. arXiv preprint arXiv:1312.4400 (2013)
23. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European Conference on Computer Vision. pp. 21–37. Springer (2016)
24. Liu, Y., Tu, Z., Lin, L., Xie, X., Qin, Q.: Real-time spatio-temporal action localization via learning motion representation. In: Proceedings of the Asian Conference on Computer Vision (2020)
25. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025 (2015)
26. Pan, J., Chen, S., Shou, M.Z., Liu, Y., Shao, J., Li, H.: Actor-context-actor relation network for spatio-temporal action localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 464–474 (2021)
27. Pramono, R.R.A., Chen, Y.T., Fang, W.H.: Hierarchical self-attention network for action localization in videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 61–70 (2019)
28. Qi, M., Qin, J., Li, A., Wang, Y., Luo, J., Van Gool, L.: stagnet: An attentive semantic rnn for group activity recognition. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 101–117 (2018)
29. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7263–7271 (2017)
30. Rodriguez, M.D., Ahmed, J., Shah, M.: Action mach a spatio-temporal maximum average correlation height filter for action recognition. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2008)
31. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
32. Sharma, S., Kiros, R., Salakhutdinov, R.: Action recognition using visual attention. In: Neural Information Processing Systems: Time Series Workshop (2015)
33. Shu, T., Todorovic, S., Zhu, S.C.: Cern: confidence-energy recurrent network for group activity recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5523–5531 (2017)
34. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

35. Soomro, K., Idrees, H., Shah, M.: Action localization in videos through context walk. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3280–3288 (2015)
36. Soomro, K., Idrees, H., Shah, M.: Predicting the where and what of actors and actions through online action localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2648–2657 (2016)
37. Soomro, K., Shah, M.: Unsupervised action discovery and localization in videos. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 696–705 (2017)
38. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
39. Tian, Y., Sukthankar, R., Shah, M.: Spatiotemporal deformable part models for action detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2642–2649 (2013)
40. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4489–4497 (2015)
41. Tran, D., Yuan, J.: Max-margin structured output regression for spatio-temporal action localization. In: Advances in neural information processing systems. pp. 350–358 (2012)
42. Wang, J., Jiao, J., Bao, L., He, S., Liu, Y., Liu, W.: Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4006–4015 (2019)
43. Wang, L., Qiao, Y., Tang, X.: Video action detection with relational dynamic-poselets. In: European Conference on Computer Vision. pp. 565–580. Springer (2014)
44. Wang, M., Ni, B., Yang, X.: Recurrent modeling of interaction context for collective activity recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3048–3056 (2017)
45. Weinzaepfel, P., Harchaoui, Z., Schmid, C.: Learning to track for spatio-temporal action localization. In: Proceedings of the IEEE international conference on computer vision. pp. 3164–3172 (2015)
46. Wu, J., Wang, L., Wang, L., Guo, J., Wu, G.: Learning actor relation graphs for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9964–9974 (2019)
47. Xie, J., Girshick, R., Farhadi, A.: Unsupervised deep embedding for clustering analysis. In: International Conference on Machine Learning (ICML). pp. 478–487 (2016)
48. Zhang, D., He, L., Tu, Z., Zhang, S., Han, F., Yang, B.: Learning motion representation for real-time spatio-temporal action localization. Pattern Recognition **103**, 107312 (2020)