# Layout-induced Video Representation for Recognizing Agent-in-Place Actions

Ruichi Yu[1,2*] Hongcheng Wang[2]
Ang Li[1] Jingxiao Zheng[1] Vlad I. Morariu[3†] Larry S. Davis[1]
[1]University of Maryland, College Park [2]Comcast Applied AI Research [3]Adobe Research
[1]{yrcbsg, angli, jxzheng, lsd}@umiacs.umd.edu,
[2]hongcheng_wang@comcast.com, [3]morariu@adobe.com

## Abstract

*We address scene layout modeling for recognizing agent-in-place actions, which are actions associated with agents who perform them and the places where they occur, in the context of outdoor home surveillance. We introduce a novel representation to model the geometry and topology of scene layouts so that a network can generalize from the layouts observed in the training scenes to unseen scenes in the test set. This Layout-Induced Video Representation (LIVR) abstracts away low-level appearance variance and encodes geometric and topological relationships of places to explicitly model scene layout. LIVR partitions the semantic features of a scene into different places to force the network to learn generic place-based feature descriptions which are independent of specific scene layouts; then, LIVR dynamically aggregates features based on connectivities of places in each specific scene to model its layout. We introduce a new Agent-in-Place Action (APA) dataset to show that our method allows neural network models to generalize significantly better to unseen scenes.*

## 1. Introduction

Recent advances in deep neural networks have brought significant improvements to many fundamental computer vision tasks, including video action recognition [8, 36, 46, 48, 22, 37, 19, 4]. Current action recognition methods are able to detect, recognize or localize general actions and identify the agents (people, vehicles, *etc.*) [8, 36, 46, 22, 37, 19, 4, 7, 28, 26, 42, 40, 18]. However, in applications such as surveillance, relevant actions often involve locations and moving directions that relate to to scene layouts–for example, it might be of interest to detect (and issue an alert about) a person walking towards the front door of a house, but not to detect a person walking along the sidewalk. So, what makes an action "interesting" is how the it interacts with the geometry and topology of the scene.

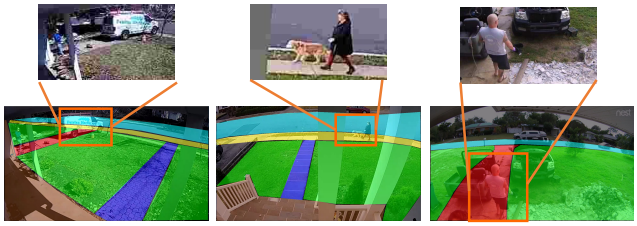



Figure 1. Example agent-in-place actions and segmentation maps. Different colors represent different places. We zoom in to the agents performing the actions for clarity. An agent-in-place action is represented as <*agent, action, place*>. Same colors indicate same place types (*e.g.,* green for lawn, blue for walkway, *etc.*).

Examples of these actions in outdoor home surveillance scenarios and the semantic segmentation maps of scenes are shown in Fig.1. We refer to these actions as "*agent-in-place*" actions to distinguish them from the widely studied generic action categories. From those examples we observe that although the types of place are limited (*e.g.,* street, walkway, lawn), the layout (*i.e.,* structure of places, including reference to location, size, appearance of places and their adjacent places) vary significantly from scene to scene. Without large-scale training data (which is hard to collect considering privacy issues), a naive method that directly learns from raw pixels in training videos without layout modeling can easily overfit to scene-specific patterns and absolute pixel coordinates, and exhibit poor generalization on layouts of new scenes.

To address the generalization problem, we propose the *Layout-Induced Video Representation* (LIVR), which explicitly models scene layout for action recognition by encoding the layout in the network architecture given semantic segmentation maps. The representation has three com-

*Work done at the Comcast Applied AI Research.
†Was affiliated with the University of Maryland during the work.

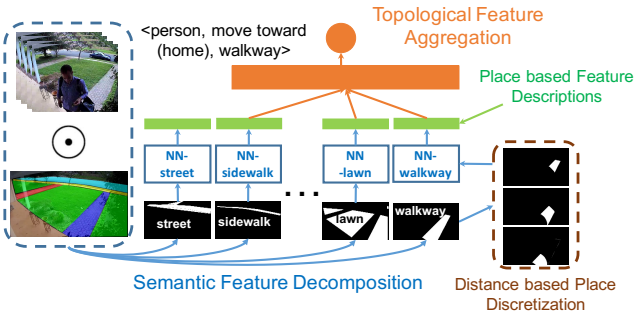
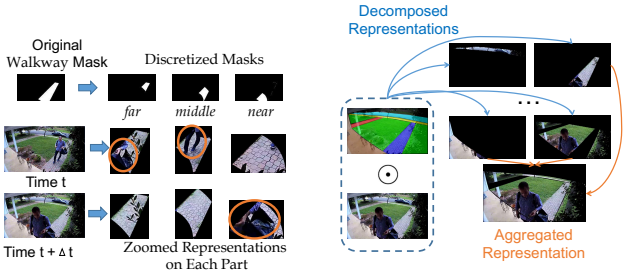



Figure 2. Framework of LIVR. Given the segmentation map, we decompose the semantic features into different places and extract place-based feature descriptions individually. Then we dynamically aggregate them at inference time according to the topology of the scene. ⊙ denotes the masking operation for spatial decomposition. "NN" stands for neural network.

ponents: *1)* A semantic component represented by a set of bitmaps used for decomposing features in different *"places"* (*e.g.,* walkway, street, *etc.*), which forces the network to learn place-based features that are independent of scene layout; *2)* A geometric component represented by a set of coarsely quantized distance transforms of each semantic place incorporated into the network to model moving directions; *3)* A topological component represented through the connection structure in a dynamically gated fully connected layer of the network–essentially aggregating representations from adjacent (more generally $h$-connected for $h$ hops in the adjacency graph of the semantic map) places. By encoding layout information (class membership of places, layout geometry and topology) into the network architecture using this decompotision-aggregation framework, we encourage our model to abstract away low-level appearance variations and focus on modeling high-level scene layouts, and eliminate the need to collect massive amounts of training data.

The first two components require *semantic feature decomposition* (Fig.2). We utilize bitmaps encoded with the semantic labels of places to decompose video representations into different places and train models to learn place-based feature descriptions. This decomposition encourages the network to learn features of generic place-based motion patterns that are independent of scene layouts. As part of the semantic feature decomposition, we encode scene geometry to model moving directions by discretizing a place into parts based on a quantized distance transform w.r.t. another place. Fig.2 (brown) shows the discretized bitmaps of *walkway* w.r.t. *porch*. As illustrated in Fig.3(a), features decomposed by those discretized bitmaps capture moving agents in spatial-temporal order, which reveals the moving direction, and can be generalized to different scene layouts.

The actions occurring in one place may be projected onto adjacent places from the camera view (see Fig.3(b)).




(a) Place Discretization (b) Topological Feature Aggregation

Figure 3. (a) illustrates distance-based place discretization. We segment the bit mask representing a given semantic class based on the distance transform with respect to a second class, to explicitly represent the spatial-temporal order of moving agents which captures the moving direction w.r.t. that place. For example, this figure shows the partition of the *walkway* map into components that are "far," "middle" and "near" to the porch class. We use *move toward (home)* action as an example: we first observe the person on the part of *far* and *middle* (distance), and after some time, the person appears in the part of *near*. We use orange ellipses to highlight person parts. (b) illustrates the motivation behind topological feature aggregation. We seek a representation that covers the entire body of the person, which can be accomplished by aggregating the masked images from places that are connected to *walkway*.

We propose *topological feature aggregation* to dynamically aggregate the decomposed features within the place associated with that action and adjacent places. The aggregation controls the "on/off" state of neuron connections from generic place-based feature descriptions to action nodes to model scene layout based on topological connectivity among places.

We created the Agent-in-Place Action (APA) dataset, which to the best of our knowledge, is the first dataset that addresses recognizing actions associated with scene layouts. APA dataset contains over 5,000 15s videos obtained from 26 different surveillance scenes with around 7,100 actions from 15 categories. To evaluate the generalization of LIVR, we split the scenes into observed and unseen scenes. Extensive experiments show that LIVR significantly improves the generalizability of the model trained on only observed scenes and tested on unseen scenes (improving the mean average precision (mAP) from around 20% to more than 50%). Consistent improvements are observed on almost all action categories.

## 2. Related Work

**Video Action Recognition Methods and Datasets.** Recent advances in video action recognition were driven by many large scale action recognition datasets. UCF101 [37], HMDB [22] and Kinetics [19] were widely used for recognizing actions in video clips [42, 30, 47, 8, 36, 46, 7, 28,

26, 40, 18, 43]; THUMOS [17], ActivityNet[4] and AVA [13] were introduced for temporal/spatial-temporal action localization [34, 50, 27, 38, 54, 55, 3, 5, 24]. Recently, significant attention has been drawn to model human-human [13, 39] and human-object interactions in daily actions [32, 35, 44, 29]. In contrast to these datasets that were designed to evaluate motion and appearance modeling, or human-object interactions, our Agent-in-Place Action (APA) dataset is the first one that focuses on actions that are defined with respect to scene layouts, including interaction with places and moving directions. Recognizing these actions requires the network to not only detect and recognize agent categories and motion patterns, but also how they interact with the layout of the semantic classes in the scene. With the large variations of scene layouts, it is critical to explicitly model scene layout in the network to improve generalization on unseen scenes.

**Surveillance Video Understanding.** Prior work focuses on developing robust, efficient and accurate surveillance systems that can detect and track actions or events [14, 9, 6, 56, 15, 49]. Recently, ReMotENet [52] skips expensive object detection [31, 12, 10, 11, 23] and utilizes 3D ConvNets to detect motion of an object-of-interest in surveillance videos. We employ a similar 3D ConvNet model as proposed in [52] as a backbone architecture for extracting place-based feature descriptions for our model.

**Knowledge Transfer.** The biggest challenge of agent-in-place action recognition is to generalize a model trained with limited scenes to unseen scenes. Previous work on knowledge transfer for both images and videos has been based on visual similarity, which requires a large amount of training data [2, 16, 25, 41, 1, 51, 53]. For trajectory prediction, Ballan *et al.*[2] transferred the priors of statistics from training scenes to new scenes based on scene similarity. Kitani *et al.*[21] extracted static scene features to learn scene-specific motion dynamics for predicting human activities. Instead of utilizing low-level visual similarity for knowledge transfer, our video representation abstracts away appearance and location variance and models geometrical and topological relationships in a scene. which are more abstract and easier to generalize from limited training scenes.

# 3. Layout-Induced Video Representation

## 3.1. Framework Overview

The network architecture of layout-induced video representation is shown in Fig.4. For each video, we stack sampled frames of a video clip into a 4-D tensor. Our backbone network is similar to the architecture of ReMotENet [52], which is composed of 3D Convolution (3D-conv) blocks. A key component of our framework is semantic feature decomposition, which decomposes feature maps according to

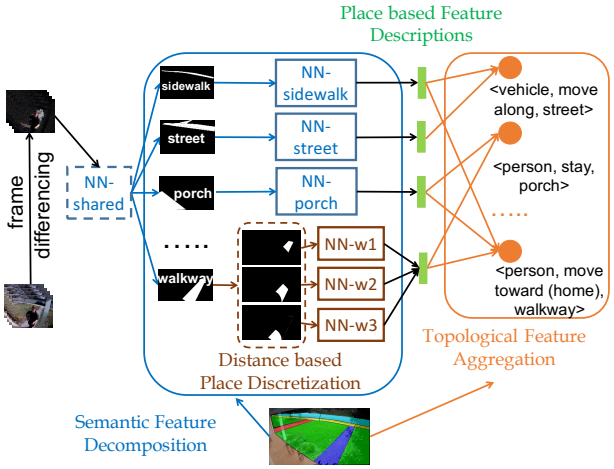



Figure 4. Layout-induced Video Representation Network: The dashed blue box indicates a shared 3D ConvNet to extract low-level features. We utilize the segmentation maps to decompose features into different places, and the solid blue boxes indicate that we train place-based models to extract place-based feature descriptions. When relevant to the activities of interest, we conduct distance-based place discretization to model moving directions; finally, we leverage the connectivity of places to aggregate the place-based feature descriptions at inference level.

region semantics obtained from given segmentation masks. This feature decomposition can be applied after any 3D-conv layer. Spatial Global Max Pooling (SGMP) is applied to extracted features within places, allowing the network to learn abstract features independent of shapes, sizes and absolute coordinates of both places and moving agents. For predicting each action label, we aggregate features from different places based on their connectivity in the segmentation map, referred to as Topological Feature Aggregation.

## 3.2. Semantic Feature Decomposition

**Segmentation Maps.** Semantic Feature Decomposition utilizes a segmentation map of each place to decompose features and cause the network to extract place-based feature descriptions individually. The segmentation maps can be manually constructed using a mobile app we developed [1] to annotate each place by drawing points to construct polygons. This is reasonable since most smart home customers have only one or two cameras in their home. We employ human annotations because automatic semantic segmentation methods segment places based on appearance (*e.g.,* color, texture, *etc.*), while our task requires differentiation between places with similar appearance based on functionality. For example, walkway, street and driveway have different functionalities in daily life, which can be easily and efficiently differentiated by humans. However, they may

[1] Details for the app can be found in the supplementary materials.

confuse appearance-based methods due to their similar appearance. Furthermore, since home surveillance cameras are typically fixed, users can annotate one map per camera very efficiently. However, we will discuss the performance of our method using automatically generated segmentation maps in Sec. 5.4.

**Place-based Feature Descriptions (PD).** Given a segmentation map, we extract place-based feature descriptions as shown in the blue boxes in Fig.4. We first use the segmentation map to decompose feature maps spatially into regions, each capturing the motion occurring in a certain place. The decomposition is applied to features instead of raw inputs to retain context information[2]. Let $\mathbf{X}_L \in \mathbb{R}^{w_L \times h_L \times t_L \times c}$ be the output tensor of the $L^{th}$ conv block, where $w_L, h_L, t_L$ denote its width, height and temporal dimensions, and $c$ is the number of feature maps. The place-based feature description of a place indexed with $p$ is

$$f_{L,p}(\mathbf{X}_L) = \mathbf{X}_L \odot \mathbb{I}\left[\mathbf{M}_L = p\right] \tag{1}$$

where $\mathbf{M}_L \in \mathbb{I}^{w_L \times h_L \times 1}$ is the segmentation index map and $\odot$ is a tiled element-wise multiplication which tiles the tensors to match their dimensions. Place descriptions can be extracted from different levels of feature maps. $L = 0$ means the input level; $L > 0$ means after the $L^{th}$ 3D-conv blocks. A higher $L$ generally allows the 3D ConvNet to observe more context and to abstract features. We treat $L$ as a hyper-parameter and study its effect in Sec. 5.

**Distance-based Place Discretization (DD).** Many actions are naturally associated with moving directions *w.r.t.* some scene element (*e.g.,* the house in home surveillance). To learn general patterns of the motion direction in different scenes, we further discretize the place segmentation into several parts, and extract features from each part and aggregate them to construct the place-based feature description of this place. For illustration, we use *porch* as the anchor place (shown in Fig.5). We compute the distance between each pixel and the *porch* in a scene (distance transform), and segment a place into $k$ parts based on their distances to *porch*. The left bottom map in 5 shows the porch distance transform of a scene. Let $D_L(x)$ be the distance transform of a pixel location $x$ in the $L^{th}$ layer. The value of a pixel $x$ in the part indexing map $\mathbf{M}_L^{\Delta}$ is computed as

$$M_L^{\Delta}(x) = \left\lfloor \frac{D_L^{\max}(x) - D_L^{\min}(x)}{k(D_L(x) - D_L^{\min}(x))} \right\rfloor \tag{2}$$

where $D_L^{\max}(x) = \max\{D_L(x') | M_L(x') = M_L(x)\}$ and $D_L^{\min}(x) = \min\{D_L(x') | M_L(x') = M_L(x)\}$ are the max and min of pixel distances in the same place. They can be efficiently pre-computed. The feature description corresponding to the $i^{th}$ part of $p^{th}$ place in $L^{th}$ layer is

$$f_{L,p,i}^{\Delta}(\mathbf{X}_L) = \mathbf{X}_L \odot \mathbb{I}[\mathbf{M}_L = p \wedge \mathbf{M}_L^{\Delta} = i] \tag{3}$$

where $\odot$ is the tiled element-wise multiplication.

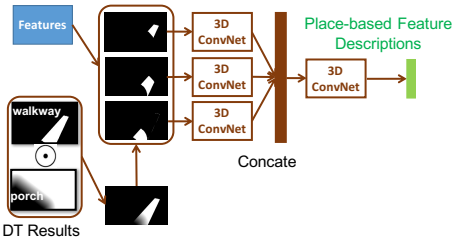



Figure 5. The process of distance-based place discretization.

Discretizing a place into parts at different distances to the anchor place and explicitly separating their spatial-temporal features allows the representation to capture moving agents in spatial-temporal order and extract direction-related abstract features. However, not all places need to be segmented since some places (such as *sidewalk*, *street*) are not associated with any direction-related action (*e.g.,* moving toward or away from the house). For these places, we still extract the whole-place feature descriptors $f_{L,p}$. We discuss the effect of different choices of place discretization and the number of parts $k$, and show the robustness of our framework to these parameters in Sec. 5. To preserve temporal ordering, we apply 3D-conv blocks with spatial-only max pooling to extract features from each discretized place, and concatenate them channel-wise. Then, we apply 3D-conv blocks with temporal-only max pooling to abstract temporal information. Finally, we obtain a 1-D place-based feature description after applying GMP (see Fig.5). The final description obtained after distance-based place discretization has the same dimensionality as non-discretized place descriptions.

### 3.3. Topological Feature Aggregation (Topo-Agg)

Semantic feature decomposition allows us to extract a feature description for each place individually. To explicitly model the layout of a scene, we need to aggregate these place features based on connectivity between places. Each action category is one-to-one mapped to a place. To predict the confidence of an action $a$ occurring in a place $p$, features from adjacent places might provide contextual information, while the ones extracted far from place $p$ are distractors. To explicitly model the topological structure of places in a scene, we propose *Topological Feature Aggregation*, which utilizes the spatial connectivity between places to guide feature aggregation.

Specifically, as shown in Fig.6, given a scene segmentation map, a source place $p$ and a constant $h$, we employ

[2]An agent can be located at one place, but with part of its body projected onto another place in the view of the camera. If we use the binary map as a hard mask at input level, then for some places such as *sidewalk*, *driveway* and *walkway*, only a small part of the moving agents will remain after the masking operation.

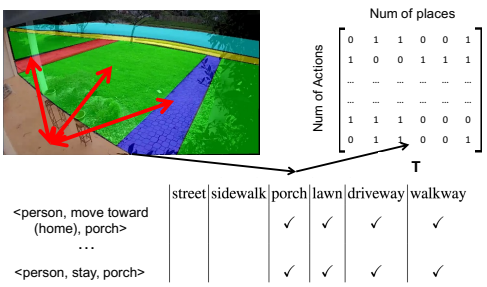



Figure 6. Topological feature aggregation which utilizes the connectivities between different places in a scene to guide the connections between the extracted place-based feature descriptions and the prediction labels. For clear visualization, we use the source places as *porch* in (b) with $h = 1$. The ✓ indicates we aggregate features from a certain place to infer the probability of an action.

a Connected Component algorithm to find the $h$-connected set $C_h(p)$ which contains all places connected to place $p$ within $h$ hops. The constant $h$ specifies the minimum number of steps to walk from the source to a destination place. Given the $h$-connected place set $C_h$, we construct a binary action-place matrix ($\mathbf{T} \in \mathbb{R}^{n_a \times n_p}$) for the scene where $n_a$ is the number of actions and $n_p$ is the number of places. $\mathbf{T}_{i,j} = 1$ if and only if place $j$ is in the $C_h$ of the place corresponding to action $i$. Fig.6 shows an example segmentation map with its action-place mapping, where $C_0(porch) = \{porch\}$, $C_1(porch) = \{porch, walkway, driveway, lawn\}$, $C_2(porch)$ includes all except for *street*, and $C_3(porch)$ covers all six places. It is worth noting that since the vocabulary of our actions is closed, $\mathbf{T}$ is known at both training and testing time given the segmentation maps.

We implement topological feature aggregation using a gated fully connected layer with customized connections determined by the action-place mapping $\mathbf{T}$. Given $n_p$ $m$-D features extracted from $n_p$ places, we concatenate them to form a $(n_p \times m)$-D feature vector. We use $\mathbf{T}$ to determine the "on/off" status of each connection of a layer between the input features and the output action nodes. Let $\mathbf{T}_* = \mathbf{T} \otimes \mathbb{I}^{1 \times m}$ be the actual mask applied to the weight matrix $\mathbf{W} \in \mathbb{R}^{n_a \times n_p m}$ where $\otimes$ is the matrix Kronecker product. The final output is computed by $\mathbf{y} = (\mathbf{W} \odot \mathbf{T}_*)\mathbf{f}_*$, where $\odot$ is element-wise matrix multiplication, $\mathbf{f}_*$ is the concatenated feature vector as the input of the layer. We omit bias for simplicity. Let $J$ be the training loss function (cross-entropy loss). The derivative of $\mathbf{W}$ is $\nabla_{\mathbf{W}} J = (\nabla_{\mathbf{y}} J \mathbf{f}_*^{\mathsf{T}}) \odot \mathbf{T}_*$, which is exactly the usual gradient $(\nabla_{\mathbf{s}} J \mathbf{f}^T)$ masked by $\mathbf{T}_*$. At training time, we only back-propagate the gradients to connected neurons.

## 4. Agent-in-Place Action Dataset

We introduce a video dataset for recognizing agent-in-place actions[3]. We collected outdoor home surveillance videos from internal donors and webcams to obtain over $7,100$ actions from around $5,000$ 15-second video clips with $1280 \times 720$ resolution. These videos are captured from 26 different outdoor cameras which cover various layouts of typical American front and back yards.

We select 15 common agent-in-place actions to label and each is represented as a tuple containing an action, the agent performing it, and the place where it occurs. The agents, actions, and places involved in our dataset are: $agent = \{person, vehicle, pet\}$; $action = \{move\ along, stay, move\ away\ (home), move\ toward\ (home), interact\ with\ vehicle, move\ across\}$; $place = \{street, sidewalk, lawn, porch, walkway, driveway\}$.

The duration of each video clip is 15s, so multiple actions can be observed involving one or more agents in one video. We formulate action recognition as a multi-label classification task. We split the 26 cameras into two sets: observed scenes (5) and unseen scenes (21) to balance the number of instances of each action in observed and unseen scenes and at the same time cover more scenes in the unseen set. We train and validate our model on observed scenes, and test its generalization capability on the unseen scenes. Details about the APA dataset including statistics can be found in the supplementary material.

## 5. Experiments

### 5.1. Implementation Details

**Network Architecture.** Unlike traditional 3D ConvNets which conduct spatial-temporal max-pooling simultaneously, we found that decoupling the pooling into spatial-only and temporal-only leads to better performance (details are in the supplementary materials). We utilize nine blocks of 3D ConvNets with the first five blocks using spatial-only max pooling and the last four blocks using temporal-only max pooling for each place-specific network. The first two blocks have one 3D-conv layer each, and there are two conv layers with ReLU in between for the remaining blocks. For each place-specific network, we use 64 $3 \times 3 \times 3$ conv filters per 3D-conv layer. After conducting SGMP on features extracted by each place-specific network, the final concatenated 1-D feature dimension is $6 \times 64$ for 6 places. The inference is conducted with a gated fully connected layer, whose connections ("on/off" status) are determined by action labels and scene topology. We decompose semantics after the second conv blocks ($L = 2$); we conduct distance-based place discretization on $PL_{DT} = \{walkway, driveway, lawn\}$ and choose $k = 3$;

[3]The dataset is pending legal review.

Table 1. **The Path from Traditional 3D ConvNets to our Methods.** B/L1 and B/L2 are baseline models with raw pixels and and ConcateMap as input, respectively. For our proposed models: V1 uses segmentation maps to extract place-based feature descriptions only. V3 applies distance-based place discretization for some places. Both V1 and V3 use a FC layer to aggregate place features; V2 and V4 uses topological feature aggregation. H and FPS2 indicates using higher resolutions and FPS, and MF means using more filters per conv layer. Besides our baselines, we also compare LIVR with two state-of-the-art action recognition methods: [52, 45].

| Network Architecture | B/L1 | B/L2 | B/L2 +MF | LIVR-V1 | LIVR-V2 | LIVR-V3 | LIVR-V4 | LIVR-V4+H | LIVR-V4+MF | LIVR-V4+FPS2 | TSN [45] | ReMotENet [52] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3D ConvNet? | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | - | - |
| ConcateMap? | | ✓ | ✓ | | | | | | | | - | - |
| place-based feature description? | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | - | - |
| distance-based place discretization? | | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | - | - |
| topological feature aggregation? | | | | | ✓ | | ✓ | ✓ | ✓ | ✓ | - | - |
| higher resolutions? | | | | | | | | ✓ | | | - | - |
| more filters? | | | ✓ | | | | | | ✓ | | - | - |
| higher FPS? | | | | | | | | | | ✓ | - | - |
| Observed scenes mAP | 51.09 | 54.12 | 53.02 | 55.69 | 57.12 | 58.02 | **59.71** | 59.64 | 59.52 | 59.01 | 56.71 | 55.92 |
| Unseen scenes mAP | 19.21 | 21.16 | 20.45 | 41.57 | 43.78 | 47.76 | 50.65 | 49.03 | **50.98** | 49.56 | 23.21 | 22.05 |

for topological feature aggregation, we choose $h = 1$.

**Anchor Place.** For our dataset, the directions mentioned are all relative to the house location, and *porch* is a strong indicator of the house location. So we only conduct distance transform to *porch* [4], but the distance-based place discretization method can represent moving direction w.r.t any arbitrary anchor place.

**Training and Testing Details.** Our action recognition task is formulated as multi-label classification without mutual exclusion. The network is trained using the Adam optimizer [20] with 0.001 initial learning rate. For input video frames, we follow [52] to use FPS 1 and down-sample each frame to $160 \times 90$ to construct a $15 \times 160 \times 90 \times 3$ tensor for each video as input. Suggested by [52], small FPS and low resolution are sufficient to model actions for home surveillance where most agents are large and the motion patterns of actions are relatively simple. We evaluate the performance of recognizing each action independently and report Average Precision (AP) for each action and mean Average Precision (mAP) over all categories.

**Dataset Split.** We split the 26 scenes into two sets: observed scenes and unseen scenes. We further split the observed scenes into training and validation sets with a sample ratio of nearly $1:1$. The model is trained on observed scenes and test on unseen scenes. The validation set is used for tuning hyperparameters, which are robust with different choices (see Sec. 5.4).

## 5.2. Baseline Models

We follow [52] to employ 3D ConvNets as our baseline (B/L) model. The baseline models share the same 3D ConvNets architecture with our proposed model, except that the last layer is fully connected instead of gated through topological feature aggregation. The difference between baselines is their input: B/L1 takes the raw frames as input; B/L2 incorporates the scene layout information by directly concatenating the 6 segmentation maps to the RGB channels in each frame (we call this method ConcateMap), resulting in an input of 9 channels per frame in total. We train the baseline models using the same setting as in the proposed model, and the performance of the baselines are shown in column 2-5 in Table 1. We observe that: 1) the testing performance gap between observed and unseen scenes is large, which reveals the poor generalization of the baseline models; 2) marginal improvements are obtained by incorporating scene layout information using ConcateMap, which suggests that it is difficulty for the network to learn the human-scene interactions directly from the raw pixels and segmentation maps. In addition, we also train a B/L2 model with $6\times$ more filters per layer to evaluate whether model size is the key factor for the performance improvement. The result of this enlarged B/L2 model is shown in column 5 of Table 1. Overall, the baseline models which directly extract features jointly from the entire video suffer from overfitting, and simply enlarging the model size or directly using the segmentation maps as features does not improve their generalization.

## 5.3. Evaluation on the Proposed Method

**The path from traditional 3D ConvNets to our method.** We show the path from the baselines to our method in Table 1. In column 6-9, we report the mAP of our models on observed scene validation set and unseen scene testing set. We observe three significant performance gaps, especially on unseen scenes: 1) from B/L2 to LIVR-V1, we obtain around $20\%$ mAP improvement by applying the proposed semantic feature decomposition to extract place feature descriptions; 2) from LIVR-V1 to LIVR-V3, our model is further improved by explicitly modeling moving directions by place discretization; 3) when compared to using a fully connected layer for feature aggregation (V1 and V3), our topological method (V2 and V4) leads to another significant improvement, which shows the efficacy of feature aggregation based

[4] If there is no *porch* in a scene, the user draws a line (click to generate two endpoints) to indicate its location.

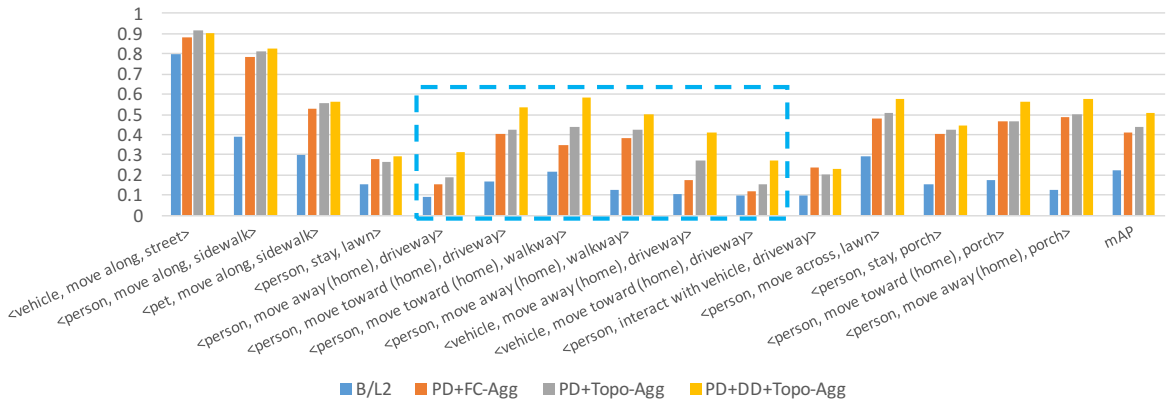



Figure 7. Per-category average precision of the baseline 3 and our methods on unseen scenes. The blue dashed box highlights actions which require modeling moving directions. We observe that the proposed place-based feature descriptions (PD), distance-based place discretization (DD) and topological feature aggregation (Topo-Agg) significantly improve the average precision on almost all action categories. FC-Agg stands for using a FC layer to aggregate place descriptions.

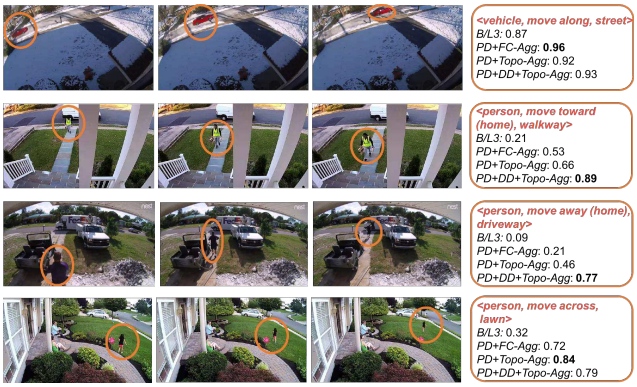



Figure 8. Qualitative examples: The predicted confidences of groundtruth actions using different methods. We use 3 frames to visualize a motion and orange ellipses to highlight moving agents.

on scene layout connectivity. We also evaluate the effect of resolutions, FPS and number of filters using our best model (LIVR-V4). Doubling the resolution ($320 \times 180$), FPS (2) and number of filters (128) only results in a slight change of the model's accuracy (columns 10-12 in Table 1). Besides our baselines, we also apply other state-of-the-art video action recognition methods (TSN [45] and ReMotENet [52]) on our dataset. LIVR outperforms them by a large margin, especially on the unseen scenes. Per-category results are shown in Fig. 7 and more discussions are included in the supplementary materials.

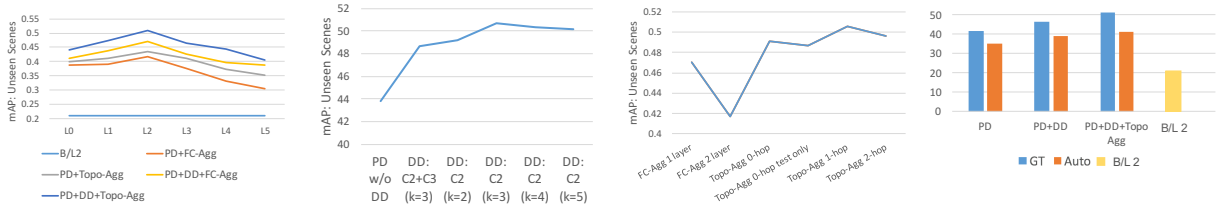
**Qualitative Results.** Some example actions are visualized using three frames in temporal order and the predicted probabilities of the groundtruth actions using different methods are reported in Fig.8. It is observed that for relatively easy actions such as <*vehicle, move along, street*>, performance is similar across approaches. However, for challenging actions, especially ones requiring modeling moving directions such as <*person, move toward (home), walkway*>, our method outperforms baselines significantly.

### 5.4. Ablation Analysis on Unseen Scenes

**Place-based Feature Description.** The hyper-parameter for PD is the level $L$, controlling when to decompose semantics in different places. Fig.9(a) and 9(c) show that the generalization capability of our model is improved when we allow the network to observe the entire video at input level, and decompose semantics at feature level (after the 2nd conv blocks). Generally, the improvements of PD are robust across different feature levels.

**Distance-based Place Discretization.** We study different strategies for determining $PL_{DT}$ and the number of parts to discretize ($k$) per place. From our observations, including the anchor place–*porch*, the six places in our dataset can be clustered into three categories with regard to the distance to camera: C1 includes only *porch*, which is usually the closest place to camera; C2 includes *lawn*, *walkway*, *driveway*, and actions occurring in those places usually require modeling the moving direction directly; C3 includes *sidewalk* and *street*, which are usually far away from the house, and actions on them are not sensitive to directions (*e.g.,* "*move along*"). We evaluate our method with two strategies to apply DD on: 1) all places belong to C2 and C3; 2) only places in C2. The results are shown in Fig.9(b). We observe that applying DD on C3 dose not help much, but if we only apply DD on places in C2, our method achieves the best performance. In terms of the number of discretized parts $k$, we evaluate $k$ from 2 to 5 and observe from Fig.9(b) that the performance is robust when $k \geqslant 3$.

**Topological Feature Aggregation.** We evaluate different $h$ values to determine the $h$-connected set and different strategies to construct and utilize the action-place mapping $\mathbf{T}$.




(a) Decomposition Level $L$ (b) Place Discretization (c) Topological Aggregation (d) Segmentation: GT v.s. Auto

Figure 9. Evaluation: (a) The effect of extracting place-based feature descriptions (PD) at different levels using different variants of our proposed model. (b) Different strategies for distance-based place discretization. (c) Different feature aggregation approaches on unseen scenes. (d) Performance of LIVR using groundtruth (GT) and automatically generated (Auto) segmentation map.

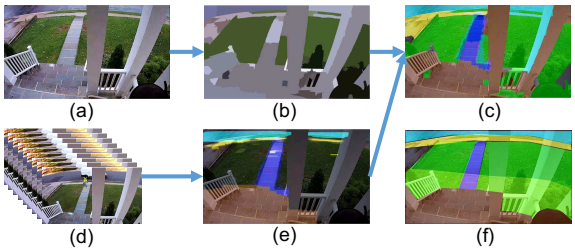



Figure 10. Process of Automatically Generating Segmentation Maps. (a) is the input camera image. (b) is the output of normalized cut method. (d) is the set of all videos captured by this camera. (e) shows the heatmaps we obtained by analyzing the patterns of moving objects from the videos. (c) is the generated segmentation map. (f) is the ground truth map.

The results are shown in Fig.9(c). We set $L = 2$, and use both PD and DD. We observe that Topo-Agg achieves its best performance when $h = 1$, *i.e.,* for an action occurring in place $P$, we aggregate features extracted from place $P$ and its directly connected places. In addition, we compare Topo-Agg to the naive fully connected inference layer (FC-Agg: 1 layer) and two fully-connected layers with 384 neurons each and a ReLU layer in between (FC-Agg: 2 layers). Unsurprisingly, we observe that the generalizability drops significantly with an extra fully-connected layer, which reflects overfitting. Our Topo-Agg outperforms both methods. We also conduct an experiment where we train a fully connected inference layer and only aggregate features based on topology at testing time ("Topo-Agg: 1-hop test only") and it shows worse performance.

**LIVR with Automatically Generated Segmentation Maps.** To evaluate the performance of LIVR using imperfect segmentation maps, we developed an algorithm to automatically generate the segmentation maps. As shown in Fig.10, we first apply normalized cut [33] on the camera images to obtain segments (Fig.10 (b))[5]. Then, to further differentiate different places with similar appearance (*e.g.,* walkway and street), we developed an algorithm to utilize the historical statistics obtained from previous videos (Fig.10 (d)) of a scene to generate heatmaps of some specific places[6] (Fig.10 (e)). Then, the two results are combined to obtain final segmentation maps (Fig.10 (c)). Our method can generate reasonably good segmentation maps when compared to the groundtruth maps obtained manually (Fig. 10 (f)). We evaluate LIVR using the imperfect maps and observe some performance degradations (around 10%), but LIVR still outperforms the baselines by a large margin (around 20%), which demonstrate the effectiveness of our method even if the segmentation maps are imperfect. Details of how we generate the segmentation maps can be found in the supplementary materials.

## 6. Conclusion

To improve the generalization of a deep network that learns from limited training scenes, we explicitly model scene layout in a network by using layout-induced video representations, which abstracts away low-level appearance variance but encodes the semantics, geometry and topology of scene layouts. An interesting future directions would be to include integrate the estimation of the semantic maps into the network architecture, which may require collecting more scenes for training.

## 7. Acknowledgement

Partial support from the Office of Naval Research under Grant N000141612713 (Visual Common Sense Reasoning for Multiagent Activity Prediction and Recognition) is acknowledged. The fruitful discussion with MD Mahmudul Hasan and Jan Neumann is highly appreciated. We thank Eddie Kessler for proofreading this paper. Special thanks go to all Comcast TPX volunteers who donated their home videos.

[5]We also tried deep learning based methods trained on semantic segmentation datasets, but they perform poorly on our camera images. Details can be found in the supplementary materials.

[6]We utilize the patterns of moving objects to differentiate places. For example, street is a place where vehicles move with limited scale changes (from the camera perspective), and walkway is a place where people with notably large scale changes walk.

## References


[1] Lamberto Ballan, Marco Bertini, Giuseppe Serra, and Alberto Del Bimbo. A data-driven approach for tag refinement and localization in web videos. *Computer Vision and Image Understanding*, 140:58–67, Nov. 2015. 3

[2] Lamberto Ballan, Francesco Castaldo, Alexandre Alahi, Francesco Palmieri, and Silvio Savarese. Knowledge transfer for scene-specific motion prediction. In *European Conference on Computer Vision (ECCV)*, 2016. 3

[3] Shyamal Buch, Victor Escorcia, Bernard Ghanem, Li Fei-Fei, and Juan Carlos Niebles. End-to-end, single-stream temporal action detection in untrimmed videos. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2017. 3

[4] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 1, 3

[5] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A. Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster R-CNN architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 3

[6] Robert Collins, Alan Lipton, Takeo Kanade, Hironobu Fujiyoshi, David Duggins, Yanghai Tsin, David Tolliver, Nobuyoshi Enomoto, and Osamu Hasegawa. A system for video surveillance and monitoring. Technical Report CMU-RI-TR-00-12, Carnegie Mellon University, Pittsburgh, PA, May 2000. 3

[7] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Trevor Darrell, and Kate Saenko. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2625–2634. IEEE Computer Society, 2015. 1, 3

[8] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 1933–1941, 2016. 1, 3

[9] Florent Fusier, Valéry Valentin, François Brémond, Monique Thonnat, Mark Borg, David Thirde, and James Ferryman. Video understanding for complex activity recognition. *Machine Vision and Applications*, 18(3):167–188, Aug 2007. 3

[10] Mingfei Gao, Ang Li, Ruichi Yu, Vlad I. Morariu, and Larry S. Davis. C-wsl: Count-guided weakly supervised localization. *arXiv preprint arXiv:1711.05282*, 2017. 3

[11] Mingfei Gao, Ruichi Yu, Ang Li, Vlad I. Morariu, and Larry S. Davis. Dynamic zoom-in network for fast object detection in large images. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[12] Ross Girshick. Fast R-CNN. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2015. 3

[13] Chunhui Gu, Chen Sun, David A. Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[14] Ismail Haritaoglu, David Harwood, and Larry S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:809–830, 2000. 3

[15] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K. Roy-Chowdhury, and Larry S. Davis. Learning temporal regularity in video sequences. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3

[16] James Hays and Alexei A. Efros. Scene completion using millions of photographs. *ACM Trans. Graph.*, 26(3), July 2007. 3

[17] Haroon Idrees, Amir Roshan Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The THUMOS challenge on action recognition for videos "in the wild". *CoRR*, abs/1604.06182, 2016. 3

[18] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1):221–231, Jan. 2013. 1, 3

[19] 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. *CoRR*, abs/1705.06950, 2017. 1, 2

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 6

[21] Kris M. Kitani, Brian D. Ziebart, James Andrew Bagnell, and Martial Hebert. Activity forecasting. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part IV*, ECCV'12, pages 201–214, Berlin, Heidelberg, 2012. Springer-Verlag. 3

[22] H. Kuhne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *IEEE International Conference on Computer Vision (ICCV)*, 2011. 1, 2

[23] Ang Li, Jin Sun, Joe Yue-Hei Ng, Ruichi Yu, Vlad I. Morariu, and Larry S. Davis. Generating holistic 3d scene abstractions for text-based image retrieval. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[24] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, pages 988–996, New York, NY, USA, 2017. ACM. 3

[25] Ce Liu, Jenny Yuen, and Antonio Torralba. *Nonparametric Scene Parsing via Label Transfer*, pages 207–236. Springer International Publishing, Cham, 2016. 3

[26] Chih-Yao Ma, Min-Hung Chen, Zsolt Kira, and Ghassan AlRegib. TS-LSTM and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition. *CoRR*, abs/1703.10667, 2017. 1, 3

[27] Chih-Yao Ma, Asim Kadav, Iain Melvin, Zsolt Kira, Ghassan AlRegib, and Hans Peter Graf. Attend and interact: Higher-order object interactions for video understanding. In *The*

*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[28] Joe Yue-Hei Ng, Matthew J. Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, pages 4694–4702. IEEE Computer Society, 2015. 1, 3

[29] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. *CoRR*, abs/1808.07962, 2018. 3

[30] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, 2017. 3

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015. 3

[32] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, pages 1–28, 2015. 3

[33] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, Aug. 2000. 8

[34] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *CVPR*, 2017. 3

[35] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *ECCV*, 2016. 3

[36] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'14, pages 568–576, Cambridge, MA, USA, 2014. MIT Press. 1, 3

[37] 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. 1, 2

[38] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XI*, pages 335–351, 2018. 3

[39] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. *CoRR*, abs/1807.10982, 2018. 3

[40] Graham W. Taylor, Rob Fergus, Yann LeCun, and Christoph Bregler. Convolutional learning of spatio-temporal features. In *Proceedings of the 11th European Conference on Computer Vision: Part VI*, ECCV'10, pages 140–153, Berlin, Heidelberg, 2010. Springer-Verlag. 1, 3

[41] Joseph Tighe and Svetlana Lazebnik. Superparsing: Scalable nonparametric image parsing with superpixels. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, pages 352–365, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. 3

[42] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 1, 3

[43] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6450–6459, 2018. 3

[44] Geoffrey Vaquette, Astrid Orcesi, Laurent Lucat, and Catherine Achard. The daily home life activity dataset: A high semantic activity dataset for online recognition. *2017 12th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2017)*, pages 497–504, 2017. 3

[45] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. *CoRR*, abs/1608.00859, 2016. 6, 7

[46] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Val Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 1, 3

[47] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *CVPR*, 2018. 3

[48] C. Xu and J. J. Corso. Actor-action semantic segmentation with grouping-process models. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1

[49] Dan Xu, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe. Learning deep representations of appearance and motion for anomalous event detection. In Mark W. Jones Xianghua Xie and Gary K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 8.1–8.12. BMVA Press, September 2015. 3

[50] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017. 3

[51] Xun Xu, Timothy M Hospedales, and Shaogang Gong. Discovery of shared semantic spaces for multiscene video query and summarization. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(6):1353–1367, 2017. 3

[52] Ruichi Yu, Hongcheng Wang, and Larry Davis. Remotenet: Efficient relevant motion event detection for large-scale home surveillance videos. In *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2018. 3, 6, 7

[53] Jenny Yuen and Antonio Torralba. A data-driven approach for event prediction. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, pages 707–720, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. 3

[54] Da Zhang, Xiyang Dai, and Yuan-Fang Wang. Dynamic temporal pyramid network: A closer look at multi-scale modeling for activity detection. *CoRR*, abs/1808.02536, 2018. 3

[55] Yue Sheng Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Dahua Lin, and Xiaoou Tang. Temporal action detection with structured segment networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2933–2942, 2017. 3

[56] Shifu Zhou, Wei Shen, Dan Zeng, Mei Fang, Yuanwang Wei, and Zhijiang Zhang. Spatial-temporal convolutional neural networks for anomaly detection and localization in crowded scenes. *Image Commun.*, 47(C):358–368, Sept. 2016. 3