Action Recognition by Hierarchical Mid-level Action Elements

Tian Lan*, Yuke Zhu*, Amir Roshan Zamir and Silvio Savarese
Stanford University

Abstract

Realistic videos of human actions exhibit rich spatiotemporal structures at multiple levels of granularity: an action can always be decomposed into multiple finer-grained elements in both space and time. To capture this intuition, we propose to represent videos by a hierarchy of mid-level action elements (MAEs), where each MAE corresponds to an action-related spatiotemporal segment in the video. We introduce an unsupervised method to generate this representation from videos. Our method is capable of distinguishing action-related segments from background segments and representing actions at multiple spatiotemporal resolutions. Given a set of spatiotemporal segments generated from the training data, we introduce a discriminative clustering algorithm that automatically discovers MAEs at multiple levels of granularity. We develop a structured model that captures a rich set of spatial, temporal and hierarchical relations among the segments, where the action label and multiple levels of MAE labels are jointly inferred. The proposed model achieves state-of-the-art performance in multiple action recognition benchmarks. Moreover, we demonstrate the effectiveness of our model in real-world applications such as action recognition in large-scale untrimmed videos and action parsing.

1. Introduction

In this paper we address the problem of learning models of human actions and using these models for recognizing and parsing human actions from videos. This is a very challenging problem. Most of the human actions are complex spatiotemporal hierarchical processes. Consider, for instance, the action in Fig. 1. It is composed of a collection of spatiotemporal processes ranging from the entire action sequence, "taking food from fridge" to simple elementary actions such as "stretching arm" or "grasping a tomato". Each of these actions is often characterized by a complex distribution of motion segments (e.g. *open* and *close*), objects (e.g. *fridge* and *food*), body parts (e.g. *arm*) along with their interactions (e.g. *grasp a tomato*). Thus, in order to achieve a full understanding of the action that takes place in a scene, one must recognize and parse this complex structure of *mid-level action elements* (MAEs) at different levels of semantic and spatiotemporal resolution.

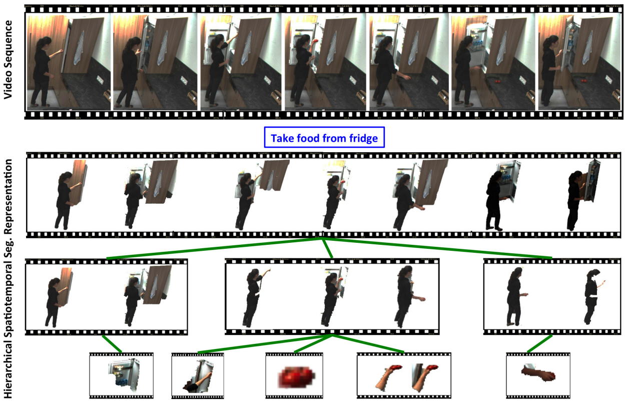


Figure 1. **A representation of *hierarchical spatiotemporal segments* for action.** Our method automatically discovers representative and discriminative mid-level action elements for a given action class. These elements are encoded in the spatiotemporal segments which usually cover different aspects of an action at different levels of granularity, ranging from an entire action sequence, which comprises the actor along with the objects the actor interacts with (the first row of the hierarchy), to the action elements such as fine-grained body part movements and objects (the last row).

Most of the existing methods cannot do this. A large body of work focuses on associating the entire video clip with a single class label from a pre-defined set of action categories (e.g., "take food from fridge" versus "cook food") (Fig. 1) [22, 21, 41] – essentially, a video classification problem. Methods such as [12, 26] do propose methodologies for temporally segmenting or parsing the action (e.g., "take food from fridge") into a sequence of sub-action labels (e.g., open fridge, grasp food, close fridge) but cannot organize these sub-actions into hierarchical structures of MAEs such as the one in Fig. 1. Critically, most of these methods assume that the fine-grained action labels or their temporal structures are pre-defined or hand-specified by an expert as opposed to being automatically inferred from the videos in a data-driven fashion. This assumption prevents such methods from scaling up to a large number of complex actions. Finally, a portion of previous research focuses

*indicates equal contribution

on modeling an action by just capturing the spatiotemporal characteristics of the actor [9, 37, 20, 33] whereby neither the objects nor the background the actor interacts with are used to better contextualize the classification process. Other methods [16, 41, 17, 36] do propose a holistic representation for activities which inherently captures some degree of background context in the video, but are unable to spatially localize or segment the actors or relevant objects.

In this work, we propose a model that is capable of modeling complex actions as a collection of *mid-level action elements* (MAEs) that are organized in a hierarchical way. Compared to previous approaches, our framework enables: 1) **Multi-resolution reasoning** – videos can be decomposed into a hierarchical structure of spatiotemporal MAEs at multiple scales; 2) **Parsing capabilities** – actions can be described (parsed) as a rich collection of spatiotemporal MAEs that capture different characteristics of the action ranging from small body motions, objects to large pieces of volumes containing person-object interactions. These MAEs can be spatially and temporally localized in the video sequence; 3) **Data-driven learning** – the hierarchical structure of MAEs as well as the their labels do not have to be manually specified, but are learnt and automatically discovered using a newly proposed weakly-supervised agglomerative clustering procedure. Note that some of the MAEs might have clear semantic meanings (see Fig. 1), while others might correspond to arbitrary but discriminative spatiotemporal segments. In fact, these MAEs are learnt so as to establish correspondences between videos from the same action class while maximizing their discriminative power for different action classes. Our model has achieved state-of-the-art results on multiple action recognition benchmarks and is capable of recognizing actions from large-scale untrimmed video sequences.

2. Related Work

The literature on human action recognition is immense. we refer the readers to a recent survey [2]. In the following, we only review the closely related work to our work.

Space-time segment representation: Representing actions as 2D+t tubes is a common strategy for action recognition [3, 5, 24]. Recently, there are works that use hierarchical spatiotemporal segments to capture the multi-scale characteristics of actions [5, 24]. Our representation differs in that we can discriminatively discover the *mid-level action elements* (MAEs) from a pool of region proposals.

Temporal action localization: While most action recognition approaches focus on classifying trimmed video clips [21, 9, 36], there are works that attempt to localize action instances from long video sequences [8, 12, 22, 4, 31]. In [31], a grammar model is developed for localizing action and (latent) sub-action instances in the video. Our work considers a more detailed parsing at both space and time, and at different semantic resolutions.

Hierarchical structure: Hierarchical structured models are popular in action recognition due to its capability in capturing the multi-level granularity of human actions [39, 19, 18, 33]. We follow a similar spirit by representing an action as a hierarchy of MAEs. However, most previous works focus on classifying single-action video clips where they treat these MAEs as latent variables. Our method localizes MAEs at both spatial and temporal extent.

Data-driven action primitives: Action primitives are discriminative parts that capture the appearance and motion variations of the action [26, 42, 13, 18]. Previous representations of action primitives such as interest points [42], spatiotemporal patches [13] and video snippets [26] typically lack multiple levels of granularity and structures. In this work, we represent action primitives as MAEs, which are capable of capturing different aspects of actions ranging from the fine-grained body part segments to the large chunks of human-object interactions. A rich set of spatial, temporal and hierarchical relations between the MAEs are also encoded. Both the MAE labels and the structures of MAEs are discovered in a data-driven manner.

Before diving into details, we first give an overview of our method. 1) *Hierarchical spatiotemporal segmentation.* Given a video, we first develop an algorithm to automatically parse the video into a hierarchy of spatiotemporal segments (see Fig. 2). We run this algorithm for each video independently, and in this way, each video is represented as a spatiotemporal segmentation tree (Section 3). 2) *Learning.* Given a set of spatiotemporal segmentation trees (one tree per video) in training, we propose a graphical model that captures the hierarchical dependencies of MAE labels at different levels of granularity. We consider a weakly supervised setting, where only the action label is provided for each training video, while the MAE labels are discriminatively discovered by clustering the spatiotemporal segments. The structure of the model is defined by the spatiotemporal segmentation tree where inference can be carried out efficiently (Section 4). 3) *Recognition and parsing.* A new video is represented by the spatiotemporal segmentation tree. We run our learned models on the tree for recognizing the actions and parsing the videos into MAE labels at different spatial, temporal and semantic resolutions.

3. Action Proposals: Hierarchical Spatiotemporal Segments

In this section, we describe our method for generating a hierarchy of action-related spatiotemporal segments from a video. Our method is unsupervised, i.e. during training, the spatial locations of the persons and objects are not annotated. Thus, it is important that our method can automatically extract the action-related spatiotemporal segments such as actors, body parts and objects from the video.

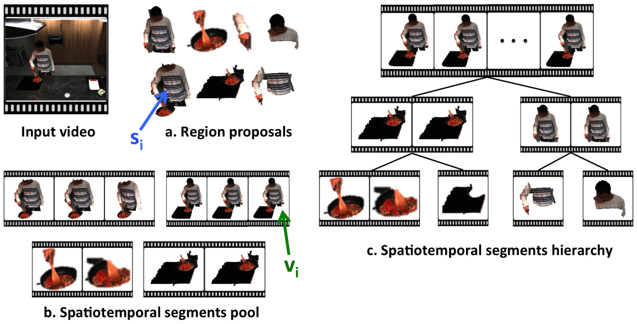

Input video
a. Region proposals
s_i
v_i
b. Spatiotemporal segments pool
c. Spatiotemporal segments hierarchy

Figure 2. **Constructing the spatiotemporal segment hierarchy.** (a) Given a video, we first generate action-related region proposals for each frame. (b) Then, we cluster these proposals to produce a pool of spatiotemporal segments. (c) The last step is to agglomeratively cluster the spatiotemporal segments into a hierarchy.

An overview of the method is shown in Fig. 2.

Our method for generating action proposals includes three major steps. **A. Generating action-related spatial segments.** We initially generate a diverse set of region proposals using the method of [10]. This method works on a single frame of video, and returns a large number of segmentation masks that are likely to contain objects or object parts. We then score each region proposal using both appearance and motion cues, and we look for regions that have generic object-like appearance and distinct motion patterns relative to their surroundings. We further prune the background region proposals by training an SVM using the top scored region proposals as positive examples along with patches randomly sampled from the background as negative examples. The region proposals with scores above a threshold (-1) are considered as action-related spatial segments. **B. Obtaining the spatiotemporal segment pool.** Given the action-related spatial segments for each frame, we seek to compute "tracklets" of these segments over time to construct the spatiotemporal segments. We perform spectral clustering based on the color, shape and space-time distance between pairs of spatial segments to produce a pool of spatiotemporal segments. In order to maintain the purity of each spatiotemporal segment, we set the number of clusters to a reasonably large number. The pool of spatiotemporal segments correspond to the action proposals at the finest scale (bottom of Fig. 1). **C. Constructing the hierarchy.** Starting from the initial set of fine-grained spatiotemporal segments, we agglomeratively cluster the most similar spatiotemporal segments into super-spatiotemporal segments until a single super-spatiotemporal segment is left. In this way, we produce a hierarchy of spatiotemporal segments for each video. The nodes in the hierarchy are action proposals at different levels of granularity. Due to space constraints, we refer the details of the method to Sec. S1 in the supplementary material [1].

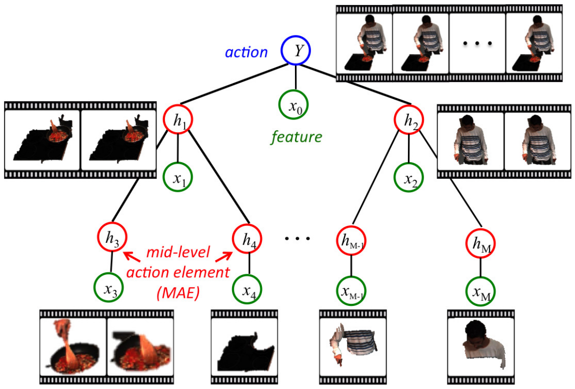


Figure 3. **Graphical illustration of the model.** In this example, we adopt the spatiotemporal hierarchy in Fig. 2 (c). The MAE labels are the red circles. The green circles are the features of each spatiotemporal segment, and the the blue circle is the action label.

4. Hierarchical Models for Action Recognition and Parsing

So far we have explained how to parse a video into a tree of spatiotemporal segments. We run this algorithm for each video independently, and in this way, each video is represented as a tree of spatiotemporal segments. Our goal is to assign each of these segments to a label so as to form a *mid-level action element* (MAE). We consider a weakly supervised setting. During training, only the action label is provided for each video. We discover the MAE labels in an unsupervised way by introducing a discriminative clustering algorithm that assigns each spatiotemporal segment to an MAE label (Section 4.1). In Section 4.2, we introduce our models for action recognition and parsing, which are able to capture the hierarchical dependencies of the MAE labels at different levels of granularity.

We start by describing the notations. Given a video V_n, we first parse it into a hierarchy of spatiotemporal segments, denoted by $V_n = \{v_i : i = 1, \ldots, M_n\}$ following the procedure introduced in Section 3. We extract features X_n from these spatiotemporal segments in the form of $X_n = \{x_i : i = 0, 1, \ldots, M_n\}$, where x_0 is the root feature vector, computed by aggregating the feature descriptors of all spatiotemporal segments in the video, and x_i $(i = 1, \ldots, M_n)$ is the feature vector extracted from the spatiotemporal segment v_i (see Fig. 3).

During training, each video V_n is annotated with an action label $Y_n \in \mathcal{Y}$ and $\mathcal{Y}$ is the set of all possible action labels. We denote the MAE labels in the video as $H_n = \{h_i : i = 1, \ldots, M_n\}$, where $h_i \in \mathcal{H}$ is the MAE label of the spatiotemporal segment v_i and $\mathcal{H}$ is the set of all possible MAE labels (see Fig. 3). For each training video, the MAE labels H_n are automatically assigned to clusters of spatiotemporal segments by our discriminative clustering algorithm (Section 4.1). The hierarchical structure above can be compactly described using the notation

$\mathcal{G}_n = (\mathcal{V}_n, \mathcal{E}_n)$, where a vertex $v_i \in \mathcal{V}_n$ denotes a spatiotemporal segment, and an edge $(v_i, v_j) \in \mathcal{E}_n$ represents the interaction between a pair of spatiotemporal segments. In the next section, we describe how to automatically assign MAE labels to clusters of spatiotemporal segments.

4.1. Discovering Mid-level Action Elements (MAEs)

Given a set of training videos with action labels, our goal is to discover the MAE labels $\mathcal{H}$ by assigning the clusters of *spatiotemporal segments* (Section 3) to the corresponding cluster indices. Consider the example in Fig. 1. The input video is annotated with an action label "take food from fridge" in training, and the MAEs should describe the action at different resolutions ranging from the fine-grained action and object segments (e.g. fridge, tomato, grab) to the higher-level human-object interactions (e.g. open fridge, close fridge). These MAE labels are not provided in training, but are automatically discovered by a discriminative clustering algorithm on a per-category basis. That means the MAEs are discovered by clustering the spatiotemporal segments from all the training videos within each action class. The MAEs should satisfy two key requirements: 1) *inclusivity* - MAEs should cover all, or at least most, variations in the appearance and motion of the spatiotemporal segments in an action class; 2) *discriminability* - MAEs should be useful to distinguish an action class from others.

Inspired by the recent success of discriminative clustering in generating mid-level concepts [38], we develop a two-step discriminative clustering algorithm to discover the MAEs. 1) *Initialization:* we perform an initial clustering to partition the spatiotemporal segments into a large number of homogeneous clusters, where each cluster contains segments that are highly similar in appearance and shape. 2) *Discriminative algorithm:* a discriminative classifier is trained for each cluster independently. Based on the discriminatively-learned similarity, the visually consistent clusters will then be merged into mid-level visual patterns (i.e. MAEs). The discriminative step will make sure that each MAE pattern is different enough from the rest. The two-step algorithm is explained in details below.

Initialization. We run standard spectral clustering on the feature space of the spatiotemporal segments to obtain the initial clusters. We define a similarity between every pair of spatiotemporal segments v_i and v_j extracted from all of the training videos of the same class: $K(v_i, v_j) = \exp(-d_{bow}(v_i, v_j) - d_{spatial}(v_i, v_j))$, where d_{bow} is the histogram intersection distance on the BoW representations of the dense trajectory features [41]; and $d_{spatial}$ denotes the Euclidean distance between the averaged bounding boxes of spatiotemporal segments v_i and v_j in terms of four cues: x-y locations, height and width. In order to keep the purity of each cluster, we set the number of clusters quite high, producing around 50 clusters per action. We remove clusters with less than 5 spatiotemporal segments.

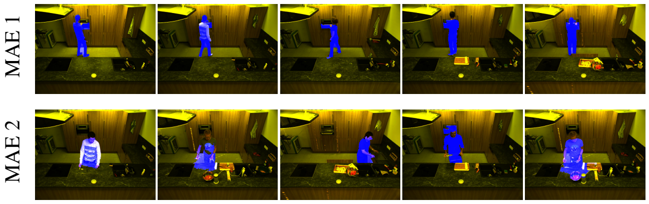


Figure 4. **Visualization of Mid-level Action Elements (MAEs).** The figure shows two clusters (i.e. two MAEs) from the action category "take out from oven". Each image shows the first frame of a spatiotemporal segment and we only visualize five examples in each MAE. The two clusters capture two different temporal stages of "take out from oven". More visualizations are available in the supplementary video [1].

Discriminative Algorithm. Given an initial set of clusters, we train a linear SVM classifier for each cluster on the BoW feature space. We use all spatiotemporal segments in the cluster as positive examples. Negative examples are spatiotemporal segments from other action classes. For each cluster, we run the classifier on all other clusters in the same action class. We consider the top K scoring detections of each classifier. We define the affinity between the initial clusters c_i and c_j as the frequency that classifier i and j fire on same cluster.

For each action class, we compute the pairwise affinities between all initial clusters, to obtain the affinity matrix. Next we perform spectral clustering on the affinity matrix of each action independently to produce the MAE labels. In this way, the spatiotemporal segments in the training set are automatically grouped into clusters in a discriminative way, where the index of each cluster corresponds to an MAE label $h_i \in \mathcal{H}$, where $\mathcal{H}$ denotes the set of all possible MAE labels. We visualize the example MAE clusters in Fig. 4.

4.2. Model Formulation

For each video, we have a different tree structure $\mathcal{G}_n$ obtained from the spatiotemporal segmentation algorithm (Section 3). Our goal is to jointly model the compatibility between the input feature vectors X_n, the action label Y_n and MAE labels H_n, as well as the dependencies between pairs of MAE labels. We achieve this by using the following potential function:

$$S_{\mathcal{V}_n}(X_n, Y_n, H_n) = \sum_{i \in \mathcal{V}_n} \alpha_{h_i}^\top x_i + \sum_{i \in \mathcal{V}_n} b_{Y_n, h_i} + \sum_{(i,j) \in \mathcal{E}_n} b'_{h_i, h_j} + \sum_{(i,j) \in \mathcal{E}} \beta_{h_i, h_j}^\top d_{ij} + \eta_{Y_n}^\top x_0 \quad (1)$$

MAE Model $\alpha_{h_i}^\top x_i$: This potential captures the compatibility between the MAE h_i and the feature vector x_i of the i-th spatiotemporal segment. In our implementation, rather than using the raw feature [41], we use the output of the MAE classifier on the feature vector of spatiotemporal segment i. In order to learn biases between different MAEs, we append a constant 1 to make x_i 2-dimensional.

Co-occurrence Model $b_{Y_n,h_i}, b'_{h_i,h_j}$: This potential captures the co-occurrence constraints between pairs of MAE labels. Since the MAEs are discovered on a per-action basis, thus we restrict the co-occurrence model to allow for only action-consistent types: $b_{Y_n,h_i} = 0$ if the MAE h_j is generated from the action class Y_n, and $-\infty$ otherwise. Similarly, $b'_{h_i,h_j} = 0$ if the pair of MAEs h_i and h_j are generated from the same action class, and $-\infty$ otherwise.

Spatial-Temporal Model $\beta^{\top}_{h_i,h_j} d_{ij}$: This potential captures the spatiotemporal relations between a pair of MAEs h_i and h_j. In our experiments, we explore a simplified version of the spatiotemporal model with a reduced set of structures: $\beta^{\top}_{h_i,h_j} d_{ij} = \beta^{\top}_{h_i} bin_s(i) + \beta^{\top}_{h_i} bin_t(i,j)$. The simplification states that the relative spatial and temporal relation of a spatiotemporal segment i with respect to its parent j is dependent on the segment type h_i, but not its parent type h_j. To compute the spatial feature bin_s, we divide a video frame into 5×5 cells, and $bin_s(i) = 1$ if the i-th spatiotemporal segment falls into the m-th cell, otherwise 0. $bin_t(i,j)$ is a temporal feature that bins the relative temporal location of spatiotemporal segment i and j into one of three canonical relations including *before*, *co-occur*, and *after*. Hence $bin_t(i,j)$ is a sparse vector of all zeros with a single one for the bin occupied by the temporal relation between i and j.

Root Model $\eta^{\top}_{Y_n} x_0$: This potential function captures the compatibility between the global feature x_0 of the video V_n and the action class Y_n. In our experiment, the global feature x_0 is computed as the aggregation of feature descriptors of all spatiotemporal segments in the video.

4.3. Inference

The goal of inference is to predict the hierarchical labeling for a video, including the action label for the whole video as well as the MAE labels for spatiotemporal segments at multiple scales. For a video V_n, our inference corresponds to solving the following optimization problem: $(Y_n^*, H_n^*) = \arg\max_{Y_n,H_n} S_{V_n}(X_n, Y_n, H_n)$. For the video V_n, we jointly infer the action label Y_n of the video and the MAE labels H_n of the spatiotemporal segments. The inference on the tree structure is exact; and we solve it using belief propagation. We emphasize that our inference returns a parsing of videos including the action label and the MAE labels at multiple levels of granularity.

4.4. Learning

Given a collection of training examples in the form of $\{X_n, H_n, Y_n\}$, we adopt a structured SVM formulation to learn the model parameters w. In the following, we develop two learning frameworks for action recognition and parsing respectively.

Action Recognition. We consider a weakly supervised setting. For a training video V_n, only the action label Y_n is provided. The MAE labels H_n are automatically discovered using our discriminative clustering algorithm. We formulate it as follows:

$$\begin{aligned} &\min_{w,\xi\geq 0} \frac{1}{2}||w||^2 + C\sum_n \xi_n \\ &S_{V_n}(X^n, H^n, Y^n) - S_{V_n}(X^n, H^*, Y^*) \\ &\geq \Delta_{0/1}\ (Y^n, Y^*) - \xi_n, \forall n, \end{aligned} \tag{2}$$

where the loss function $\Delta_{0/1}\ (Y^n, Y^*)$ is a standard 0-1 loss that measures the difference between the ground-truth action label Y^n and the predicted action Y^* for the n-th video. We use the bundle optimization solver in [7] to solve the learning problem.

Action Parsing. In the real world, a video sequence is usually not bounded for a single action, but may contain multiple actions at different levels of granularity: some actions occur in a sequential order; some actions could be composed of finer-grained MAEs. See Fig. 7 for examples.

The proposed model can naturally be extended for action parsing. Similar to our action recognition framework, the first step of action parsing is to construct the spatiotemporal segment hierarchy for an input video sequence V_n, as shown in Fig. 2. The only difference is that the input video is not a short video clip, but a long video sequence composed of multiple action and MAE instances. In training, we first associate each automatically discovered spatiotemporal segment with a ground truth action (or MAE) label. If the spatiotemporal segment contains more than one ground truth label, we choose the label with the maximum temporal overlap. We use Z_n to denote the ground truth action and MAE labels associated with the video V_n: $Z_n = \{z_i : i = 1, \ldots, M_n\}$, where $z_i \in \mathcal{Z}$ is the ground truth action (or MAE) label of the spatiotemporal segment v_i, and M_n is the total number of spatiotemporal segments discovered from the video. The goal of training is to learn a model that can parse the input video into a label hierarchy similar to the ground truth annotation Z_n. We formulate it as follows:

$$\begin{aligned} &\min_{w,\xi\geq 0} \frac{1}{2}||w||^2 + C\sum_n \xi_n \\ &S_{V_n}(X^n, Z^n) - S_{V_n}(X^n, Z^*) \geq \Delta\ (Z^n, Z^*) - \xi_n, \forall n, \end{aligned} \tag{3}$$

where $\Delta\ (Z^n, Z^*)$ is a loss function for action parsing, which we define as: $\Delta\ (Z^n, Z^*) = \frac{1}{M_n}\sum_{i\in\mathcal{V}_n} \Delta_{0/1}\ (z_i^n, z_i^*)$, where Z^n is the ground truth label hierarchy, Z^* is the predicted label hierarchy and M_n is the total number of spatiotemporal segments.

Note that the learning framework of action parsing is similar to Eq. (2), and the only difference lies in the loss function: we penalize incorrect predictions for every node of the spatiotemporal segments hierarchy.

5. Experiments

We conduct experiments on both action recognition and parsing. We first describe the datasets and experimental settings. We then present our results and compare with the state-of-the-art results on these datasets.

5.1. Experimental Settings and Baselines

We validate our methods on four challenging benchmark datasets, ranging from fine-grained actions (MPI Cooking), realistic actions in sports (UCF Sports) and movies (Hollywood2) to untrimmed action videos (THUMOS challenge). In the following, we briefly describe the datasets, experimental settings and baselines.

MPI Cooking dataset [35] is a large-scale dataset of 65 fine-grained actions in cooking. It contains in total 44 video sequences (or equally 5609 video clips, and $881,755$ frames), continuously recorded in kitchen. The dataset is very challenging in terms of distinguishing between actions of small inter-class variations, e.g. cut slices and cut dice. We split the dataset by taking one third of the videos to form the test set and the rest of the videos are used for training.

UCF-Sports dataset [34] consists of 150 video clips extracted from sports broadcasts. Compared to MPI Cooking, the scale of UCF-Sports is small and the durations of the video clips it contains are short. However, the dataset poses many challenges due to large intra-class variations and camera motion. For evaluation, we apply the same train-test split as recommended by the authors of [20].

Hollywood2 dataset [25] is composed of 1,707 video clips (823 for training and 884 for testing) with 12 classes of human actions. These clips are collected from 69 Hollywood movies, divided into 33 training movies and 36 testing movies. In these clips, actions are performed in realistic settings with camera motion and great variations.

THUMOS challenge 2014 [15] contains over 254 hours of temporally untrimmed videos and 25 million frames. We follow the settings of the action detection challenge. We use 200 untrimmed videos for training and 211 untrimmed videos for testing. These videos contain 20 action classes and are a subset of the entire THUMOS dataset. We consider a weakly supervised setting: in training, each untrimmed video is only labeled with the action class that the video contains, neither spatial nor temporal annotations are provided. Our goal is to evaluate the ability of our model in automatically extracting useful mid-level action elements (MAEs) and structures from large-scale untrimmed data.

Baselines. In order to comprehensively evaluate the performance of our method, we use the following baseline methods. 1) *DTF:* the first baseline is the dense trajectory method [41], which has produced the state-of-the-art performance in multiple action recognition benchmarks. 2) *IDTF:* the second baseline is the improved dense trajectory feature proposed in [41], which uses fisher vectors (FV) [30] to encode the dense trajectory features. FV encoding [41, 27] has been shown an improved performance over traditional Bag-of-Features encoding. 3) *root model:* the third baseline is equivalent to our model without the hierarchical structure, which only uses the IDTF features that fall into the spatiotemporal segments discovered by our method, while ignoring those in the background. 4) *sliding window:* the fourth baseline runs sliding windows of different lengths and step sizes on an input video sequence, and performs non-maximum suppression to find the correct intervals of an action. This baseline is only applied to action recognition of untrimmed videos and action parsing.

MPI Cooking	Per-Class
DTF [35, 41]	38.5
root model (ours)	43.2
full model (ours)	**48.4**

Table 1. Comparison of action recognition accuracies of different methods on the MPI Cooking dataset.

UCF-Sports	Per-Class	Hollywood2	mAP
Lan et al. [20]	73.1	Gaidon et al. [11]	54.4
Tian et al. [40]	75.2	Oneata et al. [28]	62.4
Raptis et al. [32]	79.4	Jain et al. [13]	62.5
Ma et al. [24]	81.7	Wang et al. [41]	64.3
IDTF [41]	79.2	IDTF [41]	63.0
root model (ours)	80.8	root model (ours)	64.9
full model (ours)	**83.6**	full model (ours)	**66.3**

Table 2. Comparison of our results to the state-of-the-art methods on UCF-Sports and Hollywood2 datasets. Among all of the methods, [32], [24], [11] and our full model use hierarchical structures.

THUMOS (untrimmed)	mAP
IDTF [41]	63.0
sliding window	63.8
INRIA (temporally supervised) [29]	**66.3**
full model (ours)	65.4

Table 3. Comparison of action recognition accuracies of different methods on the THUMOS challenge (untrimmed videos).

5.2. Experimental Results

We summarize the action recognition results on multiple benchmark datasets in Table 1, 2 and 3 respectively.

Action recognition. Most existing action recognition benchmarks are composed of video clips that have been trimmed according to the action of interest. On all three benchmarks (i.e. UCF-Sports, Hollywood2 and MPI), our full model with rich hierarchical structures significantly outperforms our own baseline *root model* (i.e. our model without hierarchical structures), which only considers the dense trajectories extracted from the spatiotemporal segments discovered by our method. We can also observe that the root model consistently improves dense trajectories [41] on all three datasets. This demonstrates that our automatically discovered MAEs fire on the action-related regions and thus remove the irrelevant background trajectories.

We also compare our method with the most recent results reported in the literature for UCF-Sports and Hollywood2. On UCF-Sports, all presented results follow the same train-test split [20]. The baseline IDTF [24] is among the top performance. Ma et al. [24] reported 81.7% by using a bag of hierarchical space-time segments representation. We further improve their results by around 2%. On Hollywood2, our method also achieves state-of-the-art performance. The previous best result is from [41]. We improve it further by 2%. Compared to the previous methods, our method is weakly supervised and does not require expensive bounding box annotations in training (e.g. [20, 40, 32]) or human detection as input (e.g. [41]).

On THUMOS challenge that is composed of realistic untrimmed videos (Table 3), our method outperforms both IDTF and the sliding window baseline. Given the scale of the dataset, we skip the time-consuming spatial region proposals and represent action as a hierarchy of temporal segments, i.e. each frame is regarded as a "spatial segment". Our method automatically identifies the temporal segments that are both representative and discriminative for each action class without any temporal annotation of actions in training. We also compare our methods with the best submission (INRIA [29]) of the temporal action localization challenge in THUMOS 2014. INRIA [29] uses a mixture of IDTF [41], SIFT [23], color features [6] and the CNN features [14]. Also, their model [29] is *temporally supervised*, which uses temporal annotations (the start and end frames of actions in untrimmed videos) and additional background videos in training. Our method achieves a competitive performance (within 1%) using only IDTF [41] and doesn't require any temporal supervision. We provide the average precisions (AP) of all the 20 action classes in Fig. 5. Our method outperforms [29] in 10 out of the 20 classes, especially *Diving* and *CleanAndJerk*, which contain rich structures and significant intra-class variations.

Action parsing. Given a video sequence that contains multiple action and MAE instances, our goal is to localize each one of them. Thus during training, we assume that all of the action and MAE labels as well as their temporal extent are provided. This is different from action recognition where all of the MAEs are unsupervised. We evaluate the ability of our method to perform action parsing by measuring the accuracy in temporally localizing all of the action and MAE instances. An action (or MAE) segment is considered as true positive if it overlaps with the ground truth segment beyond a pre-defined threshold. We evaluate the mean Average Precision (mAP) with the overlap threshold varying from 0.1 to 0.5.

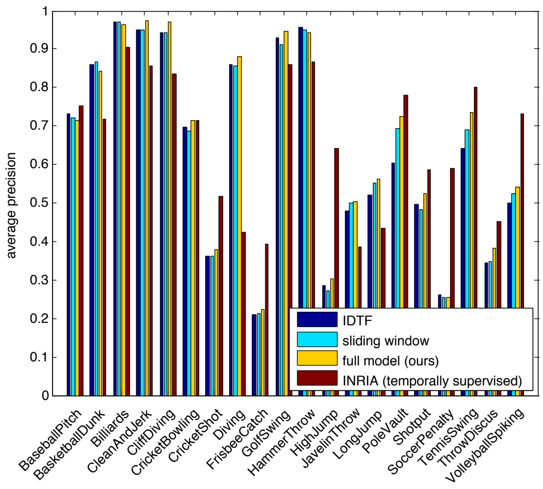


Figure 5. Average precisions of the 20 action classes of untrimmed videos from the temporal localization challenge in THUMOS.

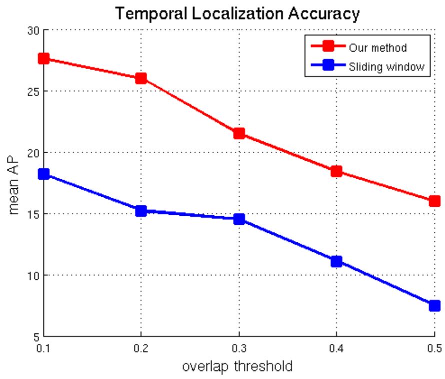
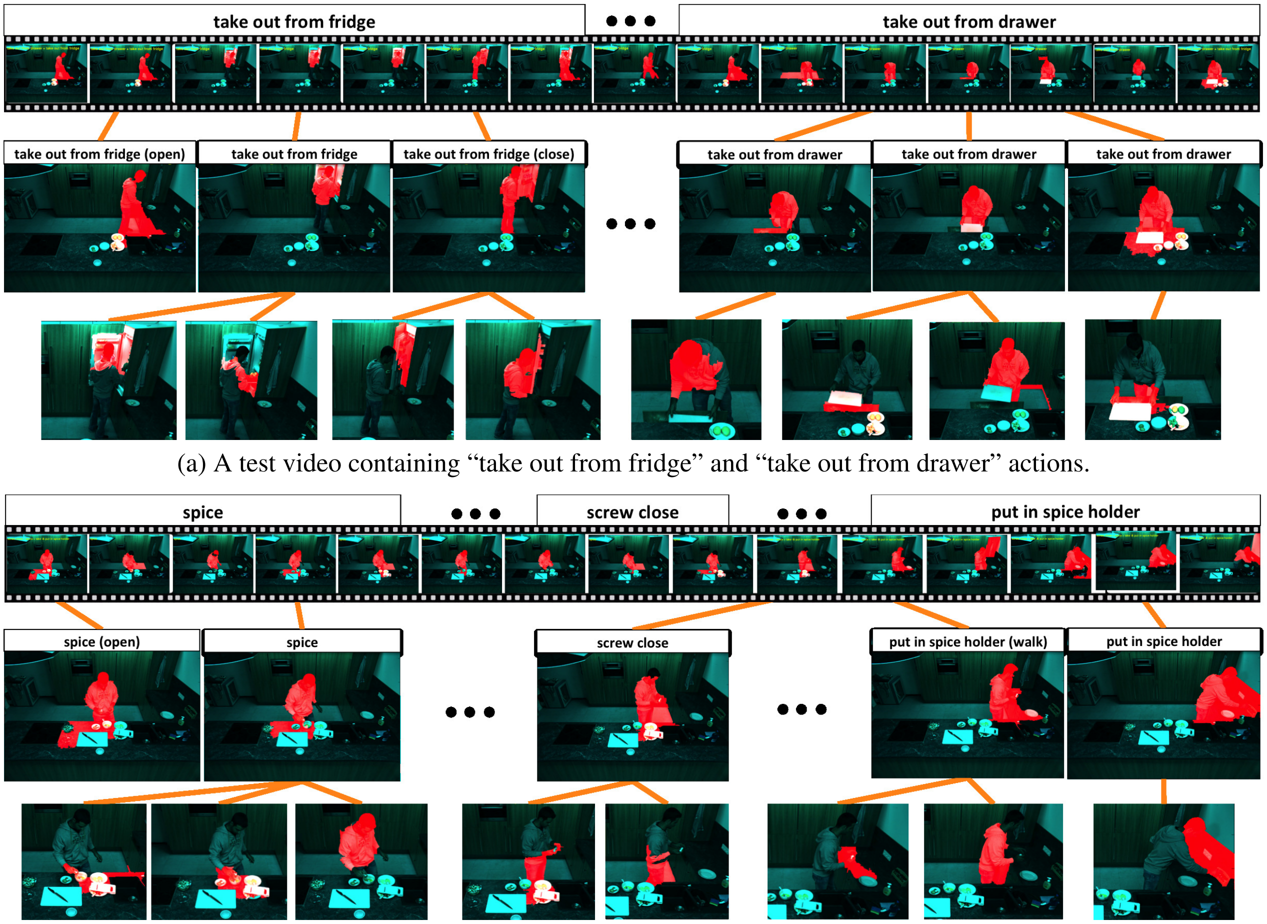


Figure 6. **Action parsing performance.** We report mean Average precision (mAP) of our method and the sliding window baseline on MPI Cooking with respect to different overlapping thresholds that determine whether an action (or MAE) segment is correctly localized.

We use the original fine-grained action labels provided in the MPI Cooking dataset as the MAEs at the bottom level of the hierarchy, and automatically generate a set of higher-level labels by composing the fine-grained action labels. The detailed setups are explained in Sec. S2 of the supplementary material [1]. Examples of higher-level action labels are: "cut apart - put in bowl", "screw open - spice - screw close". We only consider labels with length ranging from 1 to 4, and occur in the training set for more than 10 times. In this way, we have in total 120 action and MAE labels for parsing evaluation.

We compare our result with the sliding window baseline. The curves are shown in Fig. 6. Our method shows consis-



(a) A test video containing “take out from fridge” and “take out from drawer” actions.

(b) A test video containing “spice”, “screw close” and “put in spice holder” actions.

Figure 7. **Action parsing.** This figure shows the output of our action parser for two test videos. For each video, we visualize the inferred fine-grained action labels (shown on top of each image), the MAE segments (the red masks in each image) and the parent-child relations (the orange line). As we can see, our action parser is able to parse long video sequences into representative action patterns (i.e. MAEs) at multiple scales. Note that the figure only includes a few representative nodes of the entire tree obtained by our parser, we provide more visualizations in the supplementary video [1].

tent improvement over the baseline using different overlap threshold. If we consider an action segment is correctly localized based on “intersection-over-union” score larger than 0.5 (the PASCAL VOC criterion), our method outperforms the baseline by 8.5%. The mean performance gap (averaged over all different overlap threshold) between our method and the baseline is 8.6%. Some visualizations of action parsing results are shown in Fig. 7. As we can see, the story of human actions is more than just the actor: as shown in the figure, the automatically discovered MAEs cover different aspects of an action, ranging from human body and parts to spatiotemporal segments that are not directly related to humans but carry significant discriminative power (e.g. a piece of fridge segment for the action “take out from fridge”). This diverse set of mid-level visual patterns are then organized in a hierarchical way to explain the complex store of the video at different levels of granularity.

6. Conclusion

We have presented a hierarchical *mid-level action element (MAE)* representation for action recognition and parsing in videos. We consider a weakly supervised setting, where only the action labels are provided in training. Our method automatically parses an input video into a hierarchy of MAEs at multiple scales, where each MAE defines an action-related spatiotemporal segment in the video. We develop structured models to capture the rich semantic meanings carried by these MAEs, as well as the spatial, temporal and hierarchical relations among them. In this way, the action and MAE labels at different levels of granularity are jointly inferred. Our experimental results demonstrate encouraging performance over a number of standard baseline approaches as well as other reported results on several benchmark datasets.

Acknowledgments We acknowledge the support of ONR award N00014-13-1-0761.

References

[1] Supplementary material and visualization video. http://web.stanford.edu/~yukez/iccv2015.html.

[2] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Survey*, 2011.

[3] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005.

[4] P. Bojanowski, R. Lajugie, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic. Weakly supervised action labeling in videos under ordering constraints. In *ECCV*, 2014.

[5] W. Brendel and S. Todorovic. Learning spatiotemporal graphs of human activities. In *ICCV*, 2011.

[6] S. Clinchant, J.-M. Renders, and G. Csurka. Trans-media pseudo-relevance feedback methods in multimedia retrieval. In *Advances in Multilingual and Multimodal Information Retrieval*, pages 569–576. Springer, 2008.

[7] T.-M.-T. Do and T. Artieres. Large margin training for hidden markov models with partially observed states. In *ICML*, 2009.

[8] O. Duchenne, I. Laptev, J. Sivic, F. Bash, and J. Ponce. Automatic annotation of human actions in video. In *ICCV*, 2009.

[9] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, pages 726–733, 2003.

[10] I. Endres and D. Hoiem. Category independent object proposals. In *ECCV*, 2010.

[11] A. Gaidon, Z. Harchaoui, and C. Schmid. Activity representation with motion hierarchies. *ICCV*, 2014.

[12] M. Hoai, Z.-Z. Lan, and F. De la Torre. Joint segmentation and classification of human actions in video. In *CVPR*, 2011.

[13] A. Jain, A. Gupta, M. Rodriguez, and L. S. Davis. Representing videos using mid-level discriminative patches. In *CVPR*, 2013.

[14] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[15] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/THUMOS14/, 2014.

[16] A. Klaser, M. Marszalek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference*, 2008.

[17] A. Kovashka and K. Grauman. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In *CVPR*, 2010.

[18] T. Lan, L. Chen, Z. Deng, G.-T. Zhou, and G. Mori. Learning action primitives for multi-level video event understanding. In *International Workshop on Visual Surveillance and Re-Identification*, 2014.

[19] T. Lan, T.-C. Chen, and S. Savarese. A hierarchical representation for future action prediction. In *ECCV*, 2014.

[20] T. Lan, Y. Wang, and G. Mori. Discriminative figure-centric models for joint action localization and recognition. In *ICCV*, 2011.

[21] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[22] I. Laptev and P. Pérez. Retrieving actions in movies. In *ICCV*, 2007.

[23] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157, 1999.

[24] S. Ma, J. Zhang, N. Ikizler-Cinbis, and S. Sclaroff. Action recognition and localization by hierarchical space-time segments. In *ICCV*, 2013.

[25] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009.

[26] J. C. Niebles, C.-W. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010.

[27] D. Oneata, J. Verbeek, and C. Schmid. Action and event recognition with fisher vectors on a compact feature set. In *ICCV*, 2013.

[28] D. Oneata, J. Verbeek, and C. Schmid. Efficient Action Localization with Approximately Normalized Fisher Vectors. In *CVPR*. IEEE, 2014.

[29] D. Oneata, J. Verbeek, and C. Schmid. The lear submission at thumos 2014. In *CVPR THUMOS challenge*, 2014.

[30] F. Perronnin, J. Sanchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010.

[31] H. Pirsiavash and D. Ramanan. Parsing videos of actions with segmental grammars. In *CVPR*, 2014.

[32] M. Raptis, I. Kokkinos, and S. Soatto. Discovering discriminative action parts from mid-level video representations. In *CVPR*, 2012.

[33] M. Raptis and L. Sigal. Poselet key-framing: A model for human activity recognition. In *CVPR*, 2013.

[34] M. D. Rodriguez, J. Ahmed, and M. Shah. Action mach: A spatial-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008.

[35] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, 2012.

[36] M. Ryoo and J. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *ICCV*, 2009.

[37] E. Shechtman and M. Irani. Space-time behavior based correlation. In *CVPR*, 2005.

[38] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012.

[39] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. In *CVPR*, 2012.

[40] Y. Tian, R. Sukthankar, and M. Shah. Spatiotemporal deformable part models for action detection. In *CVPR*, 2013.

[41] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013.

[42] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010.