Space-Time Tree Ensemble for Action Recognition

Shugao Ma[1], Leonid Sigal[2], Stan Sclaroff[1]
[1]Computer Science Department, Boston University. [2]Disney Research Pittsburgh.

Introduction and Motivation: Human actions [2] and interactions are inherently structured patterns of body movements. A single structured model, such as those explored in [1, 3, 4], is insufficient to represent an action category in all but the simplest scenarios. Foremost, the execution of the action may differ from subject to subject; furthermore, the video capture process introduces intra-class variations due to occlusions and/or variations in camera viewpoint. As a consequence, the resulting space-time and appearance variations necessitate using a *collection* of spatio-temporal structures that can best represent the action at large.

We propose a method that discovers a collection of hierarchical space-time trees from video training data and subsequently learns a discriminative action model that builds on these discovered trees to recognize and spatially localize actions in videos. Both the model parameters and the topology of the tree structures are learned automatically from training data. The only supervision that is needed for learning is the action labels of the training videos, *i.e.*, bounding box annotations on video frames are unnecessary. Fig. 1 illustrates one simple discovered tree and its best match in a test video.

Formulation: A video is represented as a graph $G = \{V, A^t, A^s, A^h, F\}$. V is the set of vertices that are the space-time sub-volumes of the video. A^t, A^s and A^h are the time, space and hierarchical adjacency matrices containing edge labels. The rows of matrix are visual features extracted from the vertices. For each action class a, a collection of trees is then used in constructing an ensemble classifier:

$$S_a(G, \mathcal{T}) = \mathbf{w}^T \cdot \Phi(G, \mathcal{T}) = \sum_{m \in \{1,\dots,|\mathcal{T}|\}} w_m \phi_m(G, \mathcal{T}_m), \qquad (1)$$

where G denotes a test input video, $\mathcal{T}$ is the set of learned tree structures for class a and $\mathcal{T}_m$ is one of such trees in this set, and $\mathbf{w} = \{w_m; m \in \{1,\dots,|\mathcal{T}|\}\}$ is the learned weight vector. Each ϕ_m is a scoring function that measures compatibility (or degree of presence) of $\mathcal{T}_m$ in video G. In the multi-class classification setting, the predicted action class a^* of G is computed by $a^* = arg\max_a S_a(G, \mathcal{T})$.

We formalize a tree as $\mathcal{T}_m = \{N, E^t, E^s, E^h, \beta\}$ where N, $\{E^t, E^s, E^h\}$ are the nodes and adjacency matrices respectively. β are discriminative weights associated with the nodes and edges. Each node $n_i \in N$ is an index into a learned discriminative action word vocabulary $\mathcal{W}_a$ for class a; each edge E^k_{ij} $(k \in \{t, s, h\})$ is associated with a corresponding temporal, spatial or hierarchical relationship between nodes i and j, similar to the relations defined for A^k in graph G. The matching score of a tree to a graph is computed as follows:

$$\phi_m(G,\ \mathcal{T}_m) = \psi(\{\ \beta \cdot \varphi(G, \mathcal{T}_m, \mathbf{z}) \mid \mathbf{z} \in Z(G, \mathcal{T}_m)\}), \qquad (2)$$

where $\mathbf{z}$ is latent variable that represents a match of a tree $\mathcal{T}_m$ to the video G: $\mathbf{z}$ is realized as $\mathbf{z} = (z_1, \dots, z_{|N|})$ where z_i is the index of the vertex in G that is matched to the ith node in $\mathcal{T}_m$. ψ is a pooling function over the matching scores of the set of all possible (partial) matches $Z(G, \mathcal{T}_m)$.

The matching score of a specific match $\mathbf{z}$ to $\mathcal{T}_m$ is:

$$\begin{aligned} \beta \cdot \varphi(G, \mathcal{T}_m, \mathbf{z}) = & \sum_{n_i \in N} \beta_i\ p_n(z_i,\ n_i) \qquad (3) \\ & + \sum_{k \in \{t,s,h\}} \sum_{\substack{E^k_{ij} \in E^k \\ E^k_{ij} \neq 0}} \beta^k_{ij}\ p_k(A^k_{z_i z_j},\ E^k_{ij}). \end{aligned}$$

where β_i and β^k_{ij} $(k \in \{t, s, h\})$ are the tree node weights and edge weights respectively. The function p_n scores compatibility of the tree nodes with graph vertices; p_t, p_s and p_h score compatibility of the temporal, spatial and hierarchical graph edges with tree edges. Partial matching is possible by adding a *null* vertex $v_\varnothing$ to V as the 0th vertex and also adding a 0th row and column of zeros to A^s, A^t and A^h. Any node in $\mathcal{T}_m$ not matched to a vertex in G is assigned to match the 0th vertex. If *max pooling* is selected for ψ, *i.e.*, only the best partial match is considered, inference can be done efficiently via dynamic programming.

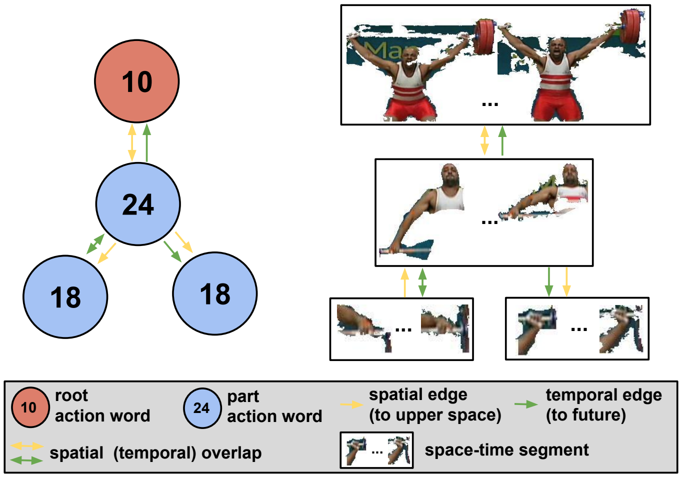

Figure 1: An example tree structure discovered by our approach (left) for the *lifting* action and its best match (right) in a test video. In the tree, one node (red) indexes to a root action word and is matched to a space-time segment (STS) of the upward movement of the whole body; three nodes (blue) index to the part action words and are matched to STSs of the upper-body and two temporally consecutive movements of the left arm and forearm respectively.

Discovering the Tree Structures $\mathcal{T}$: Given a tree structure, parameters can be learned in variety of ways, *e.g.*, using latent SVM [4]. However, discovering the tree structures themselves is the key challenge as: (1) the space of tree structures is exponential in the number of tree nodes and types of relationships allowed among the tree nodes; (2) partial presence of the trees needs to be considered; (3) without annotation of body parts, the tree nodes themselves are to be discovered. We first mine frequent subtrees from graphs of training videos using a graph mining technique. Redundant structures are subsequently suppressed by clustering the trees in a way that considers both their structure similarity and discriminative parameters (*i.e.*, β). For the remaining trees, we compute their activation entropy, which is small if the tree appears in few actions. A compact set of tree structures is then selected that have low activation entropy values.

Performance: Our proposed method achieves state-of-the-art performance in recognizing and localizing human actions and interactions in two benchmark video datasets: UCF-Sports and HighFive. We also show generalization of the learned trees by cross-dataset validation, achieving promising results on Hollywood3D dataset using trees learned on the HighFive dataset.

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

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

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

[4] Y. Wang and G. Mori. Hidden part models for human action recognition: Probabilistic versus max margin. *TPAMI*, 33(7):1310–1323, 2011.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.