ActionVLAD: Learning spatio-temporal aggregation for action classification

Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, Bryan Russell

One of the main limitations of the two-stream architectures is that they largely disregard the long-term temporal structure of the video and essentially learn a classifier that operates on individual frames or short blocks of few (up to 10) frames, which can force a consensus of classification scores over different segments of the video. This raises the question of whether such temporal averaging is capable of modeling the complex Spatio-temporal structure of human actions. The desired aggregator needs to operate over the entire video on both the appearance and the motion without requiring every frame to be classified into one of the actions.

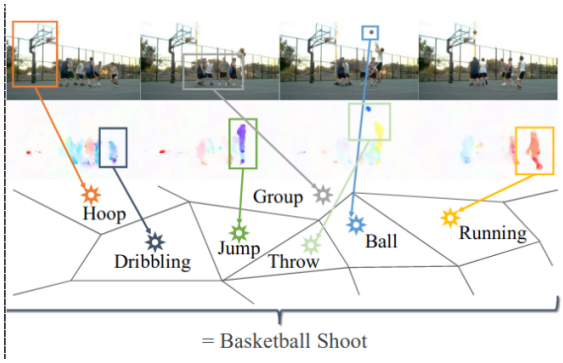

Figure 1: How do we represent actions in a video? We propose ActionVLAD, a spatio-temporal aggregation of a set of action primitives over the appearance and motion streams of a video. For example, a basketball shoot may be represented as an aggregation of appearance features corresponding to 'group of players', 'ball' and 'basketball hoop'; and motion features corresponding to 'run', 'jump', and 'shoot'. We show examples of primitives our model learns to represent videos in Fig. 6.

ActionVLAD: to solve this, the authors propose a centroid-based aggregation approach. Given N x T D-dimensional spatiotemporal features. First, we define a vocabulary of K D-dimensional centroids. For each centroid, the distance between this centroid and all of the features is computed, and the final aggregation is the weighted sum of all these N x T distances, where the weights are a softmax over the distances. As such, the results of the aggregation are K D-dimensional feature vectors. Such aggregation is then applied to each stream independently, and then the classification scores are combined.

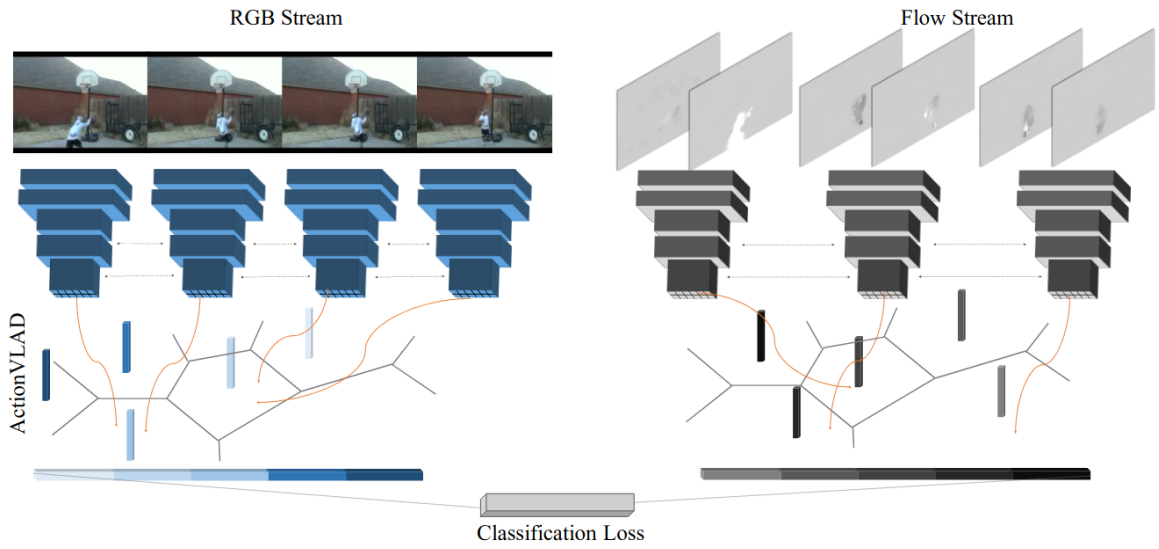

Figure 2: Our network architecture. We use a standard CNN architecture (VGG-16) to extract features from sampled appearance and motion frames from the video. These features are then pooled across space and time using the ActionVLAD pooling layer, which is trainable end to end with a classification loss. We also experiment with ActionVLAD to fuse the two streams (Sec. 3.3).