Temporal Segment Networks for Action Recognition in Videos.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, Luc Van Gool

The application of ConvNets to action recognition is constrained by three main obstacles:

- Lack of long-range temporal structure: the previous methods only focused on appearances and short-term motions (up to 10 frames), thus lacking the capacity to incorporate long-range temporal structure.
- Untrimmed videos: to have good real-world results, the model needs to be adapted for untrimmed videos where the action may occupy only a small portion of the video.
- Optical flow: the extraction of the optical flow for motion processing becomes a big performance bottleneck with larger datasets that are necessary for video recognition.

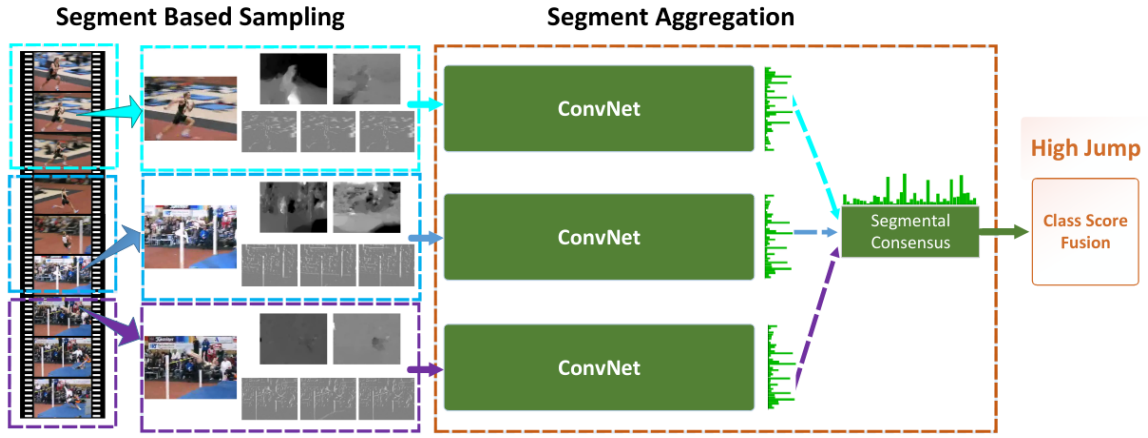

Fig. 1. Temporal segment network: One input video is divided into K segments (here we show the $K = 3$ case) and a short snippet is randomly selected from each segment. The snippets are represented by modalities such as RGB frames, optical flow (upper grayscale images), and RGB differences (lower grayscale images). The class scores of different snippets are fused by an the segmental consensus function to yield segmental consensus, which is a video-level prediction. Predictions from all modalities are fused to produce the final prediction. ConvNets on all snippets share parameters.

TSNs tries to solves these issues by providing data and computationally efficient method that can learn video representations across a long-range temporal structure. Instead of working on a single frame or a short stack of frames, TSNs operate on a sequence of short snippets sampled from the entire video. To make the sampled snippets represent the whole video, the video is first divided into several segments of equal duration, and then one snippet is randomly sampled from each segment. Each snippet is then passed through the same CNN, and the output of each one is combined with a segmental consensus module that turns segment-based features into video-level features and then into a final class prediction.

In order to learn parameters dependent on the large skunk of the video corresponding to the desired action and not just one snippet, the gradient of the aggregator must flow to all or the majority of the sampled snippets. In the case of max pooling, the gradient will only flow to one snippet and in this case,

we lack the capacity to model multiple snippets. With average pooling, the gradients will flow to all of the snippets equally, but in this case, we might want to only focus on the parts of the video where there is the desired action and ignore the background. To strike a balance, the authors propose different variants like top-k pooling, linear weighting with learned weights, or attention-based aggregation.

Action recognition in untrimmed videos: with long videos where there is a high probability that most of the snippets do not correspond to any actions. The authors propose to sample snippets at a fixed rate, and then windows of different temporal sizes (1, 2, 4, 8, and 16) are used over the sampled snippet. The score of each window is computed as the max over all of its snippets. Finally, a top-k aggregation is applied over the class scores of each window to get the final scores.