Large-scale Video Classification with Convolutional Neural Networks

1 Citation

Karpathy, Andrej, et al. ”Large-scale video classification with convolutional neural networks.” Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2014.

`http://vision.stanford.edu/pdf/karpathy14.pdf`

2 Abstract

We do video classification on a 1M sports video dataset with 487 classes. Our best network considers multiple resolutions (i.e a fovea approach) and slowly fuses information across time scales to set the state of the art (63.9% vs 55.3%). Even a single frame neural network sets state of the art (60.9%). We can generalize to the UCF-101 action recognition dataset by fine-tuning the last few layers of our network.

3 Introduction

We collect the Sports-1M dataset, which has 1M sports videos with 487 classes.

We design a slow-fusion network that gets good accuracy. To ensure that training is not too slow, we use a fovea approach where we have two streams - one that views the whole frame at low resolution (context) and another that views the center of the frame at high resolution (the fovea). In humans, the fovea is the area of the eye that can view a piece of low resolution input in high resolution.

4 Related Work

A typical video classification system works like this. We first extract shallow features either densely or sparsely (i.e. pick a few frames of interest). Then we combine them into a fixed sized feature vector after quantizing them with k-means clustering. Finally, we send these feature vectors through an SVM.

CNNs making this easier by learning features on their own. All we need to do is specify the architecture.

We think the reason CNNs haven’t made great strides on video classification is because the datasets are small. Our Sports-1M dataset aims to change that.

5 Models

We treat videos as bags of fixed-size clips. This means our models can assume fixed input size (10 frames = 1/3 of a second).

Now, a clip is a series of frames. How do we leverage information across frames? We propose four approaches:

1. Single Frame - Just take one frame in the clip and classify (i.e. send it through fully connected layers and softmax).
2. Late Fusion - Take the first and last frames in the clip, send them through conv (includes pooling and ReLU) layers. Concatenate their final feature maps and classify those.
3. Early Fusion - Use conv layers that incorporate the time dimension (3D convolution) and consider the middle few frames of the video. Send these through the CNN and classify them.
4. Slow Fusion - Divide the frames to several batches. Send each batch through some 3D conv layers. Then combine pairs of batches. Then send the new batches through more conv layers. Combine pairs again. Keep repeating until there is only one batch and classify that.

To make the compute more efficient, we use our fovea approach. The frames have size 178 × 178. We cut this resolution in half to get our context frames at size 89 × 89. We also take the middle 89 × 89 pixels to get our fovea frames. The fovea and context each go through identical conv layers and their final feature maps get concatentated at the end.

We train with Downpour SGD with momentum (0.9) and weight decay (0.0005) on about 50 machines with minibatches of size 32. Learning rate starts at 0.0001 and gets cut when validation error stops improving. We subtract mean activation from pixels.

For data augmentation, we start with the center 200 × 200 of the image, then select random crops and flips.

6 Results

Our Sports-1M labels come from the text metadata for the video, so you can consider the class labels to be weak. We do a 70/10/20 training/validation/testing split. There are about 2000 duplicate videos, but since we extract random clips from the videos, it probably isn't a big deal.

Training processes 5 clips a second and takes a month to finish. Using a better GPU could speed this up.

At test time, we take 20 clips from the video. For each clip, we take 4 random crops/flips. We run each through our model and average the class scores.

Our baseline model extracts popular computer vision features, turns them into a bag-of-words with k-means clustering and pooling on multiple scales, and then classifies them with a small neural network (this works better than an SVM).

Our CNN beats the baseline. The fovea approach gives a 2-4x speedup. Slow fusion is the best. The mistakes made by the model are sensible (e.g. confusing hiking for backpacking, confusing sledding for toboggan).

To test transfer learning, we fine-tune our model on UCF-101. We found that fine-tuning all layers or fine-tuning only the last fully connected layer performs worse than fine-tuning the last 3 layers of the CNN. We set state of the art on UCF-101. Training our network from scratch on this dataset does really poorly.

7 Conclusion

Slow-fusion is a good approach and the fovea technique gives us speed. Training on Sports-1M seems like a good way to transfer to other tasks.