SVM – Segment Video Machine

Jiaming Song   Yankai Zhang

When watching a video online, users might need:

- Detailed video description information
- Removal of repeating openings and endings
- Automatic labeling or tagging for uploaded videos
- Context-based segmentation inside one video

Introduction ❯ Our Target

A video player with the following innovative features:

- Open a video file, we'll give a segmentation and classification result for it
  - e.g. **0:00:00 – 0:10:23** news, **0:10:23 – 0:30:15** cartoon, and so on…
- The results are automatically inferred.
- User can set a filter to watch/ignore some specified segments or programs.

Methods ❯ Objective

Given a video from time 0 to T, we predict the genre for each video segment in time interval $[t, t + \Delta t)$, where $t \in [0, T]$, and Δt is the smallest time interval between two frames.

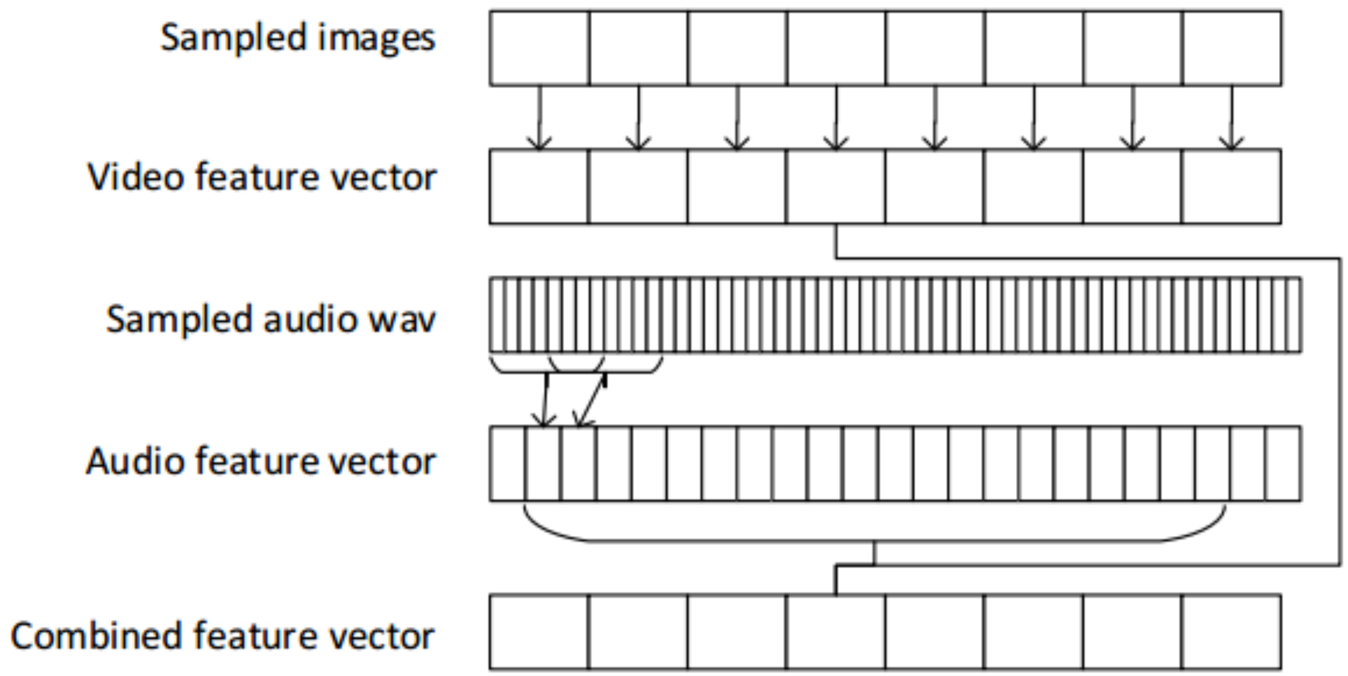
Our goal is to minimize this L1-loss function:

$$\int_0^{T-\Delta t} I\big(g(t) \neq \hat{g}(t)\big)dt$$

where $g(t)$ is the ground truth, $\hat{g}(t)$ is the prediction, and $I(x) = 0$ when x is false; 1 otherwise.

Using a segmentation tool, we can turn this into a classification problem.

Methods ❯ Pipeline



Combination Rule: **Features** from

1 sample image +

2 seconds of audio

Features are trained and tested using Linear Support Vector Machines. (LIBLINEAR)

Methods ❯ Philosophy I: Why 1+2?

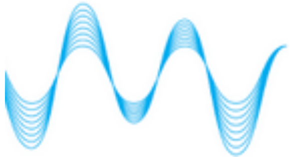
In most videos, the frame itself can describe everything.

(News is news)

However, in some videos, the frame can be misleading;

(Nature or news?)

Hence we need audio features to improve the accuracy in classification.

+  = news

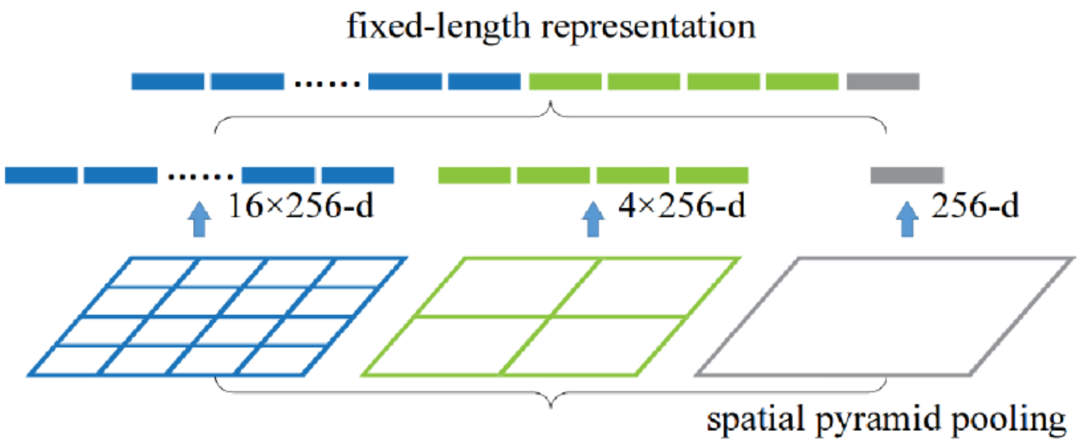
“紧紧围绕在徐老师周围”
“坚持SRT基本原则”

Methods ❯ Philosophy II: Why 1+2?

In most situations, a picture can be described using 1 short sentence.
Average sentence length = 7
Average characters per minute = 200
Average time used to speak one sentence = 7 / 200 * 60 = 2.1s

Pooling[1]

Uses pixel-level information.
Relatively Naive.



HOG[2]

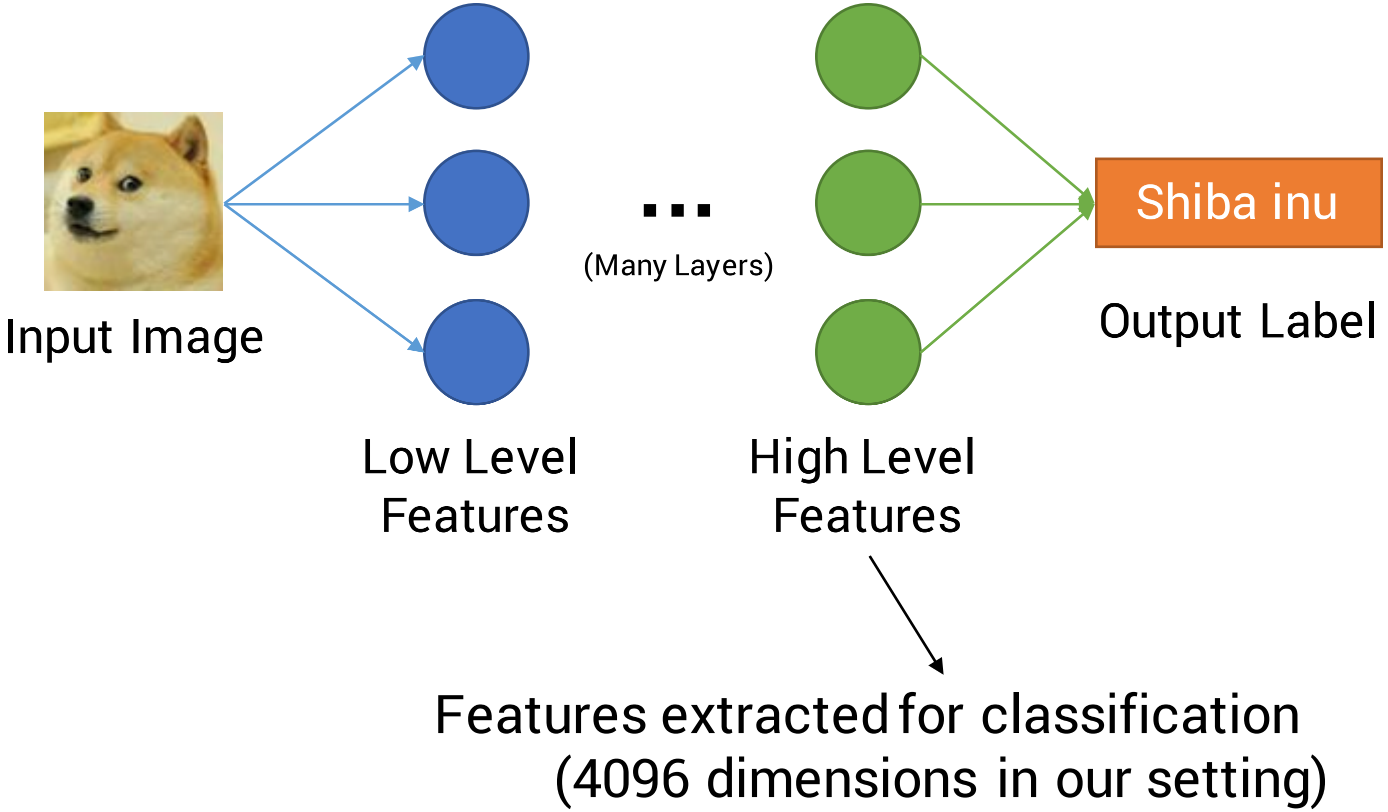
Histogram of Oriented Gradients
Used widely in detection.

[1]**He et al.** Spatial pyramid pooling in deep convolutional networks for visual recognition.
[2]**Dadal et al.** Histograms of oriented gradients for human detection.

Methods ❯ Frame Features

Deep Convolutional Neural Networks[3]

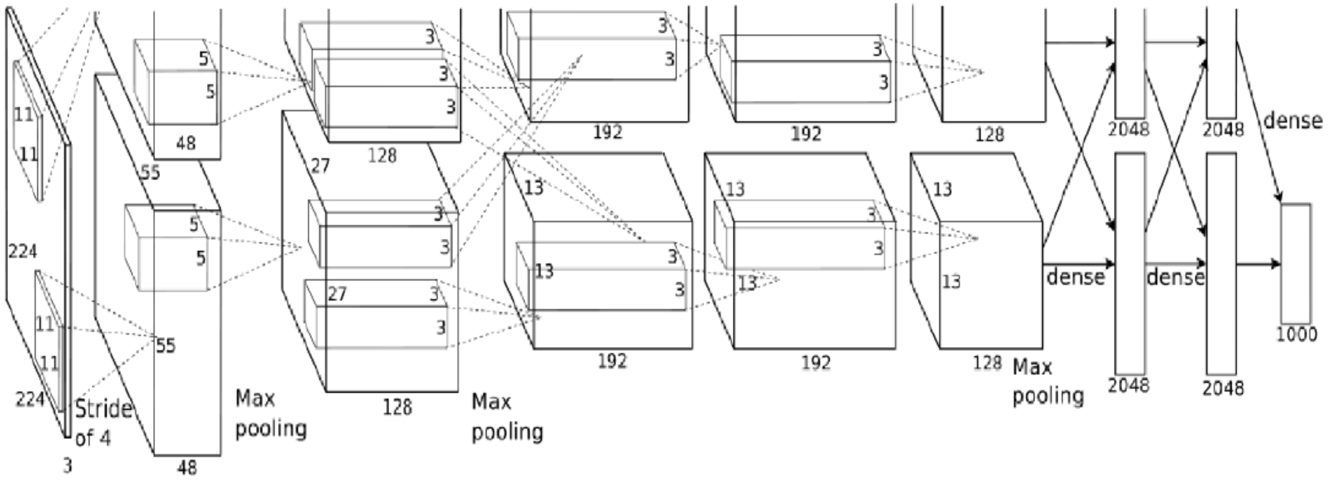
Such high accuracy.
Much layers.
Such many features.
Wow.



[3] **Donahue, et al. Decaf:** A deep convolutional activation feature for generic visual recognition.

Methods ❯ Frame Features

AlexNet – Our Feature Extraction Net



7 layers, 5 convolutional layers
Trained on the ImageNet dataset, which contains over 160G size of images, and 1000 classes, with an accuracy of ~56%

Over 500000 neurons and 60 million parameters.

Champion of 2012 ILSRVC contest.
The beginning of Deep Learning in Computer Vision.

LPCC

Everyone uses it.

$$c_0 = ln\sigma^2$$

$$c_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) c_k a_{m-k}, 1 \leq m \leq p$$

$$c_m = \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) c_k a_{m-k}, p < m \leq D$$

MFCC

Everyone uses it.

$$c_n = \sqrt{\frac{2}{K}} \sum_{k=1}^{K} (logS_k) \cos\left[\frac{n(k-0.5)\pi}{K}\right]$$

2 seconds of audio,
13 + 13 = 26 features per 20 milliseconds.

2600 features for audio features.

Philosophy III: Why 1 +2?

4096 frame features + 2600 audio features

= 6696 features

The dimensions for frame and audio features are similar, so neither part will take too much weight in classification.

But frame has a little higher weight, due to the more information it contains (“A picture is worth a thousand words”)

Python for frame extraction.

Matlab for LPCC and MFCC

OpenCV for Pooling and HOG

CUDA for DNN

LIBLINEAR for SVM

Dataset contains 5 classes, including

- nature
- news
- cartoon
- mv
- lecture

- ~1G of training set,
- ~200M of test set (which is totally different)

Training Set

Test Set

Results ❯

Frame Classification Test & Validation

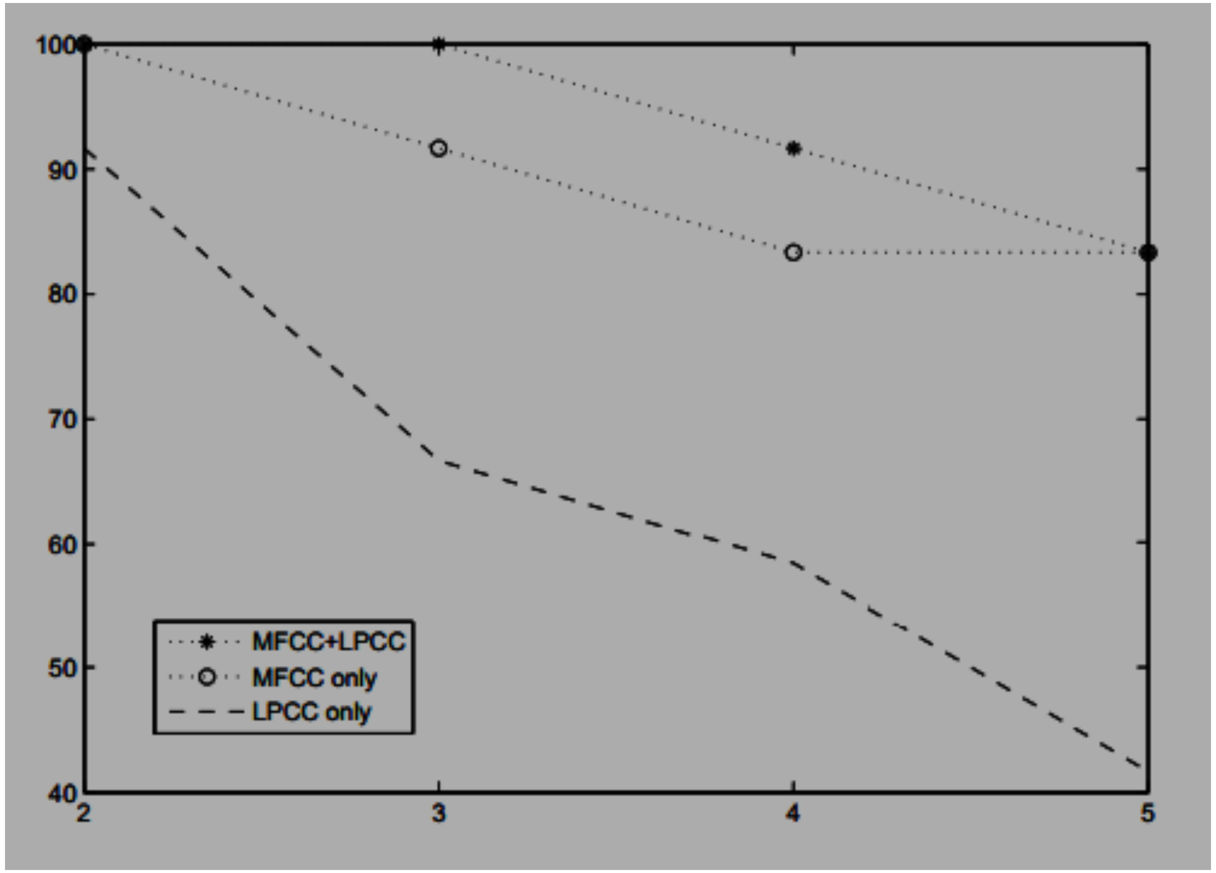
	Pooling	HOG	DNN
Validation	95.1%	99.3%	99.8%
Test	19.7%	46.25%	57.04%

The Test uses a very hard dataset, only to see the robustness of the features.

Results ❯ Image Processing Speed

SPP	HOG	DNN
10.1 Images/sec	35.9 Images/sec	24.7 Images/sec

Results ❯ Audio Classification



Accuracy with different number of audio features

Results ❯ Combined Classification

	Pooling+LPCC+MFCC	HOG+LPCC+MFCC	DNN+LPCC+MFCC
Validation	97.2%	87.1%	99.6%
Test	54.9%	80.2%	88.7%

Combined Classification vs Frame Only

	Pooling	HOG	DNN
Alone	52.4%	79.6%	80.5%
Combine	54.9%	80.2%	88.7%

Results ❯ **Conclusion**

DNN is much more robust than Pooling and HOG features.

MFCC is generally better than LPCC, but MFCC+LPCC is better than MFCC.

Classification with less classes is easier than training than multiple classes.

Using combined features is better than using frame or audio feature alone.

Contribution

Our models enables:

- Real time frame and audio feature extraction.
- Online and scalable training.
- Fast classification, segmentation and tagging.

Results ❯ Paper

An Implementation of Video Sementaition and Classifcation based on Video Features Extraction and Machine Learning

Jiaming Song JIAMING.TSONG@GMAIL.COM
Computer Science and Technology Department, Tsinghua University

Gerald Yankai Zhang ZYK12@MAILS.TSINGHUA.EDU.CN
Computer Science and Technology Department, Tsinghua University

Abstract

We present our system for video segmentation and classification with a novel way to extract multimedia features and to utilize machine learning methods. We extract SPP, HOG and DNN as visual features, LPC and MFCC as audio features from the original video file. We combine the features and then classify the video into five genres. We use the classification information to gather shots into programs. We implement an application that can parse the result of our algorithm, and playback the video with these additional information. We estimate our method and prove the robustness and efficent beyond traditional methods.

1. Introduction

In these days on the Internet, we have witnessed the continue increase of available network bandwidth. The network is capable to deal with video streams with higher and higher bitrates. Thus we are not surprised to see that video data are taking larger part of total network data. In compare with the fast development of network capability and the convenience brought by the feature of uploading self-made videos, the lack of detailed video description information is still an open problem that urges to be solved. For instance, most of the users are only interested in some specific parts of a video stream. Usually, users do not like the repeating opening and ending of a series. It is impossible to tag them by human hands. If there is a tool that can tag them automatically once the videos are uploaded, the experience of video watching will be greatly increased.

On the other hand, thanks to the flourish of machine learning, many problems which are once believed unsolvable are neatly settled. For instance, the computers are able to classify video and audio information into multiple genres with an incredible precision. But the true power of deep learning is still waiting to be developed.

In this application condition and theoretical background, we believe it is a right time to present our research topic, a video segmentation and classification system based on video features extraction and deep learning. It is a system that can segment a long video into individual programs and classify these programs into various genres. For each program, users can choose to watch some selected part of programs and ignore some of them, e.g. opening and ending of a program.

Our work is with the following contributions: 1) A set of video and audio features that can be used to solve programs classification and video segmentation problems in the future. 2) We present a novel application scenario for machine learning. 3) A fully functioned video processing and playback application that can be used in further study purpose.

The challenges that we are facing are listed as follows: 1) To combine multimedia information, we need to handle audio and video information properly at the same time. 2) To increase the precision of segmentation and classification, we need to try various ways in order to enhance the performance of our system. 3) To deal with the large amount of information, we need to find the balance point between running time and integrity of information.

The rest of this article will be arranged as follows: In Section 2, we introduce our methodology with explanations of features and algorithms we use. In Section 3, we briefly demonstrate our application. With dataset information presented, we describe our experiment environment and estimate our experiment in detail. In Section 4, we propose our plan for future work and time table.

Paper published on GitHub

Results ❯ Demo Software