Research Notes

Shailesh Kumar

Jun 21, 2019

Contents:

CHAPTER 1

Introduction

The articles in this collection are my notes from reading various papers and books during the course of my research work.

CHAPTER 2

Image Processing

CHAPTER 3

Machine Learning

CHAPTER 4

Computer Vision

CHAPTER 5

Deep Learning

5.1 General Overview

5.1.1 Applications

Convolutional Networks

- Image Classification
- Object Detection
- Object Localization
- Human Pose Estimation
- Action Classification
- Action Recognition
- Video Classification
- Image Feature Learning
- Medical Image Segmentation

5.2 Image Classification

5.3 Object Detection

5.4 Video Classification, Action Classification, Recognition, Detection

5.4.1 Large-scale video classification with convolutional neural networks

5.4.2 Youtube-8m: A large-scale video classification benchmark

5.4.3 Learning spatiotemporal features with 3d convolutional networks

In this note, we discuss [TBF+15].

Summary

An approach for learning spatiotemporal features using deep 3D convolutional networks is presented. The network is trained on large supervised dataset.

Claims

- 3D convolutional networks are more suitable for spatiotemporal feature learning.
- A homogeneous architecture with $3 \times 3 \times 3$ kernels in all layers performs quite well.
- The learned C3D (convolutional 3D) features with a simple linear classifier outperform state of the art methods on many benchmarks.

Further:

- Features are compact.
- Features are efficient to compute.
- Features are conceptually simple and easy to train.

Prior work

- Spatio-temporal interest points (STIP)
- SIFT-3D for action recognition
- HOG-3D for action recognition
- Cuboids features for behavior recognition
- Improved Dense Trajectories (iDT)
- Two stream networks
- Human detector and head tracking
- Deep Video [KTS+14]

Remarks on prior work

- Image based deep features are not directly suitable for video due to lack of motion modeling.
- iDT has good performance but it is computationally intensive and intractable on large scale datasets.

Major results

Benchmarks used

- Sports1M for action recognition
- UCF101 for action recognition
- ASLAN for action similarity labeling
- YUPENN for scene classification
- UMD for scene classification
- Object for object recognition

Proposals

Requirements of effective video descriptors:

- **Generic**: Can represent different types of videos well while being discriminative.
- **Compact**: Should be small in size to build databases of millions of videos. Storage and retrieval tasks should be scalable.
- **Efficient** to compute: Need to process thousands of videos every minute
- **Simple** to implement: Avoid complicated feature encoding methods and classifiers.

Examples of different types of videos: Landscapes, Natural scenes, Sports, TV shows, Movies, Pets, Food

Proposed 3D convnets

- $3 \times 3 \times 3$ convolution kernels for all layers work best.
- Encapsulate information related to objects, scenes and actions in a video.
- No need to fine-tune the model for specific task.
- Model appearance and motion simultaneously.
- Outperform existing results on 4 different tasks and 6 different benchmarks.
- Are compact and efficient to compute.
- No preprocessing on the video. Full video frames as input.
- 3D pooling
- Gradual pooling of space and time information.
- In 3D nets, the input as well as output is an image volume.

Comparison with 2D ConvNets

- 2D convolutional networks lose temporal information of input signal right after every convolution operation.
- Similarly, 3D pooling retains temporal information while 2D pooling loses it.
- 2D convolutional networks when applied on multiple images by treating them as different channels also result in an image only.
- Slow fusion model [KTS+14] uses 3D convolutions and average pooling in first 3 layers. Loses all the temporal information after this. The use of 3D convolutions in initial layers is probably the reason for its better performance compared to other models.

Notation

- c : number of channels in image
- l : number of frames in video clip
- h : height of frame
- w : width of frame
- $c \times l \times h \times w$: size of the video clip
- k : kernel spatial size
- d : kernel temporal depth
- $d \times k \times k$: size of convolutional kernel and pooling kernel

Basic study of 3D convolution and learning

Here our goal is to find out the right parameters for kernel temporal depth.

Network architecture

- Video frame size : 128×171.
- Non overlapped 16 frame clips.
- Jittering using random crop for training clips size: $3 \times 16 \times 112 \times 112$.
- 5 convolutional layers, 5 pooling layers.
- 2 FC layers.
- Final softmax layer.
- Number of filters in each layer: 64, 128, 256, 256, 256.
- Kernel temporal depth is variable.
- Appropriate padding is used [both spatially and temporally].
- Stride = 1
- Same convolution used. No change in image volume size in convolution layers.
- First max pooling layer of size $1 \times 2 \times 2$.
- Remaining max pooling layers of size $2 \times 2 \times 2$.

- Clip length changes as follows from layer to layer: $16 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$.
- Two 2048 (output size) FC layers
- Mini batches of 30 clips.
- Initial learning rate 0.003.
- Learning rate divided by 4 after 10 epochs.
- Training stopped after 16 epochs.

Temporal depth of convolutional kernel

Following options

- Homogeneous temporal depth (same in each layer)
- Varying temporal depth (different in each layer)

Setups

- depth-d homogeneous networks.
- depth-1 network is same as using 2D net on each frame.
- Increasing depth nets: 3-3-5-5-7.
- Decreasing depth nets: 7-5-5-3-3.

Results

- Clip accuracy is the main metric on UCF 101 dataset.
- Depth 3 network performs best among homogeneous nets.
- Depth 1 (2D net) is significantly worse than others.
- Depth 3 performs better than varying depth networks (gap is not much).
- Increasing depth net performs slightly better than decreasing depth net.

Spatiotemporal feature learning

Network architecture

- Only $3 \times 3 \times 3$ nets are used.
- 8 convolution layers, 5 pooling layers, two FC layers and then softmax layer.
- C - P - C - P - C - C - P - C - C - P - C - C - P - FC - FC - SF.

Training setup

- Sports 1M
- Five random clips each 2 second long from each video
- Frame size to 128×171.
- Random cropping (both spatial and temporal) to $16 \times 112 \times 112$ size.

- Horizontal flipping with 0.5 probability.
- SGD with mini batch of 30 clips.
- Initial learning rate: 0.003.
- Divided by 2 every 150K iterations.
- Optimization stopped after 1.9M iterations (13 epochs).

Results

- Accuracy of 84.4% at top-5 accuracy.

C3D video descriptor

- Break video into 16 frame clips with overlap of 8 frames.
- Compute the output activation of final FC layer for each clip.
- Average these activations over all clips.
- L2-Normalize the resultant vector

What C3D learns?

- Focuses on appearance in first few frames of a clip.
- Tracks the salient motion in remaining frames.
- Selectively attends to both motion and appearance.

Action recognition

Bibliography

5.5 Audio Events

CHAPTER 6

Sparse Representations

CHAPTER 7

Dictionary Learning

CHAPTER 8

Epilogue

CHAPTER 9

Indices and tables

- genindex
- modindex
- search

Bibliography

[KTS+14] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 1725–1732. 2014.

[TBF+15] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 4489–4497. 2015.

[KTS+14] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 1725–1732. 2014.

[TBF+15] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 4489–4497. 2015.