



From Classification to Temporal Localization with 3D Convolutions

Austin Le

April 16th, 2018



Papers

Method 2015

Learning Spatiotemporal Features with 3D Convolutional Networks

Dataset 2015/2016

ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding

Method 2017

Temporal Activity Detection in Untrimmed Videos with Recurrent Neural Networks



Papers

Method 2015

Learning Spatiotemporal Features with 3D Convolutional Networks

Dataset 2015/2016

ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding

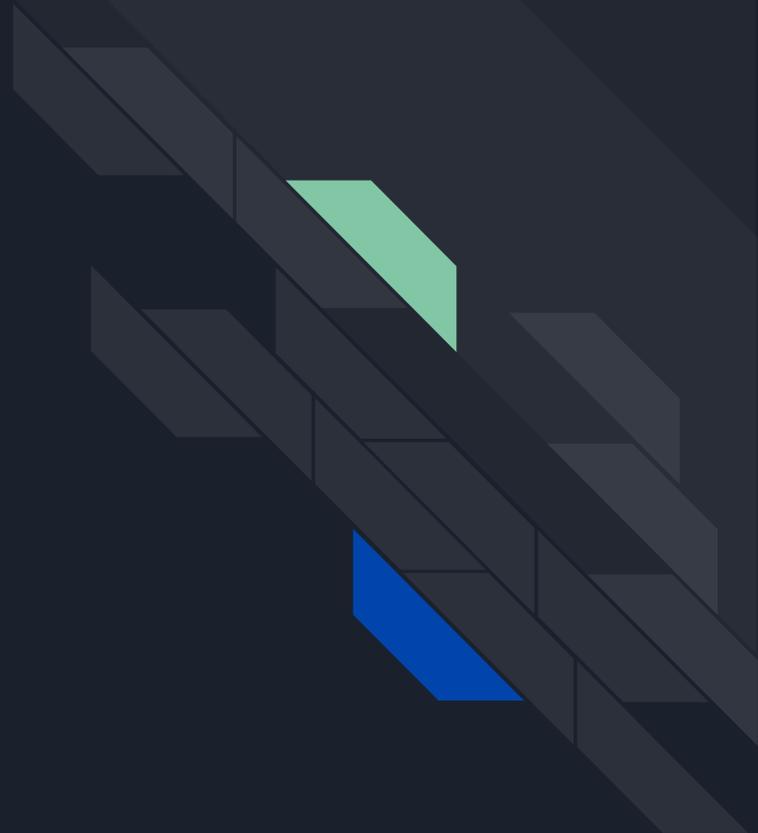
Method 2017

Temporal Activity Detection in Untrimmed Videos with Recurrent Neural Networks

Learning Spatiotemporal Features with 3D Convolutional Networks

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo
Torresani, Manohar Paluri

CVPR 2015





Motivations

- Need a **generic video descriptor** that helps solve large-scale video tasks in a homogenous way
- Image-based deep features are not enough!
 - We need to **model and learn motion** as well...

Let's use **3D convolutional networks!**



Related work

Last week...

- Karpathy et al. used full frames for training, but their method built on using 2D convolution and pooling
 - Slow Fusion method performs both spatial and temporal convolution at the beginning
- Simonyan and Zisserman used a two-stream architecture and then combined the stream outputs at the end

Now...

- The C3D model performs 3D convolutions and pooling that propagates temporal information throughout the entire network



Contributions

1. 3D convolutional deep networks are good feature learning machines that model appearance and motion simultaneously
2. A simple 3x3x3 convolutional kernel works well
3. Using these features in a simple linear model outperforms state-of-the-art (at the time)
4. Effective video descriptors (named **C3D**)

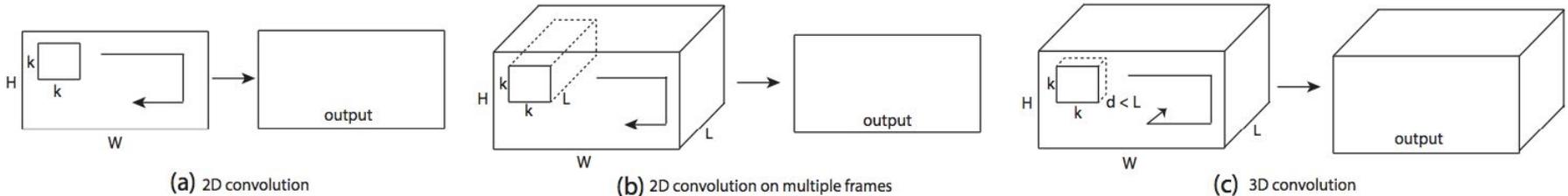


What makes an effective video descriptor?

1. **Generic**
 - To represent different types of video while being discriminative
2. **Compact**
 - To help with processing, storing, and retrieving data
3. **Efficient**
 - To quickly process videos in real world systems
4. **Simple**
 - To implement, and to work well with simple models

Insight: 3D operations are performed spatio-temporally

- **Problem**: 2D ConvNets lose temporal information right after every convolution operation



Lose temporal information.
Output is 2D.

Retain temporal information.
Output is 3D.



3D ConvNet settings

Question: 3x3xD kernels

- What's a good **depth** to use for 3D kernels?
- Experiment using UCF101 dataset

Settings

- Video frames resized to **128x171** (half-resolution)
- Videos split into **non-overlapped 16-frame clips**
- Network input dimensions: **3x16x128x171**



What's a good 3D ConvNet architecture?

Question: 3x3xD kernels

- What's a good **depth** to use for 3D kernels?
- Experiment using UCF101 dataset

Common network architecture

- 5 convolution and 5 pooling layers
- 2 fully-connected layers
- 1 softmax loss layer



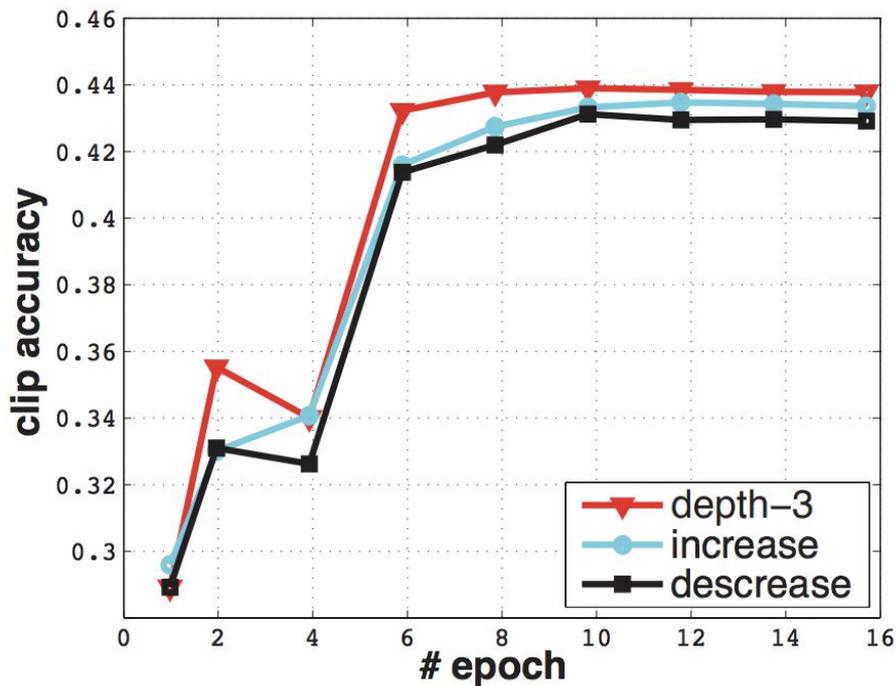
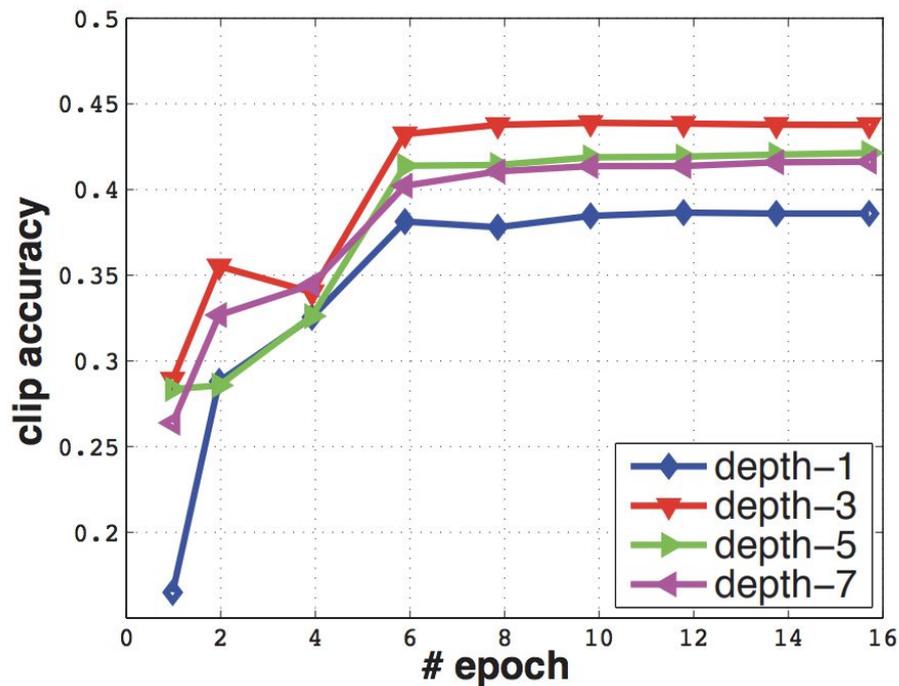
What's a good 3D ConvNet architecture?

Question: 3x3xD kernels

- What's a good **depth** to use for 3D kernels?
- Experiment using UCF101 dataset

- Homogenous temporal depth for all layers
 - depth-1: 1-1-1-1-1 (this is just 2D conv)
 - depth-3: 3-3-3-3-3
 - depth-5: 5-5-5-5-5
 - depth-7: 7-7-7-7-7
- Varying temporal depth between layers
 - Increasing: 3-3-5-5-7
 - Decreasing: 7-5-5-3-3

What's a good 3D ConvNet architecture?





What's a good 3D ConvNet architecture?

Observations

- Any two nets with a temporal depth difference of 2 differs only by 0.3% of total parameters
- Constant depth-3 performed the best
 - 3x3x3 is the best convolution kernel!

The C3D architecture

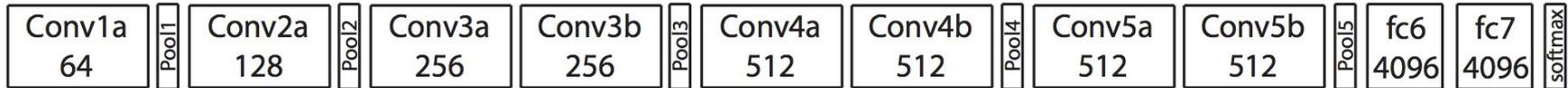


Figure 3. **C3D architecture.** C3D net has 8 convolution, 5 max-pooling, and 2 fully connected layers, followed by a softmax output layer. All 3D convolution kernels are $3 \times 3 \times 3$ with stride 1 in both spatial and temporal dimensions. Number of filters are denoted in each box. The 3D pooling layers are denoted from pool1 to pool5. All pooling kernels are $2 \times 2 \times 2$, except for pool1 is $1 \times 2 \times 2$. Each fully connected layer has 4096 output units.

8 convolution layers

5 pooling layers

2 fully-connected layers

1 softmax layer



Training C3D

1. Trained using Sports-1M from scratch

- Randomly extract 5 two-second center-cropped clips from each training video
- Random jittering and random flipping
- Predicting activity from video:
 - Randomly extract 10 clips, pass through network, and average results

2. Also fine-tuned from net pre-trained on I380K (internal dataset)



C3D video feature descriptors

Trained C3D can be used as feature extractor for videos

1. Split into 16-frame clips with 8-frame overlap
2. Pass through C3D network to get fc6 activations
3. Average the 16 fc6 activations, then L2 normalize

→ 4096-dim feature descriptor!

Evaluation: Sports-1M classification results

Method	Number of Nets	Clip hit@1	Video hit@1	Video hit@5
DeepVideo's Single-Frame + Multires [18]	3 nets	42.4	60.0	78.5
DeepVideo's Slow Fusion [18]	1 net	41.9	60.9	80.2
Convolution pooling on 120-frame clips [29]	3 net	70.8*	72.4	90.8
C3D (trained from scratch)	1 net	44.9	60.0	84.4
C3D (fine-tuned from I380K pre-trained model)	1 net	46.1	61.1	85.2

Table 2. **Sports-1M classification result.** C3D outperforms [18] by 5% on top-5 video-level accuracy. (*)We note that the method of [29] uses long clips, thus its clip-level accuracy is not directly comparable to that of C3D and DeepVideo.



What does C3D learn?

1. Learns about **appearance** in the **first few frames**
2. Tracks **salient motion** in the **subsequent frames**

So **C3D** differs from standard 2D ConvNets since it selectively learns about both **appearance** and **motion**.

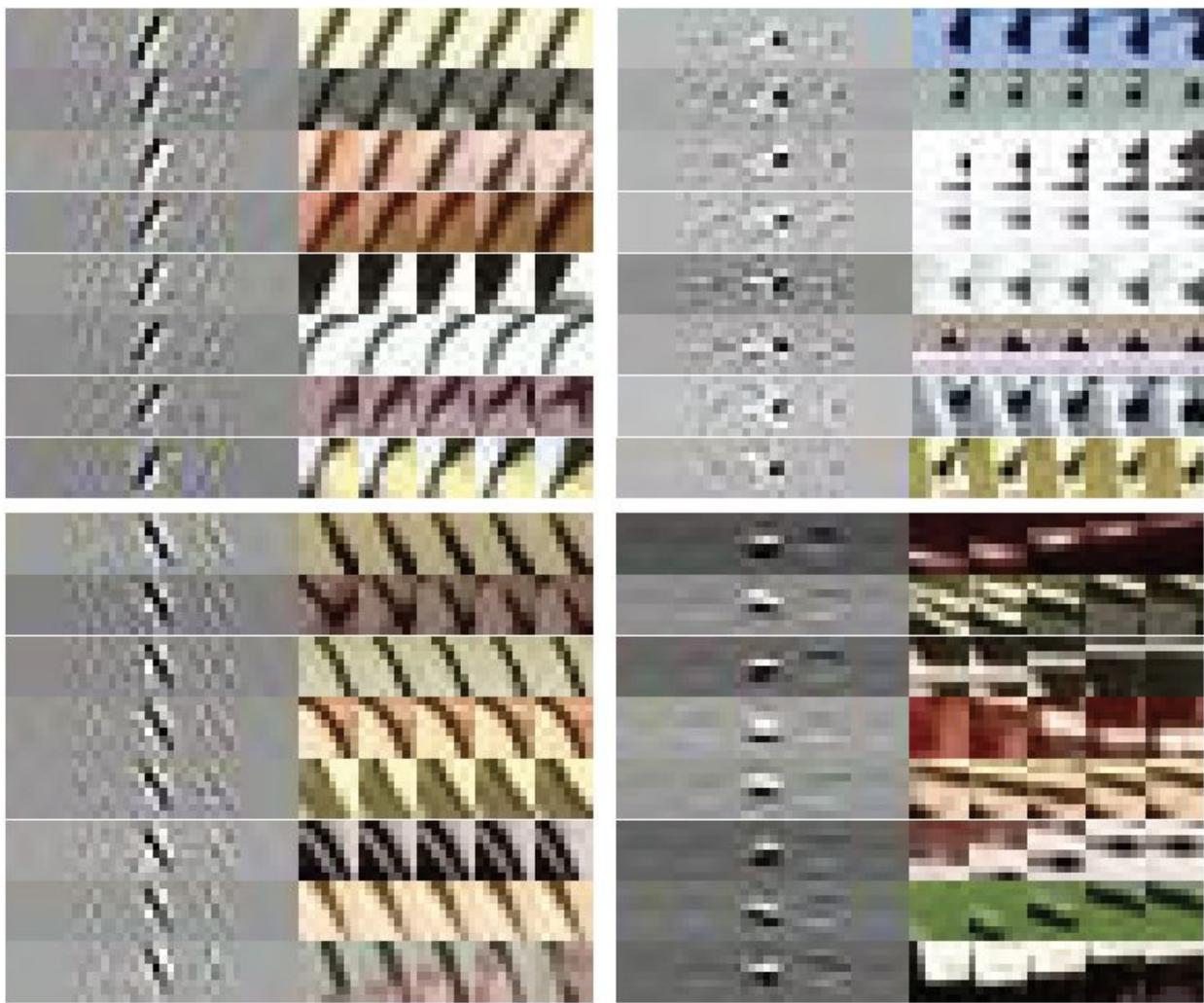
conv2a

Moving edges and blobs

Shot edges

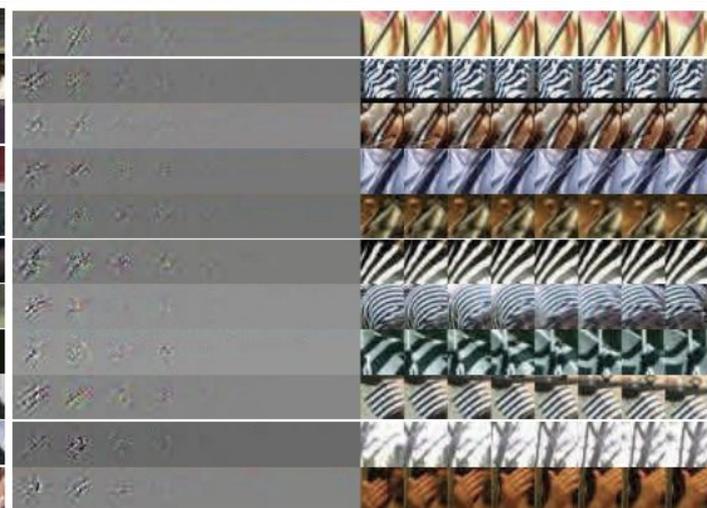
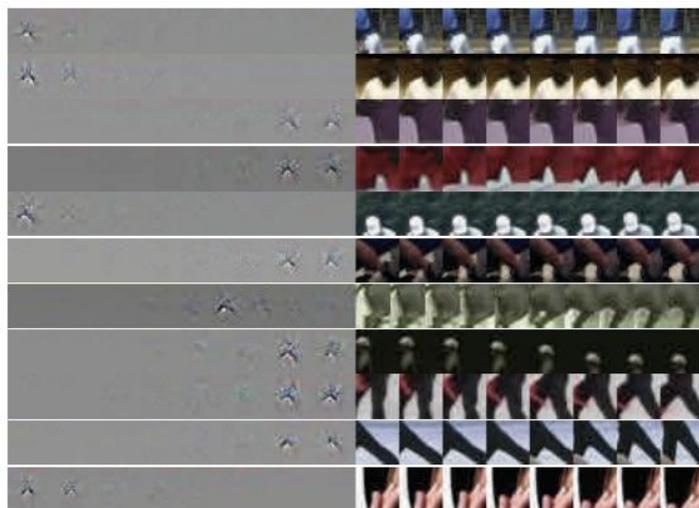
Edge orientation changes (not shown)

Color changes (not shown)

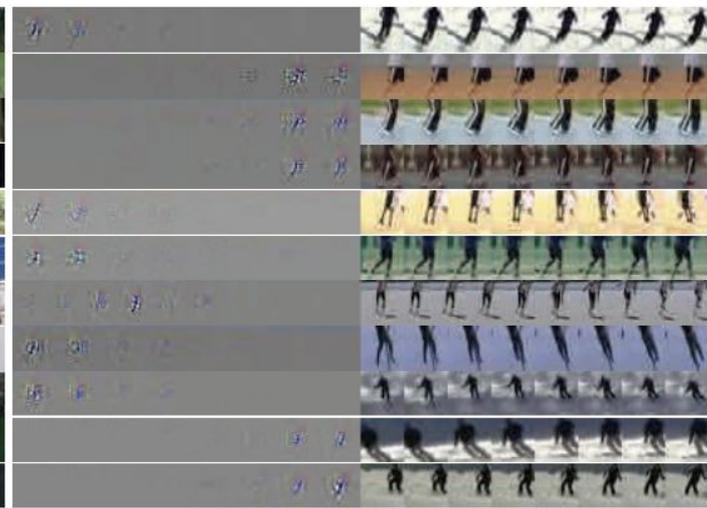
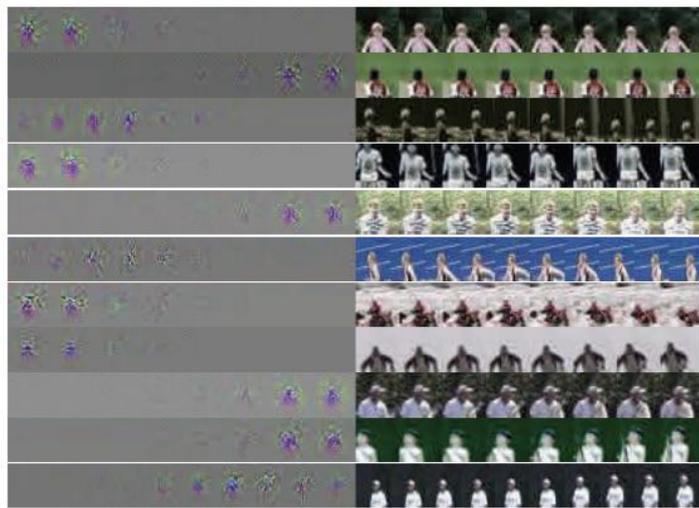


conv3b

Moving corners +
textures



Moving body parts



Object trajectories

Circular objects

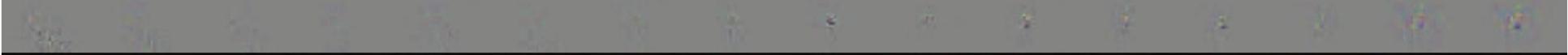


conv5b

Moving circular
objects (above)

Moving head
(below)

Face-related
motions (not
shown)





Evaluation 1: action recognition

- Evaluate C3D features on UCF101
 - 13,320 videos of 101 human action categories
- Input C3D features into multi-class linear SVM
- Train 3 networks with C3D features
 1. Trained on I380K (internal dataset)
 2. Trained on Sports-1M
 3. Trained on I380K and fine-tuned on Sports-1M

Evaluation 1: action recognition

- **C3D** combining all 3 nets performs the best
- **C3D** combined with iDT are highly complementary
- **C3D** outperforms other deep networks as well as some RNNs
- **C3D** only outperforms 2-stream networks and long-term models if combined with iDT

Method	Accuracy (%)
Imagenet + linear SVM	68.8
iDT w/ BoW + linear SVM	76.2
Deep networks [18]	65.4
Spatial stream network [36]	72.6
LRCN [6]	71.1
LSTM composite model [39]	75.8
C3D (1 net) + linear SVM	82.3
C3D (3 nets) + linear SVM	85.2
iDT w/ Fisher vector [31]	87.9
Temporal stream network [36]	83.7
Two-stream networks [36]	88.0
LRCN [6]	82.9
LSTM composite model [39]	84.3
Conv. pooling on long clips [29]	88.2
LSTM on long clips [29]	88.6
Multi-skip feature stacking [25]	89.1
C3D (3 nets) + iDT + linear SVM	90.4

Table 3. **Action recognition results on UCF101.** C3D compared with baselines and current state-of-the-art methods. Top: simple features with linear SVM; Middle: methods taking only RGB frames as inputs; Bottom: methods using multiple feature combinations.



Evaluation 2: action similarity labeling

- Evaluate C3D features on ASLAN (action similarity labeling)
 - 3,631 videos of 432 action categories
- **Task:** Given a pair of videos, do they show the same action?
- **Evaluation:**
 - Compute action similarity distance metrics to form a 48-dim feature vector for each video pair
 - Use linear SVM to determine same or different

Evaluation 2: action similarity labeling

C3D outperforms state-of-the-art with just simple feature averaging and a linear SVM

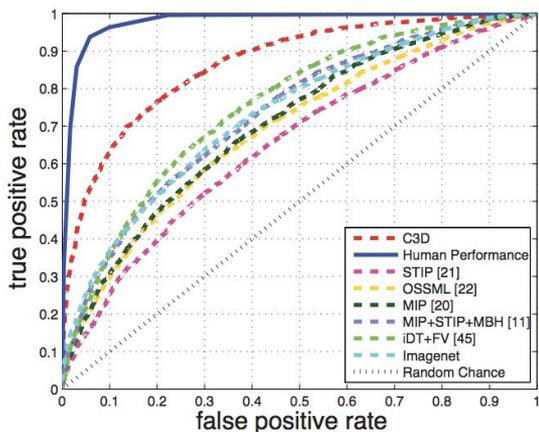


Figure 7. **Action similarity labeling result.** ROC curve of C3D evaluated on ASLAN. C3D achieves 86.5% on AUC and outperforms current state-of-the-art by 11.1%.

Method	Features	Model	Acc.	AUC
[21]	STIP	linear	60.9	65.3
[22]	STIP	metric	64.3	69.1
[20]	MIP	metric	65.5	71.9
[11]	MIP+STIP+MBH	metric	66.1	73.2
[45]	iDT+FV	metric	68.7	75.4
Baseline	Imagenet	linear	67.5	73.8
Ours	C3D	linear	78.3	86.5

Table 4. **Action similarity labeling result on ASLAN.** C3D significantly outperforms state-of-the-art method [45] by 9.6% in accuracy and by 11.1% in area under ROC curve.



Evaluation 3: scene and object recognition

- Evaluate C3D features on YUPENN, Maryland, egocentric
 - YUPENN: 420 videos of 14 scene categories
 - Maryland: 130 videos of 13 scene categories
 - egocentric: 42 types of everyday objects
- As before: extract features, use linear SVM for classification
- For video, ground-truth label is the most frequent label of clip

Evaluation 3: scene and object recognition

- Results are similar, if not better, than ImageNet
- Surprising, because C3D...
 - only trained on Sports-1M
 - without fine-tuning
 - uses linear classifier
- Suggests C3D is generic on capturing appearance and motion

Dataset	[4]	[41]	[8]	[9]	Imagenet	C3D
Maryland	43.1	74.6	67.7	77.7	87.7	87.7
YUPENN	80.7	85.0	86.0	96.2	96.7	98.1

Table 5. **Scene recognition accuracy.** C3D using a simple linear SVM outperforms current methods on Maryland and YUPENN.

Evaluation 4: runtime analysis

Method Usage	iDT CPU	Brox's CPU	Brox's GPU	C3D GPU
RT (hours)	202.2	2513.9	607.8	2.2
FPS	3.5	0.3	1.2	313.9
x Slower	91.4	1135.9	274.6	1

Table 6. **Runtime analysis on UCF101.** C3D is 91x faster than improved dense trajectories [44] and 274x faster than Brox's GPU implementation in OpenCV.

Two-stream network

C3D is also compact and simple!

Use PCA to project features to lower dimension and then re-evaluate on UCF101 classification task

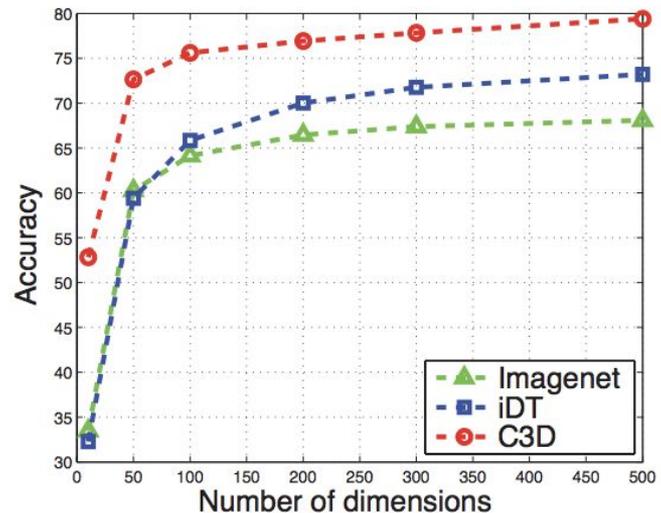


Figure 5. **C3D compared with Imagenet and iDT in low dimensions.** C3D, Imagenet, and iDT accuracy on UCF101 using PCA dimensionality reduction and a linear SVM. C3D outperforms Imagenet and iDT by 10-20% in low dimensions.

Is C3D a good generic feature?

Visualize the learned feature embedding using **t-SNE**

Features are semantically separable!

t-distributed stochastic neighbor embedding is a dimensionality reduction algorithm

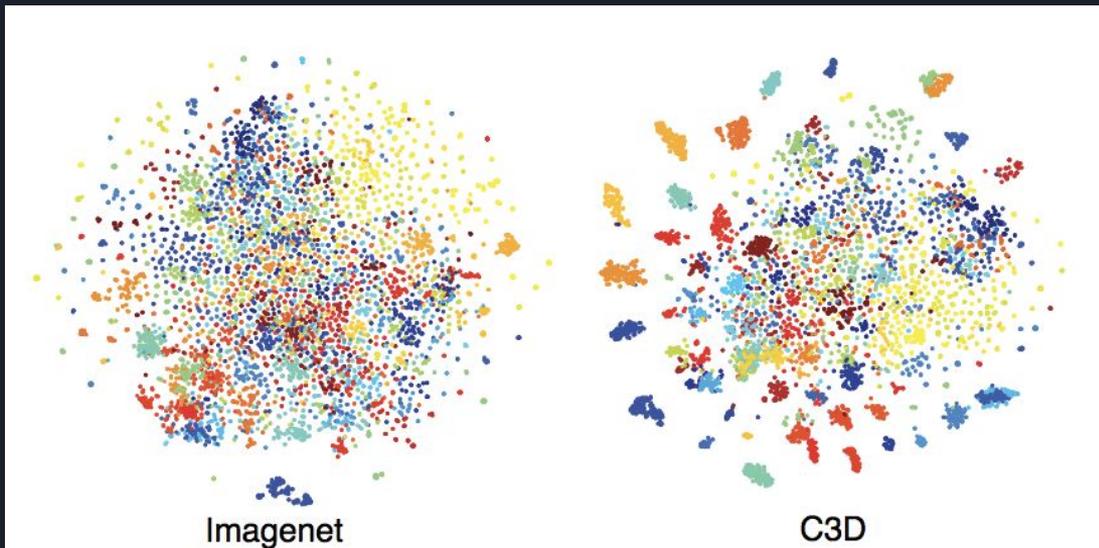


Figure 6. **Feature embedding.** Feature embedding visualizations of Imagenet and C3D on UCF101 dataset using t-SNE [43]. C3D features are semantically separable compared to Imagenet suggesting that it is a better feature for videos. Each clip is visualized as a point and clips belonging to the same action have the same color. Best viewed in color.



Concluding Thoughts

- 3D ConvNets are good at learning spatio-temporal features
- C3D features with linear classifiers are sufficient to outperform or approach current best methods on various video tasks
- C3D features are efficient, compact, and simple to use



Papers

Method 2015

Learning Spatiotemporal Features with 3D Convolutional Networks

Dataset 2015/2016

ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding

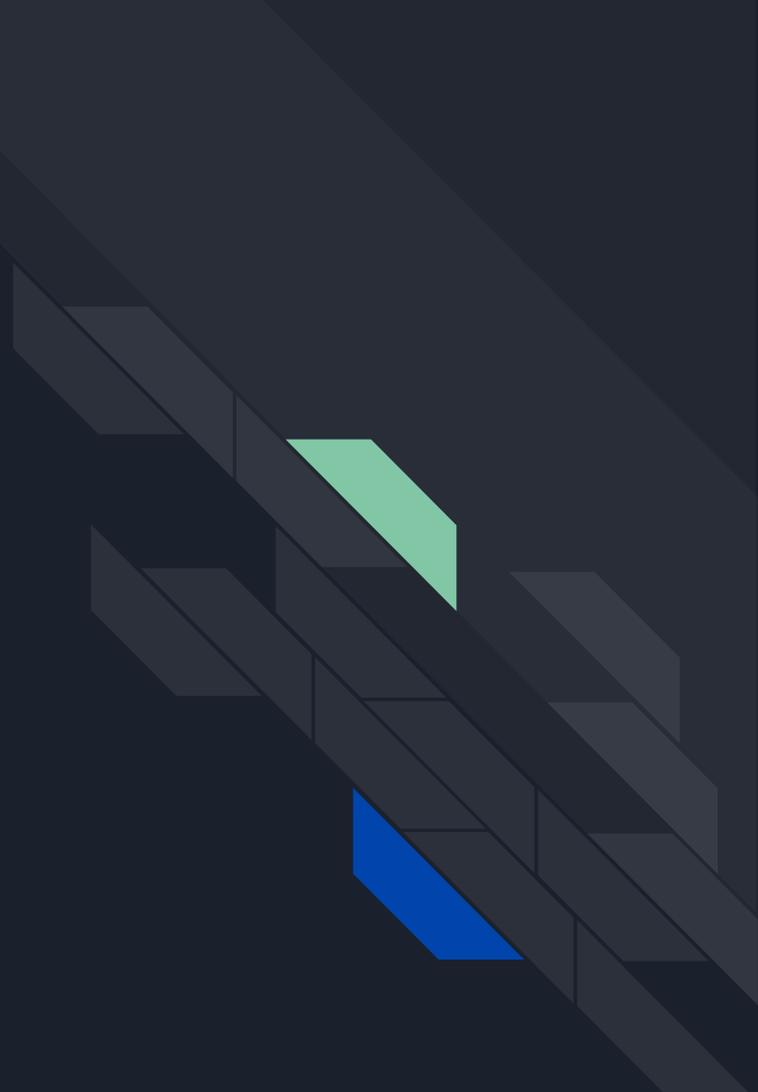
Method 2017

Temporal Activity Detection in Untrimmed Videos with Recurrent Neural Networks

ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding

Fabian Caba Heilbron, Victor Escorcia, Bernard
Ghanem, Juan Carlos Niebles

CVPR 2015





Problem: Existing video datasets and benchmarks are limited and flawed

- Range of day-to-day activities varies a lot
 - e.g. making bed, brushing teeth, etc.
- American Time Use Survey
 - Average American spends **1.7 hours a day on household activities** and only **18 minutes on sports, exercise, or recreation**
- Most video datasets are very specific and not representative of real human day-to-day life



Solution: ActivityNet?

Goals:

- Flexible framework for **continuous acquisition**, **crowdsourced annotation**, and **segmentation**
- Large-scale in **number of categories** and **number of samples**
- Diverse taxonomy and hierarchy (**4+ levels of depth**)
- Easy to use

ActivityNet's rich activity hierarchy



Figure 1. *ActivityNet* organizes a large number of diverse videos that contain human activities into a semantic taxonomy. **Top-row** shows the root-leaf path for the activity *Cleaning windows*. **Bottom-row** shows the root-leaf path for the activity *Brushing teeth*. Each box illustrates example videos that lie within the corresponding taxonomy node. Green intervals indicate the temporal extent of the activity. All figures are best viewed in color.



Existing datasets

- Hollywood
 - 12 action categories, more natural
- UCF Sports, Olympic Sports
 - More challenging, but too specific
- UCF101, HMDB51
 - 50 action categories from YouTube
 - Too short, very simplistic, difficult to scale
 - Taxonomies are too simple
 - HMDB51 organizes into only 5 semantic categories
 - UCF101 also only has 5 types



Existing datasets

- **MPII Human Pose Dataset**
 - Focuses on human pose estimation
 - Clips are too short, distribution per category is biased
- **Sports-1M**
 - Very large!
 - 500 sports-related categories
 - Somewhat limited activity taxonomy, because sports-focused
 - Automatic collection process introduces some noise



ActivityNet to fill the gap in datasets

- Large-scale dataset that covers activities that are most relevant to how humans spend their time day-to-day
- A qualitative jump in terms of number and length of each video
- Diversity of activity taxonomy and number of classes
- A human-in-the-loop annotation process for higher label accuracy
- A framework for low-cost continuous dataset expansion



Building ActivityNet

- Use activity taxonomy created by Department of Labor for use in the **American Time Use Survey (ATUS)**
- **ATUS** has over 2000 activities according to 2 dimensions
 1. Social interactions
 2. Where the activity usually takes place



Building ActivityNet

- ActivityNet selects 203 (out of 2000+) activity subcategories, belonging to 7 (out of 18) different top-level categories
 - Personal care
 - Eating and drinking
 - Household
 - Caring and helping
 - Working
 - Socializing and leisure
 - Sports and exercise



Collecting and annotating activities

1. From list of human activities, search YouTube for related videos with text-based queries
 - Queries are expanded using WordNet hyponyms, hypernyms, and synonyms
2. Verify and label retrieved (untrimmed) videos
 - Use AMT workers to review and check that video has intended activity
3. Temporally annotate ActivityNet instances
 - Use AMT workers to determine temporal extent for each activity label present in the video

Collecting and annotating activities



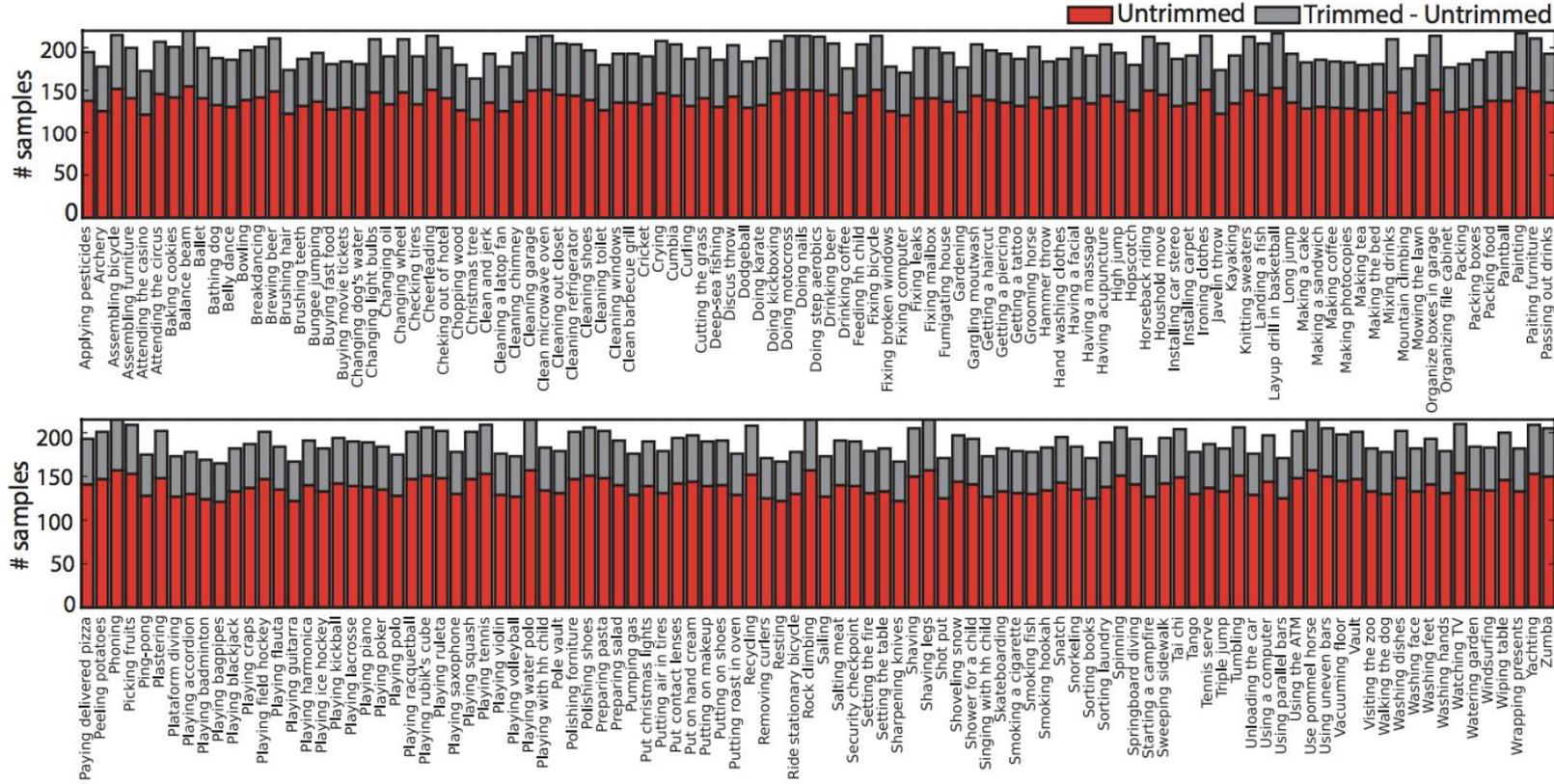
Figure 2. Video collection and annotation process. (a) We start with a large number of candidate videos, for which the labels are partially unknown. (b) AMT workers verify if an activity of interest is present in each video, so that we can discard false positive videos (in red). This results in a set of *untrimmed videos* that contain the activity (in green). (c) Finally, we obtain temporal boundaries for *activity instances* (in green) with the help of AMT workers.



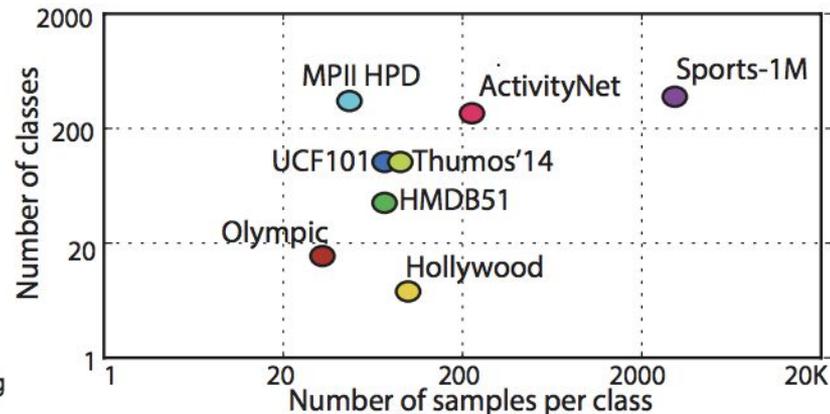
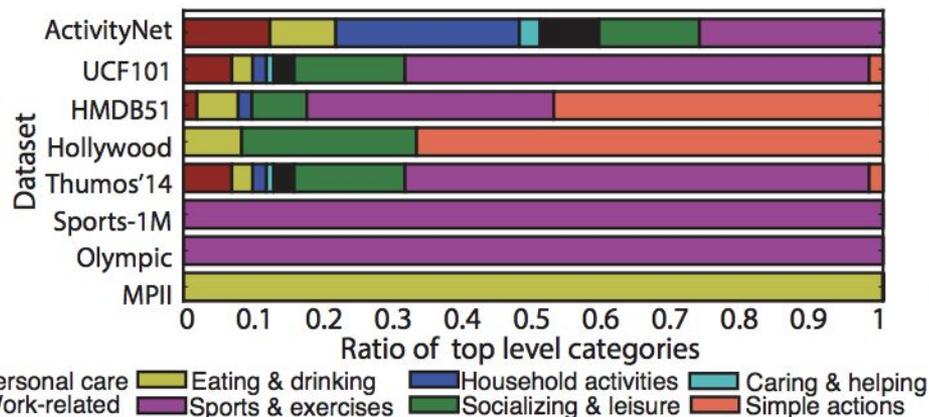
Properties of ActivityNet

- Source videos downloaded at highest quality available
 - 50% of videos are HD (1280 x 720)
- < 20 minutes long
 - Majority are 5-10 minutes
- Majority have 30 FPS
- ~1.41 activity instances per untrimmed video on average
- Activity instance distribution is close to uniform

Properties of ActivityNet



Comparison to existing datasets



ActivityNet strives to include activities in top-level categories that are rarely considered in current benchmarks

Second largest dataset, but most varied in activity types



How can ActivityNet be used?

3 tasks for evaluation

1. Untrimmed video classification
2. Trimmed activity classification
3. Activity detection

Use state-of-the-art action recognition pipeline

- Improved trajectories
- Static and deep features encoded as fisher vectors
- One-vs-all linear SVM classifier



Video representation

For each input video, construct a video representation from 3 feature types:

1. **Motion Features (MF)**
 - Local motion patterns by extracting **improved trajectories**
2. **Static Features (SF)**
 - Textual scene information by extracting **SIFT features**
3. **Deep Features (DF)**
 - Object information using **AlexNet trained on ImageNet for object recognition**



Benchmark 1: Untrimmed video classification

Task

- Predict activities (1+) in untrimmed video sequence

Dataset

- Labeled untrimmed ActivityNet videos
- 27801 videos from 203 activity classes

Classifier

- One-vs-all linear SVM
- Select prediction whose classifier has largest margin

Evaluation

- Measure mean average precision (mAP)



Benchmark 2: Trimmed activity classification

Task

- Predict correct label for video clip with a single activity instance

Dataset

- Trimmed ActivityNet instances
- 203 activity classes, with ~193 samples per class on average

Classifier

- One-vs-all linear SVM
- Select class with highest score

Evaluation

- Measure mean average precision (mAP)



Results: Benchmarks 1+2

Combining multiple features
improves overall performance!

Feature	Untrimmed Classification (mAP)		Trimmed Classification (mAP)	
	Validation	Test	Validation	Test
<i>Motion features (MF)</i>				
HOG	29.2%	28.6%	35.9%	36.1%
HOF	32.7%	31.8%	40.1%	40.2%
MBH	34.1%	33.6%	41.7%	41.9%
<i>Deep features (DF)</i>				
fc-6	28.3%	28.1%	42.7%	43.1%
fc-7	28.0%	27.9%	41.1%	41.6%
fc-8	25.3%	24.9%	38.1%	38.2%
<i>Per feature type</i>				
MF	39.8%	39.2%	47.8%	47.6%
DF	28.9%	28.7%	43.7%	43.0%
SF	24.7%	24.5%	38.3%	37.9%
<i>Combined</i>				
MF+DF	41.2%	40.9%	49.5%	49.1%
MF+SF	40.3%	40.1%	48.9%	48.6%
DF+SF	32.7%	32.6%	44.2%	44.0%
MF+DF+SF	42.5%	42.2%	50.5%	50.2%

Table 1. Summary of classification results. The first two columns report results on the untrimmed video classification task, while the last two report results on trimmed video classification. The evaluation measure is mean average precision (mAP). We report validation and test performance, when different feature combinations are used. MF and DF refers to the concatenation of HOG, HOF and MBH features, and fc-6 and fc-7 respectively.



Benchmark 3: Activity detection

Task

- Find (give temporal extent) and recognize (label) all activity instances in an untrimmed video sequence

Dataset

- ActivityNet
- 849 hours of video, where 68.8 hours contain 203 human-centric activities

Classifier

- Same one-vs-all linear SVMs from trimmed activity classification

Benchmark 3: Activity detection

Evaluation

- Measure mean average precision (mAP) over all classes
- Detection is a true positive if **intersection over union (IoU)** between predicted temporal segment and ground-truth segment exceeds threshold

Feature	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$
MF	11.7%	11.4%	10.6%	9.7%	8.9%
DF	7.2%	6.8%	4.9%	4.1%	3.7%
SF	4.2%	3.9%	3.1%	2.1%	1.9%
MF+DF+SF	12.5%	11.9%	11.1%	10.4%	9.7%

Table 2. Summary of activity detection results. We report the mAP score for all activity classes. Due to the ambiguity inherent to the temporal annotation of activities, we use multiple values for the overlap threshold (α). We also investigate the performance of the different feature types, individually and collectively.



Analysis

- Sports and exercise are the **easiest to classify**
 - Why?
- Household activities are much **harder to classify**
 - Why?
- Activities that take up the entire video (long activities) are the **hardest to classify**
- False positives tend to appear when there are similar motions

Untrimmed Video Classification

Activity	mAP	Correct predictions	Hard false positives	Hard false negatives
Platform diving	63.5%			
Ping-pong	61.1%			
Playing violin	21.4%			
Mixing drinks	17.9%			

Trimmed Activity Classification

Activity	mAP	Correct predictions	Hard false positives	Hard false negatives
Playing guitar	73.9%			
Platform diving	71.1%			
Grooming horse	28.9%			
Mowing the lawn	22.5%			

Figure 5. Example results for the two hardest and easiest activity classes in the untrimmed and trimmed classification tasks. Results are obtained using all three feature types (MF, DF, and SF). The third column shows some correct prediction samples for each class. The last two columns illustrate some hard false positive and hard false negative samples.

Analysis

Category	Validation	Test
Household	34.2%	33.9%
Caring and helping	36.2%	36.7%
Personal care	41.5%	41.3%
Work-related	53.6%	53.1%
Eating and drinking	57.6%	57.2%
Socializing and leisure	63.8%	63.3%
Sports and exercises	66.6%	66.1%
Average	50.5%	50.2%

Table 3. Accuracy analysis on activity classification. We report mAP results for classifying each top-level class in ActivityNet. Here, all three feature types are used: motion, deep and static features.

Analysis

Dataset	Method	Performance
Untrimmed video classification		
Thumos' 14	[14]	71% (mAP)
Sports-1M	[16]	63.9% (mAP)
ActivityNet		42.2% (mAP)
Trimmed activity classification		
UCF101	[40]	85.9% (Accuracy)
HMDB51	[27]	66.7% (Accuracy)
ActivityNet		45.9% (Accuracy)
Activity detection		
Thumos' 14	[25]	33.6% (mAP)
ActivityNet		11.9% (mAP)

Table 4. Cross-dataset performance comparison. State-of-the-art results are reported for each dataset. Reported results for the activity detection task corresponds to the performance obtained with $\alpha = 0.2$



Concluding Thoughts

- **ActivityNet** is a scalable benchmark for human activity understanding
- **ActivityNet** presents more variety in terms of...
 - Activity diversity
 - Richness of taxonomy
 - More categories
 - More samples per category
- **ActivityNet** can be used for...
 - Untrimmed video classification
 - Trimmed activity classification
 - Activity detection
- **ActivityNet** reveals new challenges to overcome!



Papers

Method 2015

Learning Spatiotemporal Features with 3D Convolutional Networks

Dataset 2015/2016

ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding

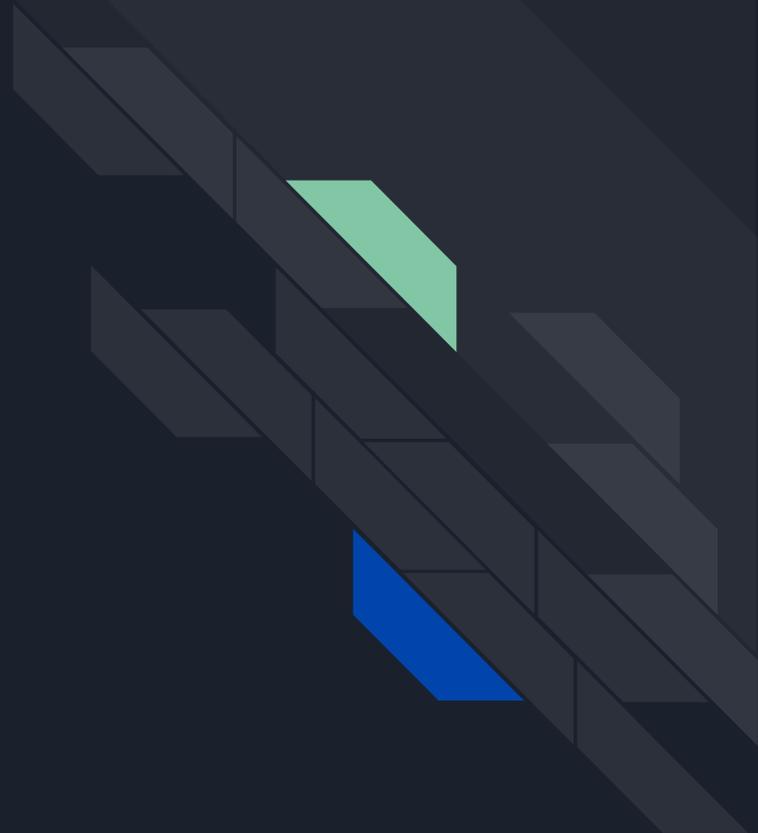
Method 2017

Temporal Activity Detection in Untrimmed Videos with Recurrent Neural Networks

Temporal Activity Detection in Untrimmed Videos with Recurrent Neural Networks

Alberto Montes, Amaia Salvador, Santiago
Pascual, Xacier Giro-i-Nieto

CVPR 2017





The state of video activity recognition (~2017)

ActivityNet Challenge 2016

- A video classification system should be able to...
 - recognize activities in untrimmed videos, and
 - provide their temporal segments

Recent work

- [Paper 1] 3D ConvNets (C3D) have been used for video classification and temporal detection
- [Not covered... yet] LSTMs have also been used for video classification and activity localization

Idea: Let's feed C3D features into a RNN

- Each input video clip is 16-frames long
- Pass 4096-dim fc6 features from C3D as inputs into RNN
- RNN has
 - Dropout of $p = 0.5$
 - FC layer with softmax activation
- Varied configurations
 - # of LSTM layers N
 - # of cells c

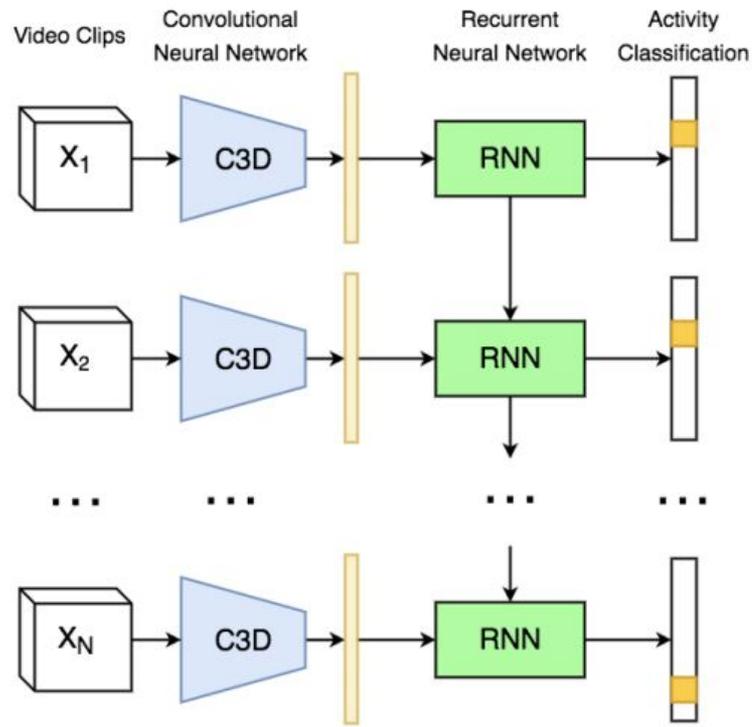


Figure 1: Global architecture of the proposed pipeline.



Post-processing

- Model outputs sequence of class probabilities for each 16-frame video clip
- **Activity prediction** for whole video
 - Average class probabilities from each 16-frame clip
 - **Prediction** = class with maximum predicted probability
- **Temporal localization** of activity
 - Apply mean filter of k samples to predicted sequences

$$\tilde{p}_i(x) = \frac{1}{2k} \sum_{j=i-k}^{i+k} p_j(x)$$

- Predict probability of *activity* vs. *no activity* for each clip
- Assign predicted activity class to clips with *activity* probability above some threshold



Training

Dataset

- ActivityNet Challenge 2016
- 640 hours of video, 64 million frames
- Untrimmed video w/ temporal annotations for ground-truth

Training: negative log-likelihood loss

- q : predicted probability distribution
- p : ground truth probability distribution
- $\rho = 0.3$ (to weight background samples less)

$$L(p, q) = - \sum_x \alpha(x) p(x) \log(q(x)), \text{ where } \alpha(x) = \begin{cases} \rho, & x = \text{background instance} \\ 1, & \text{otherwise} \end{cases}$$



Evaluation & Results

Metrics

- Mean average precision (mAP)
- Hit@3

Prediction marked as correct if...

1. Correct category label
2. IoU with ground truth larger than 0.5

Evaluation & Results

Architecture	mAP	Hit@3
3 x 1024-LSTM	0.5635	0.7437
2 x 512-LSTM	0.5492	0.7364
1 x 512-LSTM	0.5938	0.7576

Table 1: Results for classification task comparing different architectures.

γ	$k = 0$	$k = 5$	$k = 10$
0.2	0.20732	0.22513	0.22136
0.3	0.19854	0.22077	0.22100
0.5	0.19035	0.21937	0.21302

Table 2: mAP with an IoU threshold of 0.5 comparing between values of k and γ on post-processing.

Best results with
single-layer 512-LSTM cells
Overfitting otherwise!

Temporal localization hyperparameters for mean filter:
 γ = threshold for *activity vs. no activity*
 k = # samples to smooth over

Mean smoothing filter improves performance!

Evaluation & Results

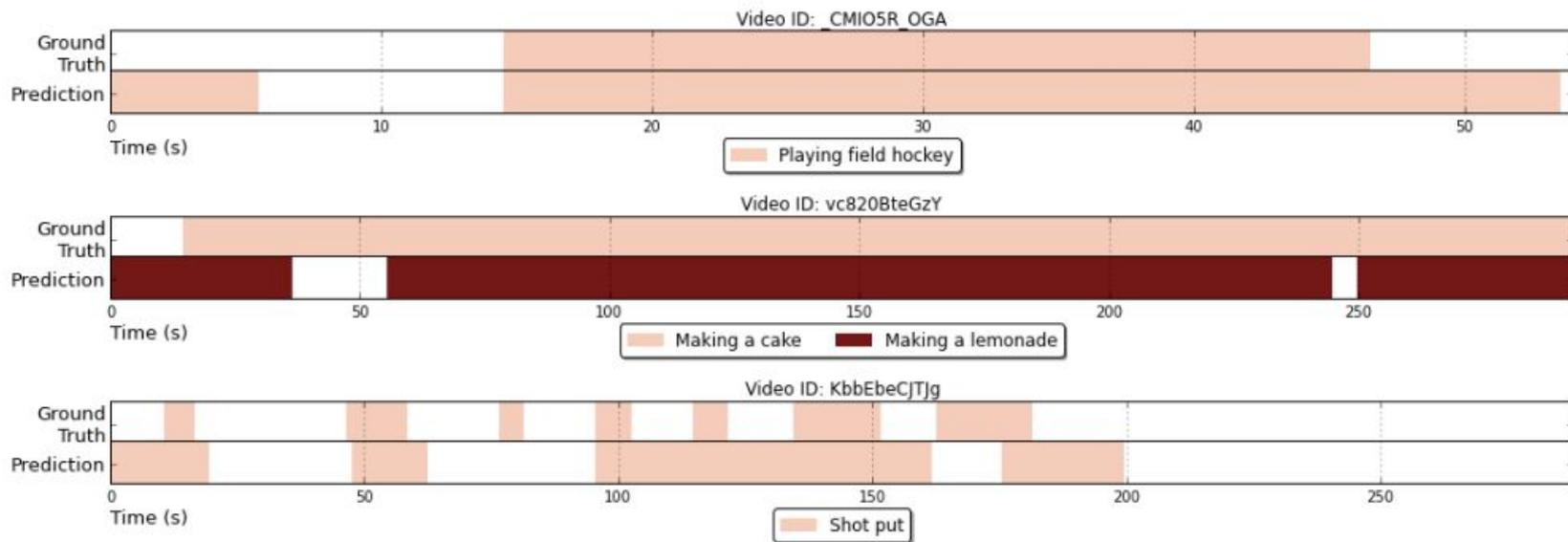


Figure 2: Examples of temporal activity localization predictions.

Evaluation & Results



Video ID: ArzhjEk4j_Y
Ground Truth: Building sandcastles

Prediction:
0.7896 Building sandcastles
0.0073 Doing motocross
0.0049 Beach soccer



Video ID: AimG8xzchfI
Activity: Curling

Prediction:
0.3843 Shoveling snow
0.1181 Ice fishing
0.0633 Waterskiing

Figure 3: Examples of activity classification.



Concluding Thoughts

- Simple pipeline combining C3D fc6 features with RNN provides competitive (?) results
- Flexibility to be extended further to more challenging tasks
- Future work
 - End-to-end training of 3D ConvNet + RNN model
 - Learn better feature representations?



Papers

Method 2015

Learning Spatiotemporal Features with 3D Convolutional Networks

Dataset 2015/2016

ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding

Method 2017

Temporal Activity Detection in Untrimmed Videos with Recurrent Neural Networks

2017?

2018?



Where are we now?

- We can use **3D ConvNets** to extract spatio-temporal features from videos
- We have a large, scalable dataset and benchmark in **ActivityNet**
- We can extend **3D ConvNets** with **recurrent neural networks** to achieve competitive performance
 - Sets up stage for more work with **RNNs** and more complicated models

Next time...

- We will see two “simple” models for temporal action localization from 2017!



From Classification to Temporal Localization with 3D Convolutions

Austin Le

April 16th, 2018

