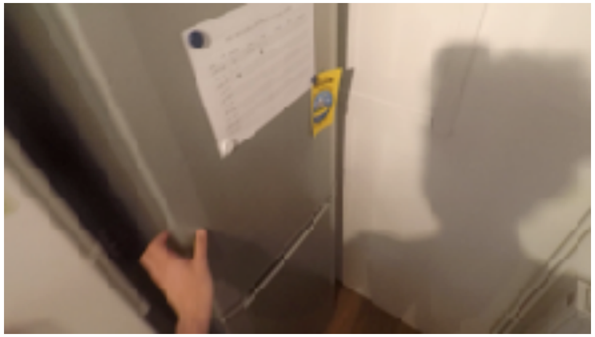
Problem Overview

Given a video, we want to classify it into one of the human action categories.

Cartwheeling

Braiding Hair



Opening a Fridge

Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset

CVPR 2017

Joao Carreira, Andrew Zisserman

Motivation #1

Imagenet benchmark has been essential for progress in image modeling over the last decade or so.

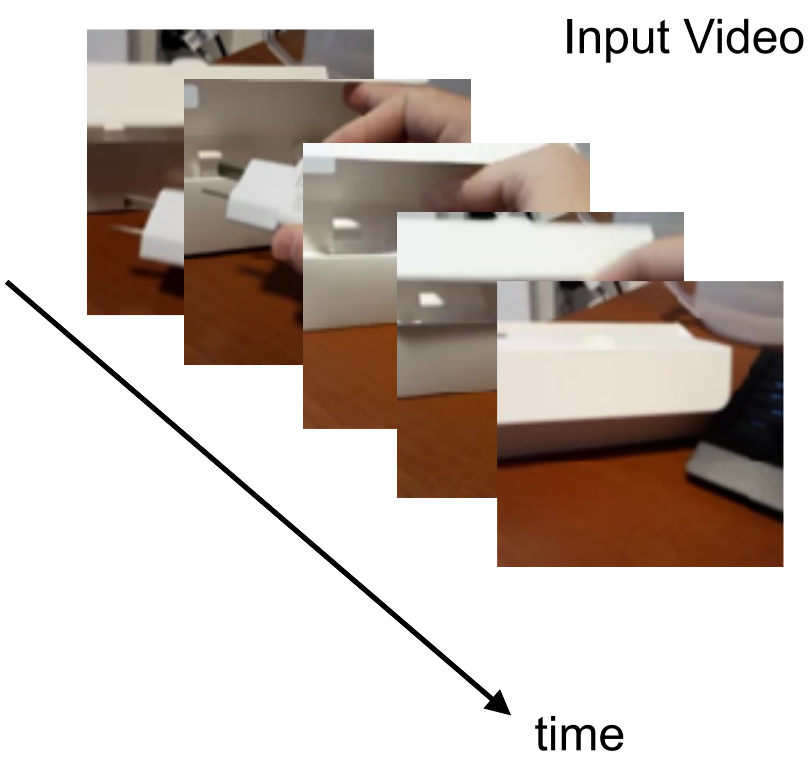
Motivation #1

Imagenet benchmark has been essential for progress in image modeling over the last decade or so.

Can large-scale video datasets be useful for video?

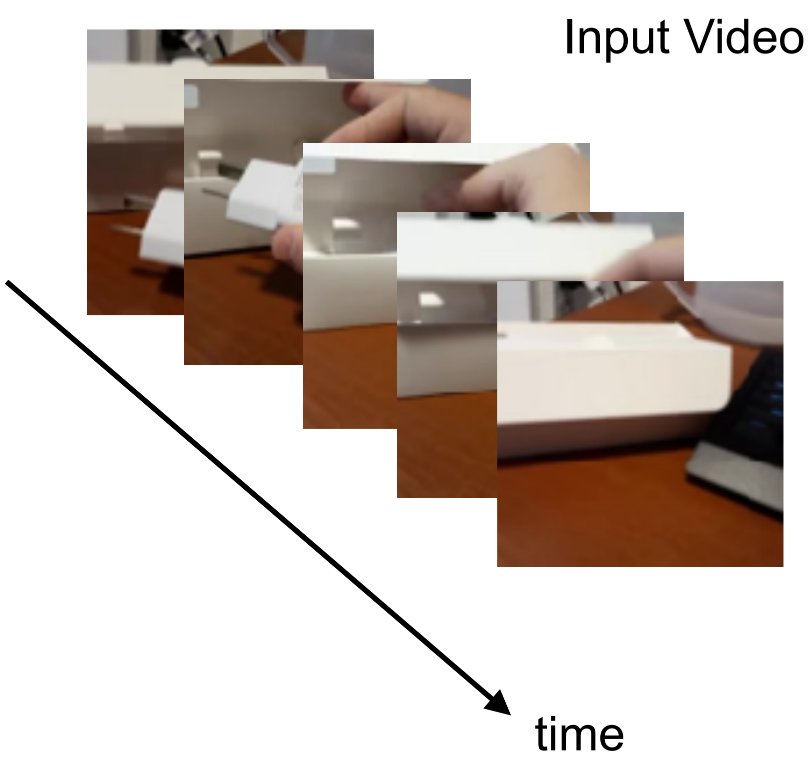
Motivation #2

A video can be viewed as a collection of images.



Motivation #2

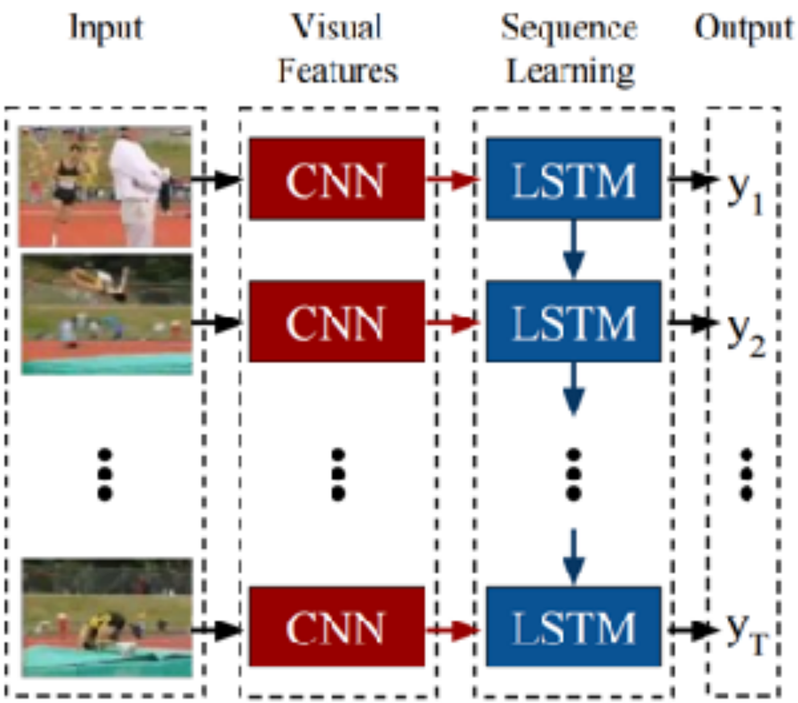
A video can be viewed as a collection of images.



How can we use pretrained image models for spatiotemporal feature learning?

Main Technical Challenge

Adapting 2D CNNs pretrained on Imagenet to video is not trivial.

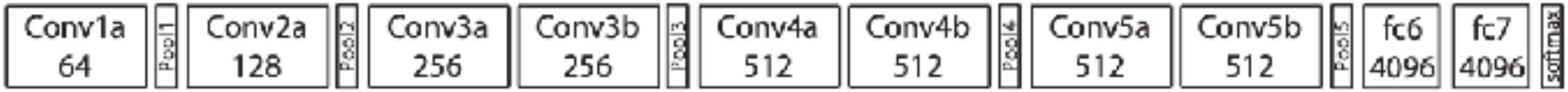


Not very effective!

"Long-term Recurrent Convolutional Networks for Visual Recognition and Description", CVPR 2015

Main Technical Challenge

Adapting 2D CNNs pretrained on Imagenet to video is not trivial.

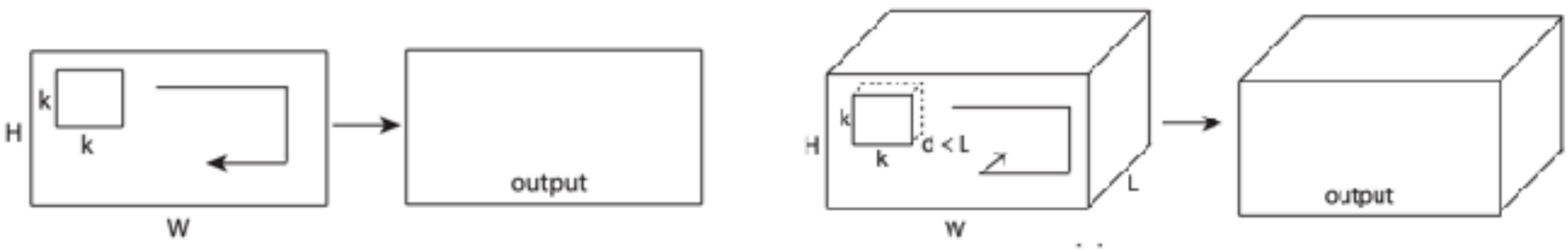


Trained from scratch, which is very costly.

“Learning Spatiotemporal Features with 3D Convolutional Networks“, ICCV 2015

Main Technical Challenge

Adapting 2D CNNs pretrained on Imagenet to video is not trivial.



a) 2D convolution

b) 3D convolution

How can we extend pretrained 2D convolutional weights to 3D for video processing?

Training 3D CNNs on Imagenet

One could train a 3D CNN on Imagenet on the stacked copies of an input image.

Stacked Copies
of an Input Image

3D CNN

Output:

A Penguin

Inflated 3D CNNs

We want to transform a pretrained 2D CNN into an equivalent 3D CNN that re-uses the learned Imagenet features.

Input: A pretrained 2D CNN $\xrightarrow{\text{Inflation from 2D to 3D}}$ **Output:** An equivalent 3D CNN

Inflated 3D CNNs

The paper propose to inflate all pretrained 2D filters to 3D.

$$f = \begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline -5 & 6 & 1 \\ \hline 2 & -2 & -4 \\ \hline \end{array}$$

a 2D grid (e.g., an image)

$$g = \begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & -1 & -2 \\ \hline 1 & 2 & -1 \\ \hline \end{array}$$

2D convolutional filter

$$h = g * f = \boxed{-8}$$

Inflated 3D CNNs

The paper propose to inflate all pretrained 2D filters to 3D.

$$f = \begin{array}{cc} \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t-1} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t+1} \end{array}$$

a 3D grid (e.g., a video clip)

$$g = \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix}$$

2D convolutional filter

$$h = g * f = \begin{array}{cc} \boxed{-8} & \text{time t-1} \\ \boxed{-8} & \text{time t} \\ \boxed{-8} & \text{time t+1} \end{array}$$

Inflated 3D CNNs

The paper propose to inflate all pretrained 2D filters to 3D.

$$f = \begin{array}{cc} \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t-1} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t+1} \end{array}$$

a 3D grid (e.g., a video clip)

$$g = \begin{array}{cc} \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} & \text{time t-1} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} & \text{time t} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} & \text{time t+1} \end{array}$$

3D convolutional filter

$$h = g * f = \boxed{-24}$$

Inflated 3D CNNs

The paper propose to inflate all pretrained 2D filters to 3D.

$$f = \begin{array}{cc} \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t-1} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t+1} \end{array} \qquad g = \begin{array}{cc} \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} / 3 & \text{time t-1} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} / 3 & \text{time t} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} / 3 & \text{time t+1} \end{array} \qquad h = g * f = \boxed{-8}$$

a 3D grid (e.g., a video clip)

3D convolutional filter

Inflated 3D CNNs

The paper propose to inflate all pretrained 2D filters to 3D.

$$f = \begin{array}{cc} \begin{bmatrix} 1 & 2 & 3 \\ -5 & 4 & 1 \\ 3 & -1 & -4 \end{bmatrix} & \text{time t-1} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 1 \\ 2 & -2 & -4 \end{bmatrix} & \text{time t} \\ \begin{bmatrix} 1 & 2 & 3 \\ -5 & 6 & 0 \\ 2 & -3 & -2 \end{bmatrix} & \text{time t+1} \end{array} \qquad g = \begin{array}{cc} \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} / 3 & \text{time t-1} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} / 3 & \text{time t} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & 2 & -1 \end{bmatrix} / 3 & \text{time t+1} \end{array} \qquad h = g * f \approx \boxed{-8}$$

a 3D grid (e.g., a video clip)

3D convolutional filter

Inflated 3D CNNs

The paper propose to inflate all pretrained 2D filters to 3D.

$$f = \begin{array}{cl} \begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline -5 & 6 & 1 \\ \hline 2 & -2 & -4 \\ \hline \end{array} & \text{time t-1} \\ \\ \begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline -5 & 6 & 1 \\ \hline 2 & -2 & -4 \\ \hline \end{array} & \text{time t} \\ \\ \begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline -5 & 6 & 1 \\ \hline 2 & -2 & -4 \\ \hline \end{array} & \text{time t+1} \end{array}$$

a 3D grid (e.g., a video clip)

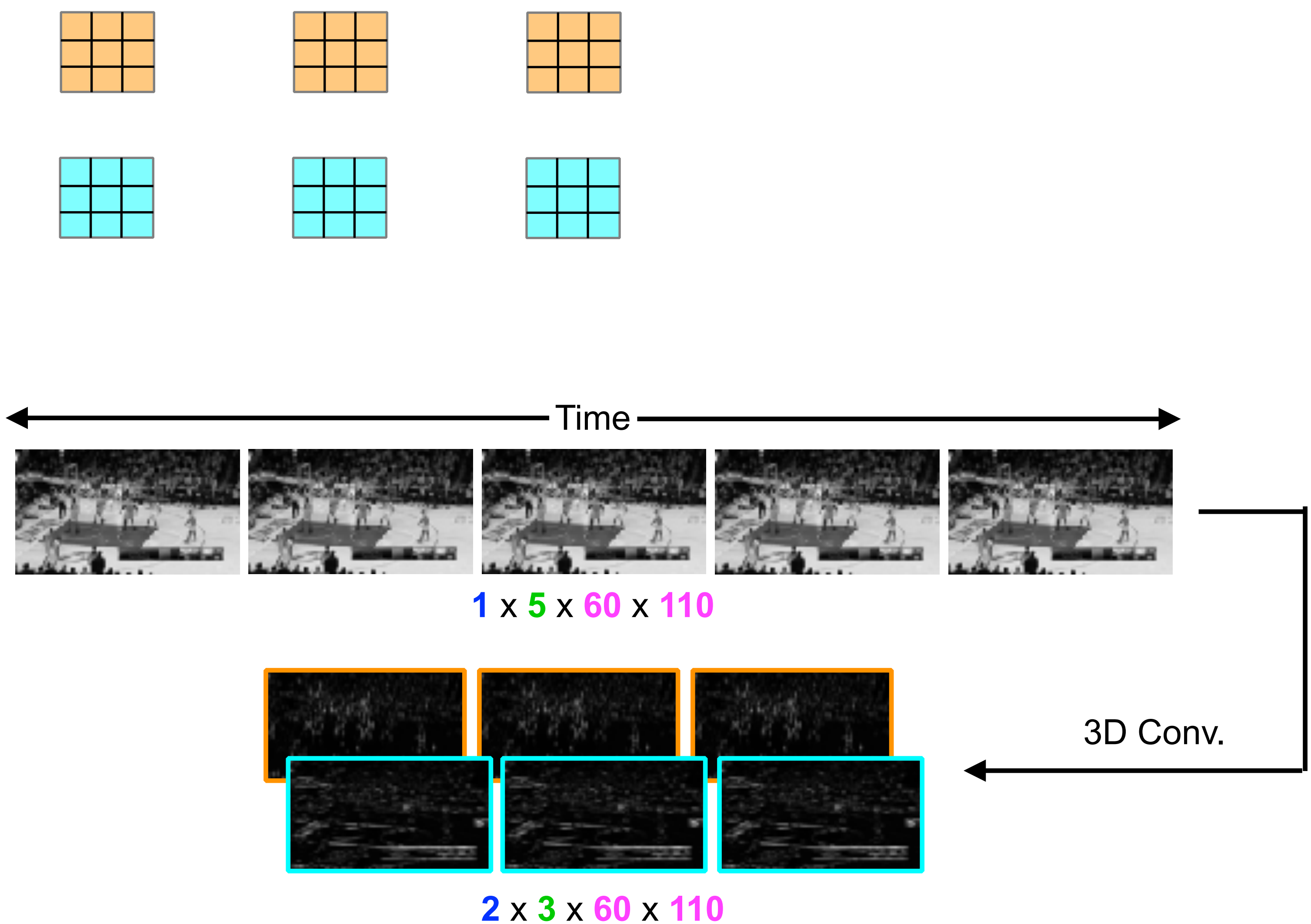
$$g = \begin{array}{cl} \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} & \text{time t-1} \\ \\ \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & -1 & -2 \\ \hline 1 & 2 & -1 \\ \hline \end{array} & \text{time t} \\ \\ \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} & \text{time t+1} \end{array}$$

3D convolutional filter

$$h = g * f = \boxed{-8}$$

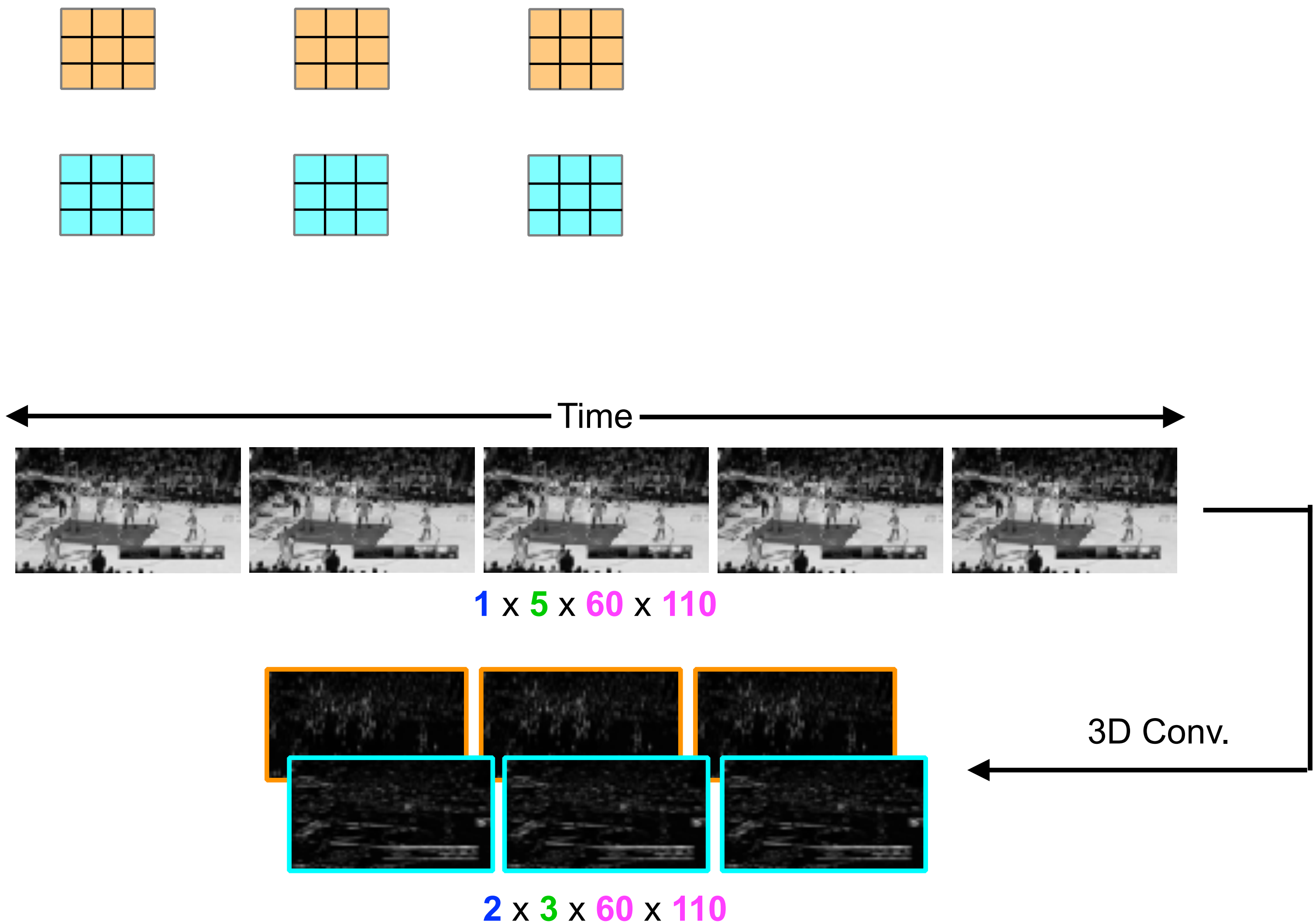
3D Convolution

Learnable 3 x 3 x 3 Convolutional Kernels (**Temporal**, **Spatial**)



3D Convolution

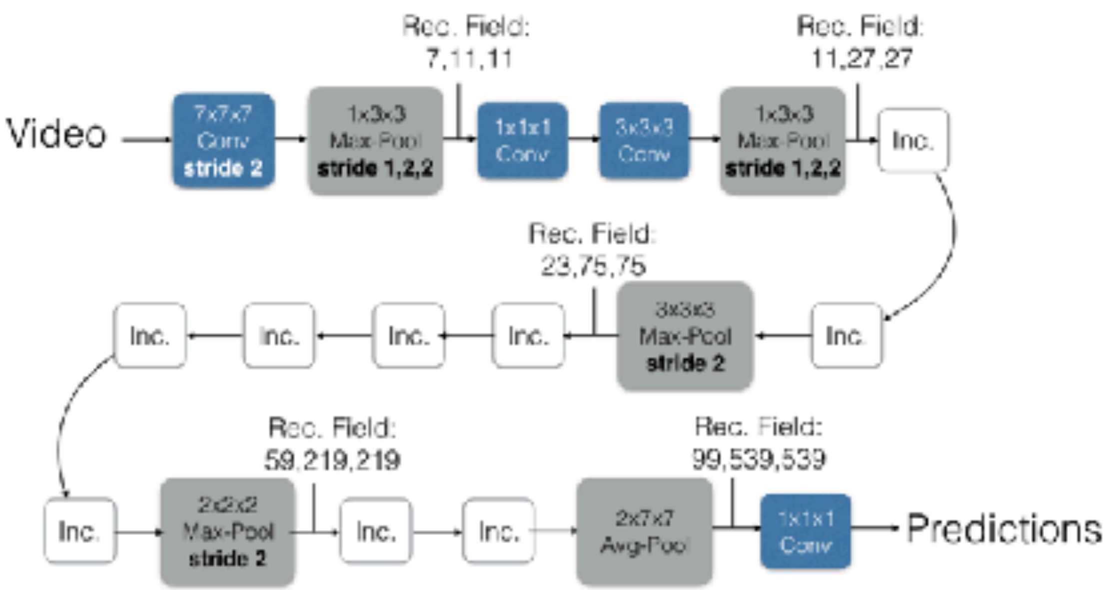
Learnable 3 x 3 x 3 Convolutional Kernels (Temporal, Spatial)



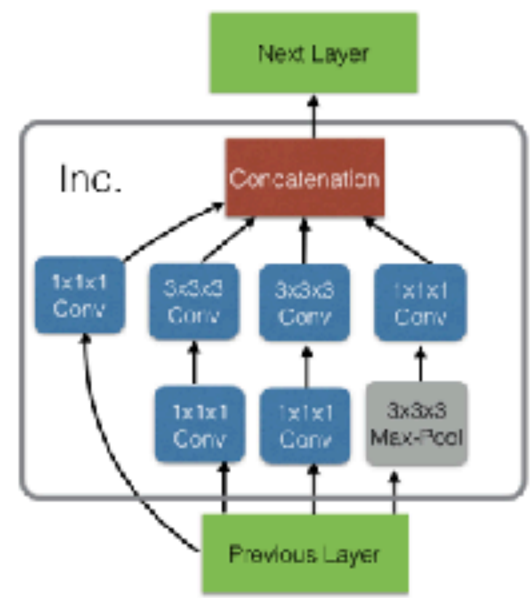
Inflated 3D CNNs

The Inflated Inception-V1 architecture (left) and its detailed inception submodule (right).

Inflated Inception-V1



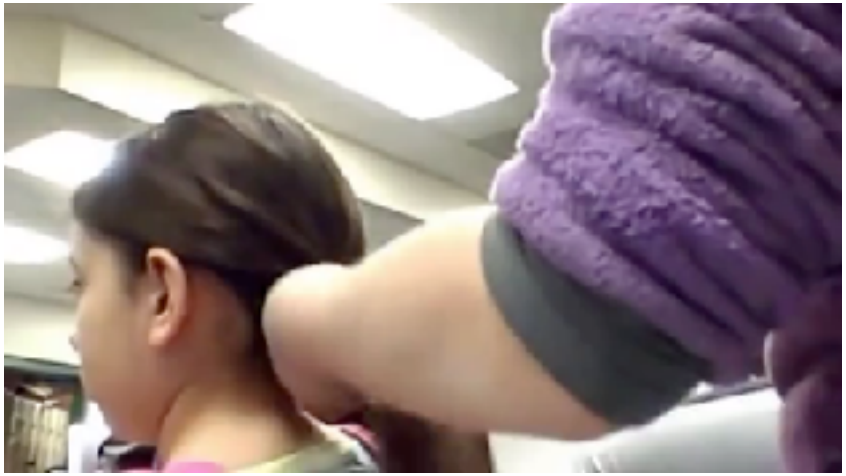
Inception Module (Inc.)



Kinetics Dataset

- ~240K YouTube videos manually annotated with 400 human action classes.
- The clips last around 10s.
- Introduced in this same paper together with I3D architecture.

Cartwheeling



Braiding Hair

Importance of Imagenet Pretraining

Comparison with and without ImageNet pretraining.

Architecture	Kinetics			ImageNet then Kinetics		
	RGB	Flow	RGB + Flow	RGB	Flow	RGB + Flow
(a) LSTM	53.9	–	–	63.3	–	–
(b) 3D-ConvNet	56.1	–	–	–	–	–
(c) Two-Stream	57.9	49.6	62.8	62.2	52.4	65.6
(d) 3D-Fused	–	–	62.7	–	–	67.2
(e) Two-Stream I3D	**68.4** (88.0)	**61.5** (83.4)	**71.6** (90.0)	**71.1** (89.3)	**63.4** (84.9)	**74.2** (91.3)

Importance of Imagenet Pretraining

Comparison with and without ImageNet pretraining.

Architecture	Kinetics			ImageNet then Kinetics		
	RGB	Flow	RGB + Flow	RGB	Flow	RGB + Flow
(a) LSTM	53.9	–	–	63.3	–	–
(b) 3D-ConvNet	56.1	–	–	–	–	–
(c) Two-Stream	57.9	49.6	62.8	62.2	52.4	65.6
(d) 3D-Fused	–	–	62.7	–	–	67.2
(e) Two-Stream I3D	**68.4** (88.0)	**61.5** (83.4)	**71.6** (90.0)	**71.1** (89.3)	**63.4** (84.9)	**74.2** (91.3)

Kinetics video pretraining is complementary to Imagenet image pretraining.

Comparison to the State-of-the-Art

Comparison to all prior action recognition methods on UCF-101 and HMDB-51.

Model	UCF-101	HMDB-51
Two-Stream [27]	88.0	59.4
IDT [33]	86.4	61.7
Dynamic Image Networks + IDT [2]	89.1	65.2
TDD + IDT [34]	91.5	65.9
Two-Stream Fusion + IDT [8]	93.5	69.2
Temporal Segment Networks [35]	94.2	69.4
ST-ResNet + IDT [7]	94.6	70.3
Deep Networks [15], Sports 1M pre-training	65.2	-
C3D one network [31], Sports 1M pre-training	82.3	-
C3D ensemble [31], Sports 1M pre-training	85.2	-
C3D ensemble + IDT [31], Sports 1M pre-training	90.1	-
RGB-I3D, Imagenet+Kinetics pre-training	95.6	74.8
Flow-I3D, Imagenet+Kinetics pre-training	96.7	77.1
Two-Stream I3D, Imagenet+Kinetics pre-training	**98.0**	80.7
RGB-I3D, Kinetics pre-training	95.1	74.3
Flow-I3D, Kinetics pre-training	96.5	77.3
Two-Stream I3D, Kinetics pre-training	97.8	**80.9**

Two-stream I3D achieves best performance on both datasets.