Lecture 9

Video Recognition, Optical Flow

Slides from: Du Tran, Rick Szeliski, Steve Seitz, Christoph Feichtenhofer

Overview

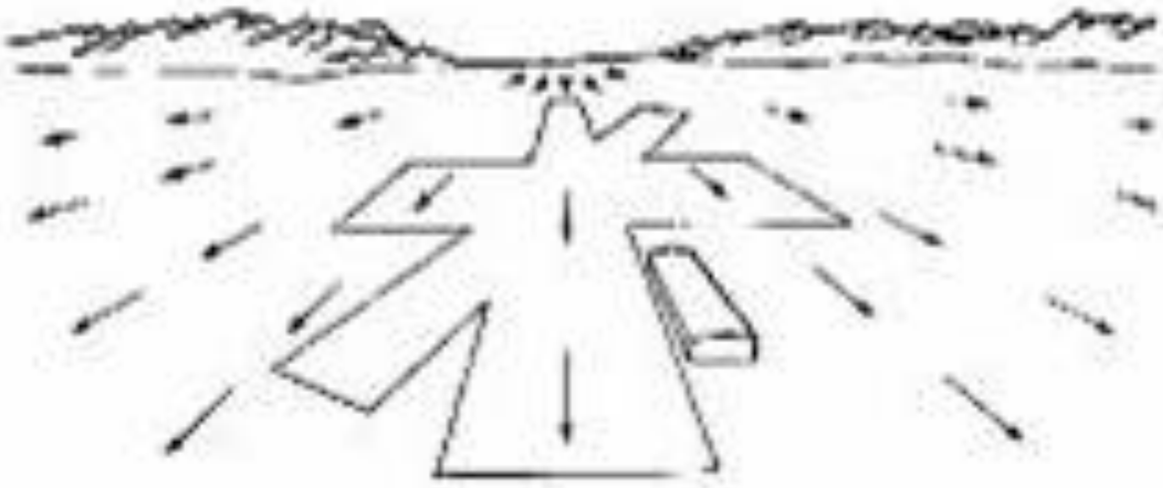
- Optical Flow
- ConvNets for Video

Overview

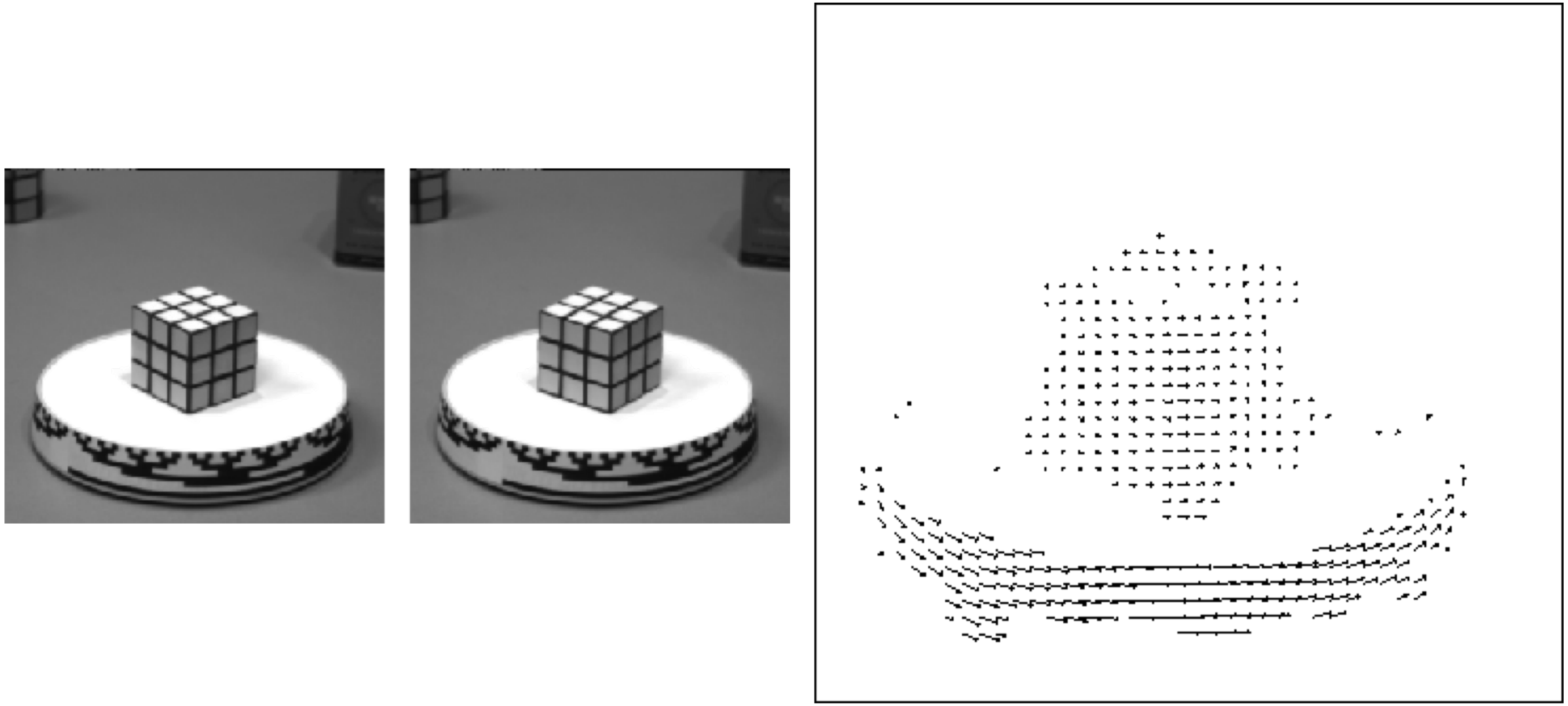
- Optical Flow
- ConvNets for Video

Optical flow

Combination of slides from Rick Szeliski, Steve Seitz, Alyosha Efros and Bill Freeman and Fredo Durand



Motion estimation: Optical flow



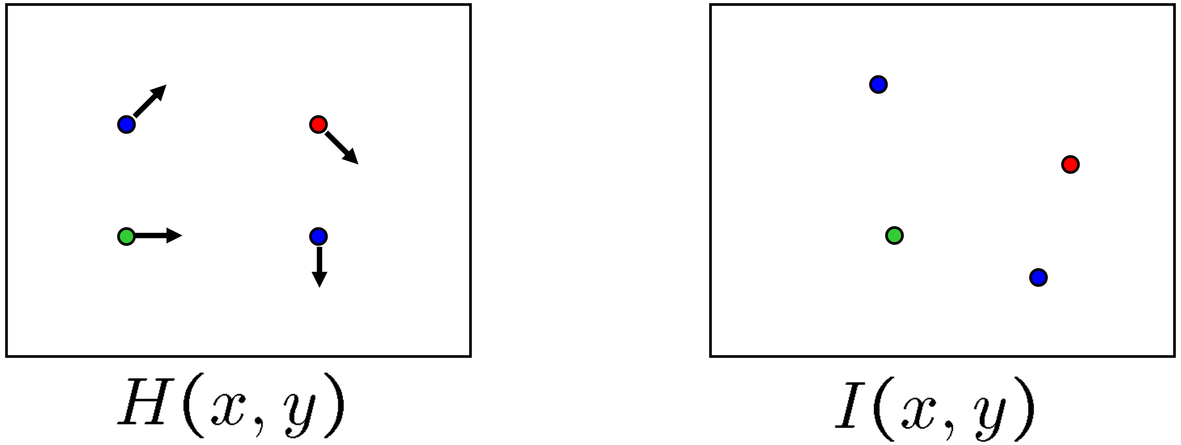
Will start by estimating motion of each pixel separately
Then will consider motion of entire image

Why estimate motion?

Lots of uses

- Feature representation for DeepNets [coming up]
- Track object behavior
- Correct for camera jitter (stabilization)
- Align images (mosaics)
- 3D shape reconstruction
- Special effects

Problem definition: optical flow



How to estimate pixel motion from image H to image I?

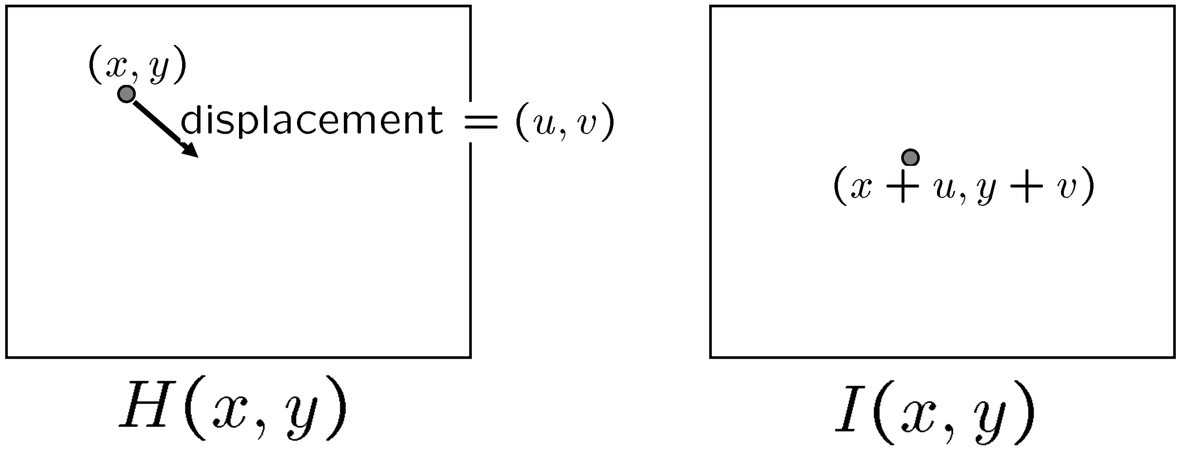
- Solve pixel correspondence problem
  - given a pixel in H, look for nearby pixels of the same color in I

Key assumptions

- color constancy: a point in H looks the same in I
  - For grayscale images, this is brightness constancy
- small motion: points do not move very far

This is called the optical flow problem

Optical flow constraints (grayscale images)



Let's look at these constraints more closely

- brightness constancy: Q: what's the equation?

  H(x,y)=I(x+u, y+v)

- small motion: (u and v are less than 1 pixel)
  - suppose we take the Taylor series expansion of I:

$$I(x+u, y+v) = I(x, y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \text{higher order terms}$$
$$\approx I(x, y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v$$

Optical flow equation

Combining these two equations

shorthand: $I_x = \frac{\partial I}{\partial x}$

$$\begin{aligned} 0 &= I(x+u, y+v) - H(x,y) \\ &\approx I(x,y) + I_x u + I_y v - H(x,y) \\ &\approx (I(x,y) - H(x,y)) + I_x u + I_y v \\ &\approx I_t + I_x u + I_y v \\ &\approx I_t + \nabla I \cdot [u \;\; v] \end{aligned}$$

In the limit as u and v go to zero, this becomes exact

$$0 = I_t + \nabla I \cdot [\tfrac{\partial x}{\partial t} \;\; \tfrac{\partial y}{\partial t}]$$

Optical flow equation

$$0 = I_t + \nabla I \cdot [u \;\; v]$$

Q: how many unknowns and equations per pixel?

2 unknowns, one equation

Intuitively, what does this constraint mean?

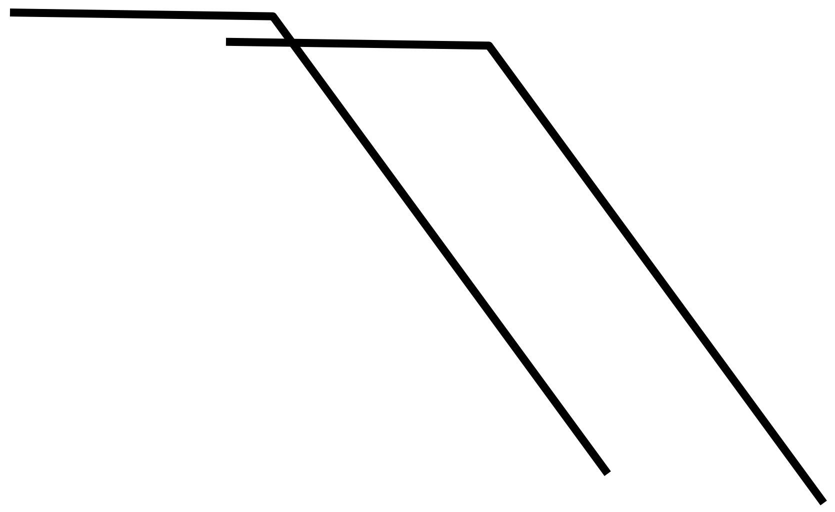
- The component of the flow in the gradient direction is determined
- The component of the flow parallel to an edge is unknown

This explains the Barber Pole illusion

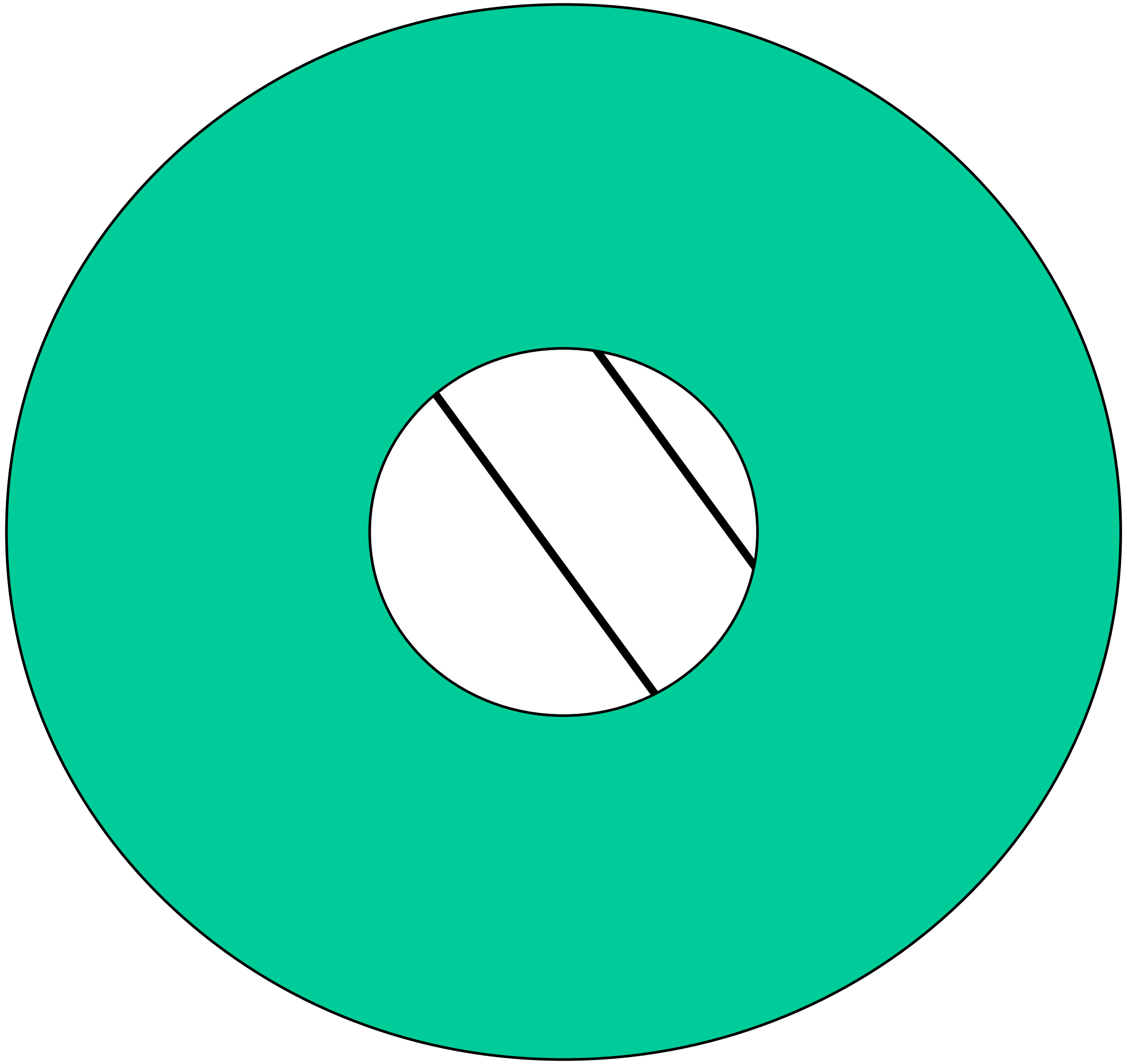
http://www.sandlotscience.com/Ambiguous/Barberpole_Illusion.htm
http://www.liv.ac.uk/~marcob/Trieste/barberpole.html

http://en.wikipedia.org/wiki/Barber's_pole

Aperture problem



Aperture problem



Solving the aperture problem

How to get more equations for a pixel?

- Basic idea: impose additional constraints
  - most common is to assume that the flow field is smooth locally
  - one method: pretend the pixel's neighbors have the same (u,v)
    - » If we use a 5x5 window, that gives us 25 equations per pixel!

$$0 = I_t(\mathbf{p_i}) + \nabla I(\mathbf{p_i}) \cdot [u \ \ v]$$

$$\underset{25\times 2}{\underbrace{\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix}}_{A}} \underset{2\times 1}{\underbrace{\begin{bmatrix} u \\ v \end{bmatrix}}_{d}} = - \underset{25\times 1}{\underbrace{\begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix}}_{b}}$$

RGB version

How to get more equations for a pixel?

- Basic idea: impose additional constraints
  - most common is to assume that the flow field is smooth locally
  - one method: pretend the pixel's neighbors have the same (u,v)
    - » If we use a 5x5 window, that gives us 25*3 equations per pixel!

$$0 = I_t(\mathbf{p_i})[0,1,2] + \nabla I(\mathbf{p_i})[0,1,2] \cdot [u \ v]$$

$$\begin{bmatrix} I_x(\mathbf{p}_1)[0] & I_y(\mathbf{p}_1)[0] \\ I_x(\mathbf{p}_1)[1] & I_y(\mathbf{p}_1)[1] \\ I_x(\mathbf{p}_1)[2] & I_y(\mathbf{p}_1)[2] \\ \vdots & \vdots \\ I_x(\mathbf{p}_{25})[0] & I_y(\mathbf{p}_{25})[0] \\ I_x(\mathbf{p}_{25})[1] & I_y(\mathbf{p}_{25})[1] \\ I_x(\mathbf{p}_{25})[2] & I_y(\mathbf{p}_{25})[2] \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p}_1)[0] \\ I_t(\mathbf{p}_1)[1] \\ I_t(\mathbf{p}_1)[2] \\ \vdots \\ I_t(\mathbf{p}_{25})[0] \\ I_t(\mathbf{p}_{25})[1] \\ I_t(\mathbf{p}_{25})[2] \end{bmatrix}$$

$$\underset{75\times 2}{A} \qquad \underset{2\times 1}{d} \qquad \underset{75\times 1}{b}$$

Note that RGB is not enough to disambiguate
because R, G & B are correlated
Just provides better gradient

Lukas-Kanade flow

Prob: we have more equations than unknowns

$$\underset{25\times 2}{A}\ \underset{2\times 1}{d} = \underset{25\times 1}{b} \longrightarrow \text{minimize } \|Ad - b\|^2$$

Solution: solve least squares problem

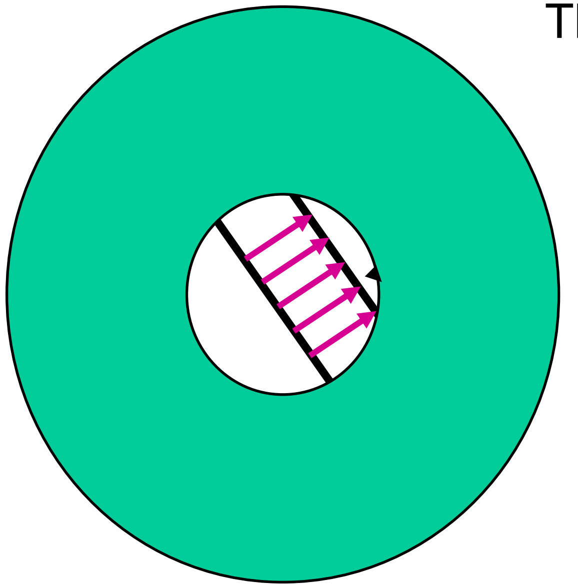
- minimum least squares solution given by solution (in d) of:

$$\underset{2\times 2}{(A^T A)}\ \underset{2\times 1}{d} = \underset{2\times 1}{A^T b}$$

$$\underbrace{\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix}}_{A^T A} \begin{bmatrix} u \\ v \end{bmatrix} = - \underbrace{\begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}}_{A^T b}$$

- The summations are over all pixels in the K x K window
- This technique was first proposed by Lukas & Kanade (1981)

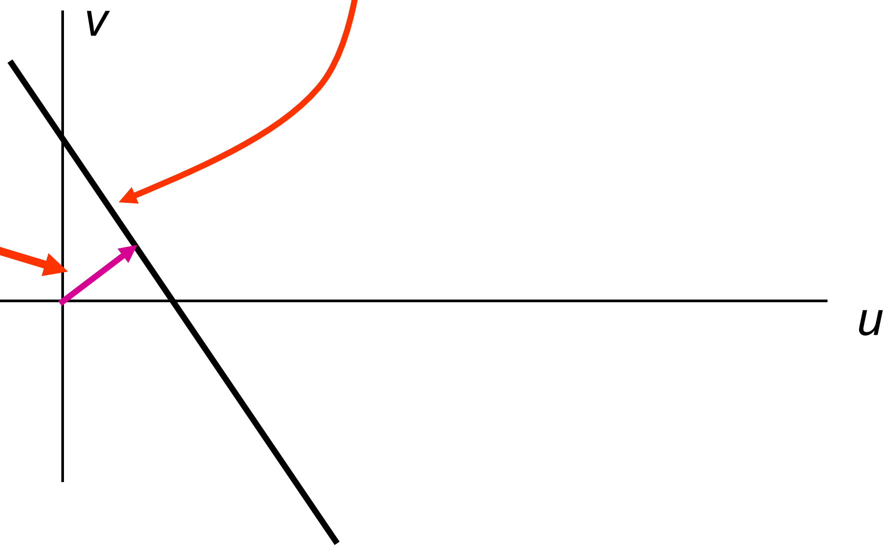
Aperture Problem and Normal Flow



The gradient constraint:

$$\boxed{\begin{aligned} I_x u + I_y v + I_t &= 0 \\ \nabla I \bullet \vec{U} &= 0 \end{aligned}}$$

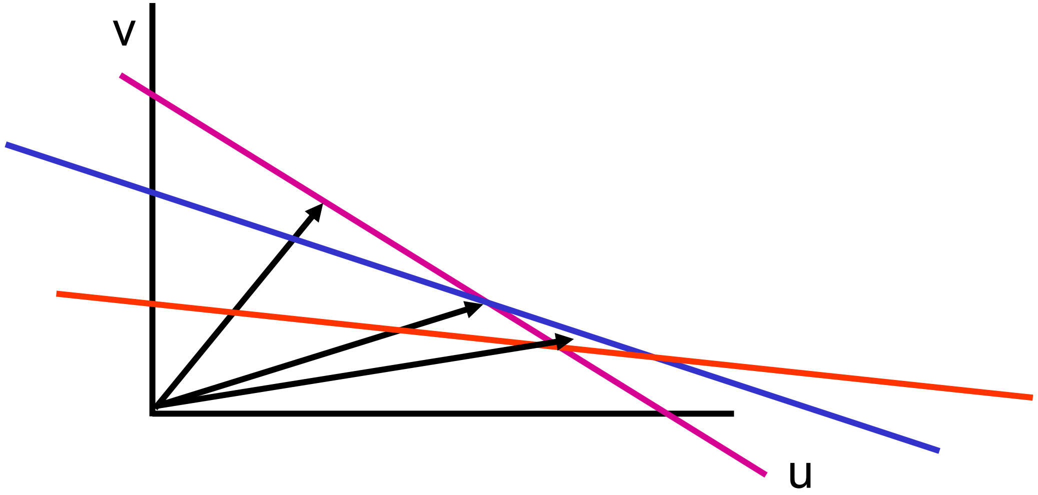
Defines a line in the *(u,v)* space



Normal Flow:

$$\boxed{u_\perp = -\frac{I_t}{|\nabla I|}\frac{\nabla I}{|\nabla I|}}$$

Combining Local Constraints



$$\nabla I^1 \bullet U = -I_t^1$$

$$\nabla I^2 \bullet U = -I_t^2$$

$$\nabla I^3 \bullet U = -I_t^3$$

etc.

Conditions for solvability

- Optimal (u, v) satisfies Lucas-Kanade equation

$$\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}$$

$$A^T A \qquad\qquad\qquad\qquad A^T b$$

When is This Solvable?

- $A^T A$ should be invertible
- $A^T A$ should not be too small due to noise
  - eigenvalues λ_1 and λ_2 of $A^T A$ should not be too small
- $A^T A$ should be well-conditioned
  - λ_1 / λ_2 should not be too large (λ_1 = larger eigenvalue)

$A^T A$ is solvable when there is no aperture problem

$$A^T A = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \ I_y] = \sum \nabla I (\nabla I)^T$$

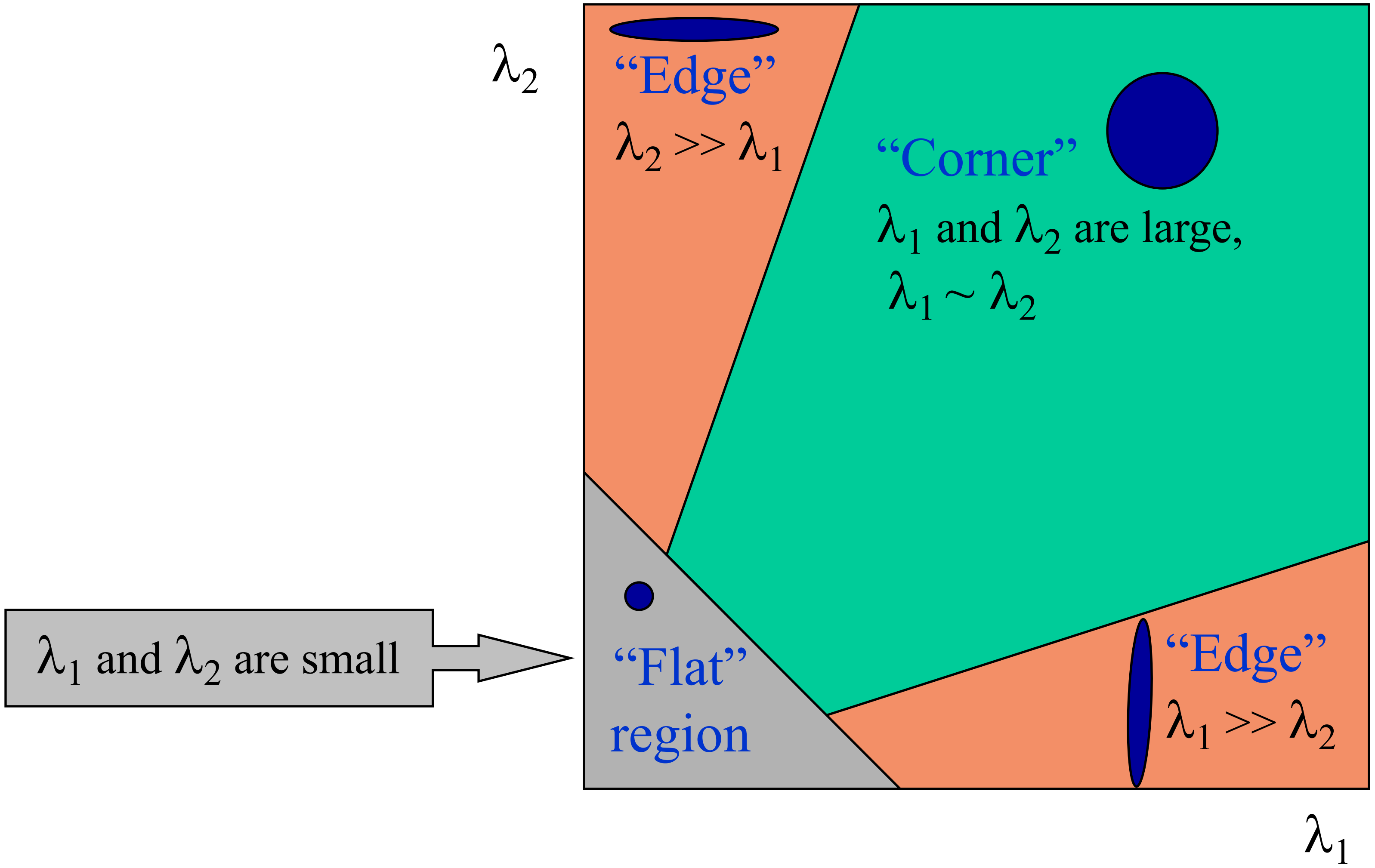
Eigenvectors of $A^T A$

$$A^T A = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \; I_y] = \sum \nabla I (\nabla I)^T$$

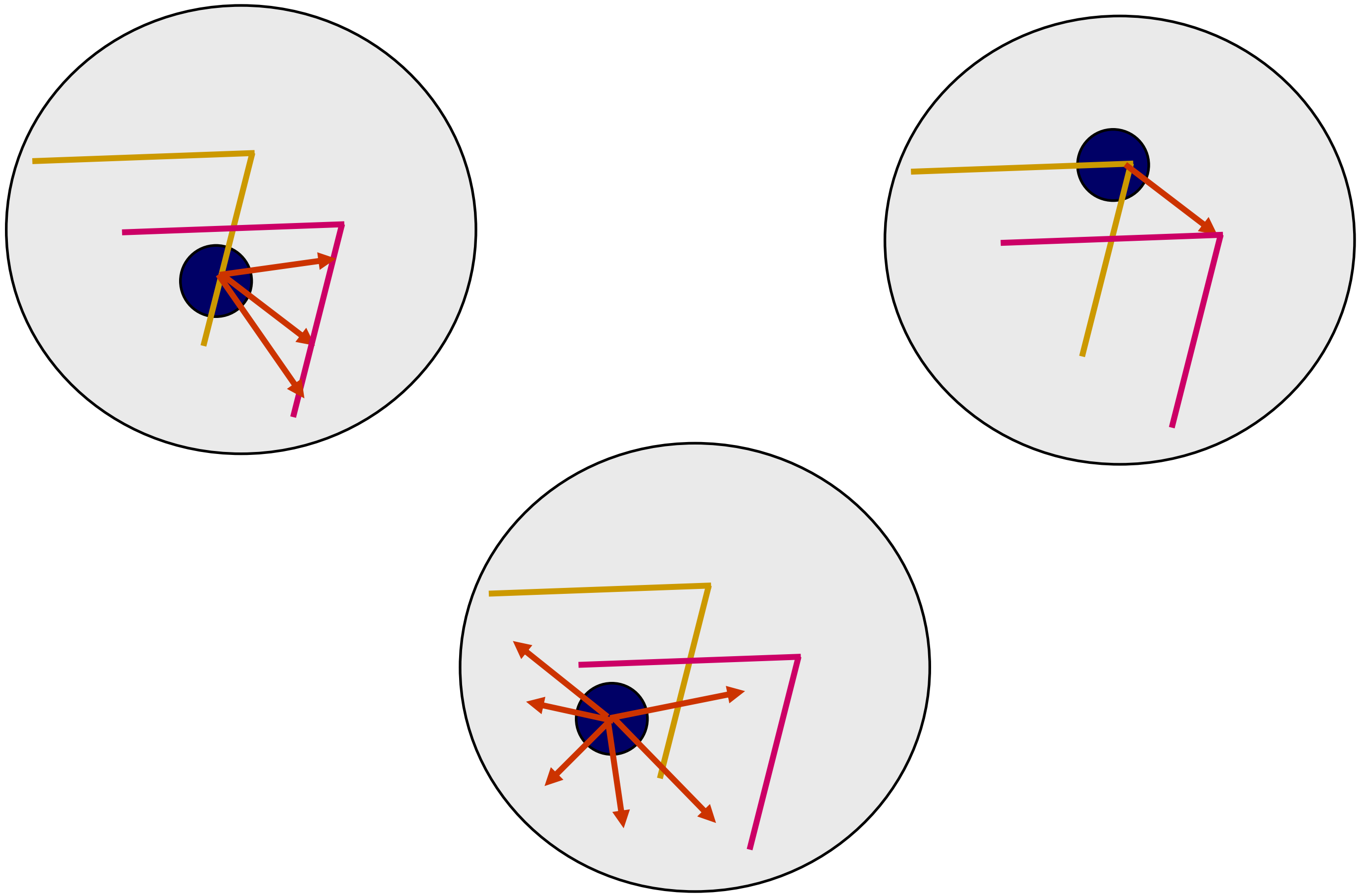
- Recall the Harris corner detector: $M = A^T A$ is the *second moment matrix*
- The eigenvectors and eigenvalues of M relate to edge direction and magnitude
  - The eigenvector associated with the larger eigenvalue points in the direction of fastest intensity change
  - The other eigenvector is orthogonal to it

Interpreting the eigenvalues

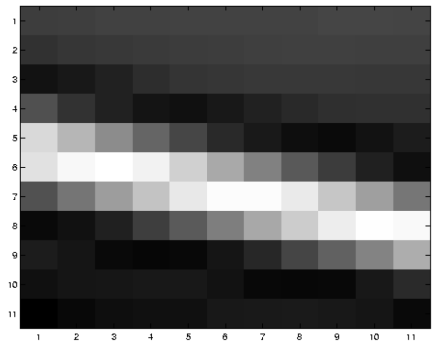
Classification of image points using eigenvalues of the second moment matrix:

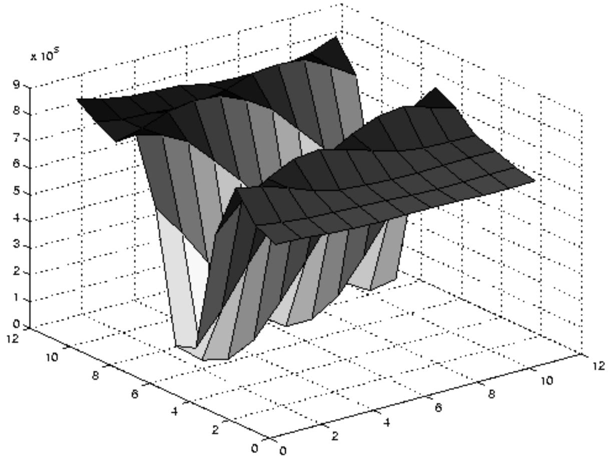


Local Patch Analysis



Edge



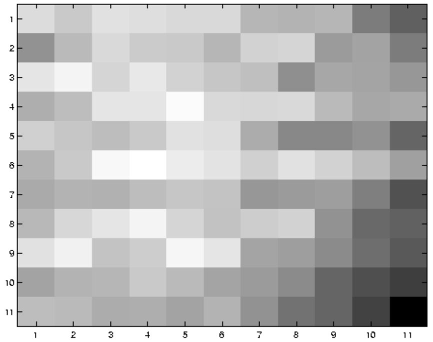


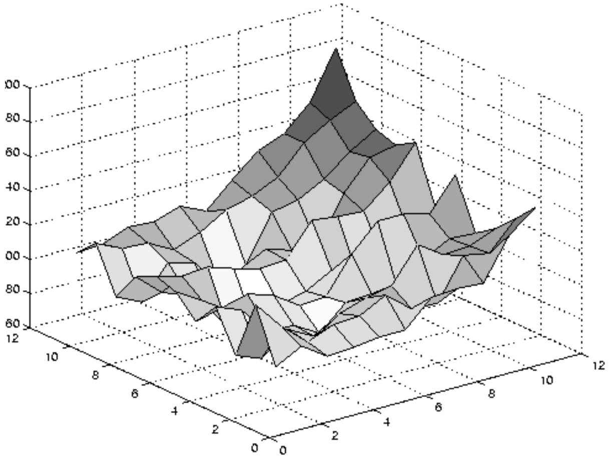
$$\sum \nabla I (\nabla I)^T$$

– large gradients, all the same

– large λ_1, small λ_2

Low texture region

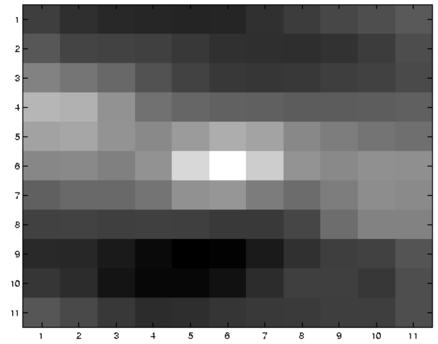


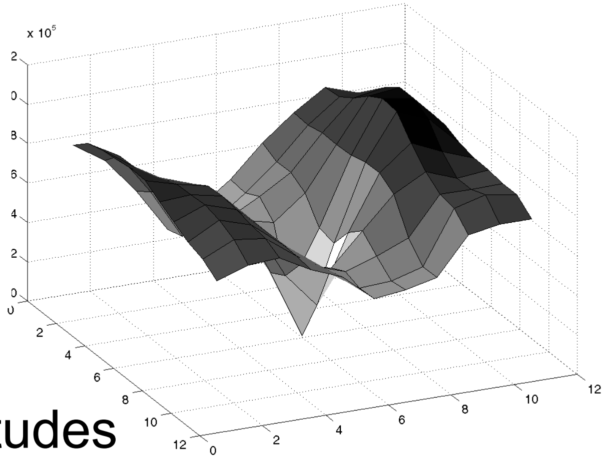


$$\sum \nabla I (\nabla I)^T$$

– gradients have small magnitude
– small λ_1, small λ_2

High textured region





$$\sum \nabla I (\nabla I)^T$$

– gradients are different, large magnitudes

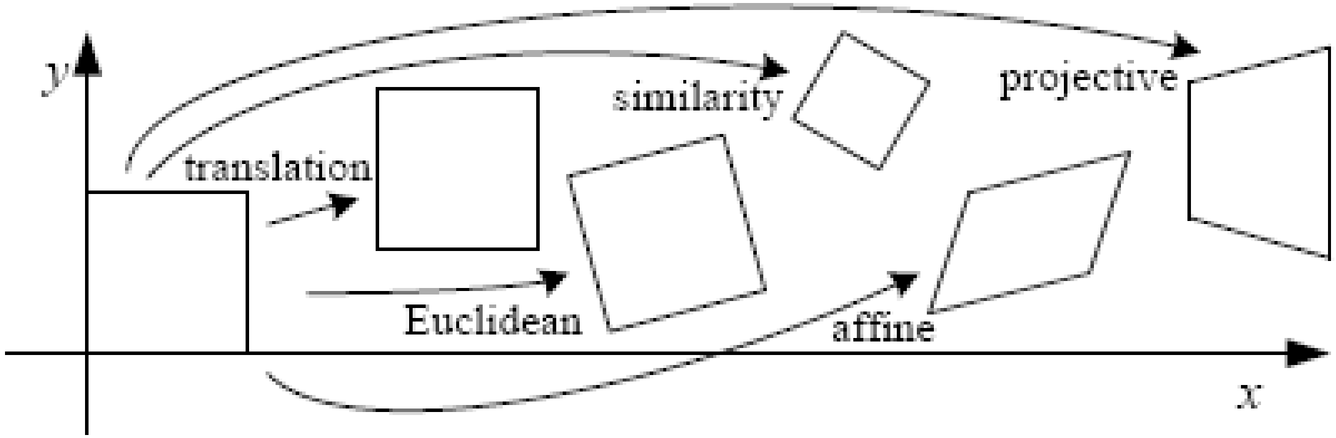
– large λ_1, large λ_2

Observation

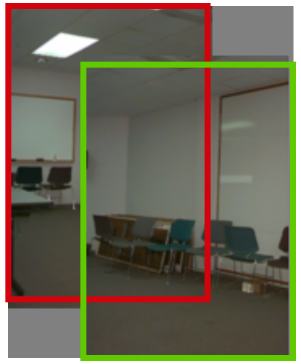
This is a two image problem BUT

- Can measure sensitivity by just looking at one of the images!
- This tells us which pixels are easy to track, which are hard
  - very useful later on when we do feature tracking...

Motion models

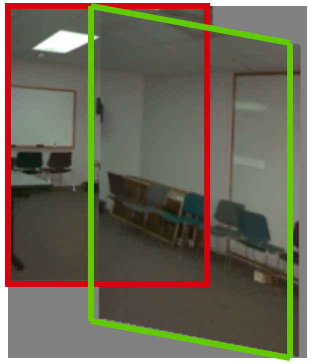


Translation



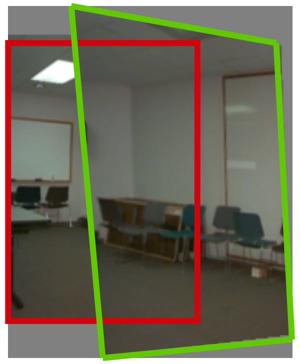
2 unknowns

Affine



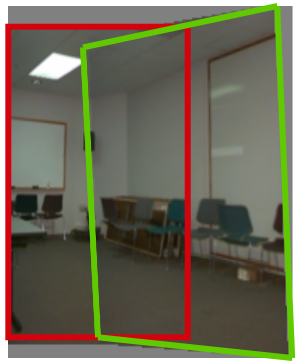
6 unknowns

Perspective



8 unknowns

3D rotation

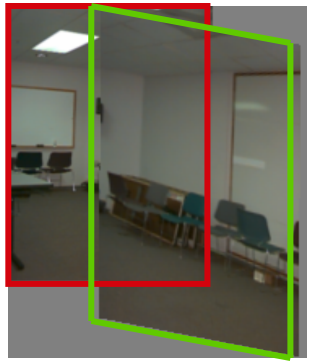


3 unknowns

Affine motion

$$u(x, y) = a_1 + a_2 x + a_3 y$$

$$v(x, y) = a_4 + a_5 x + a_6 y$$

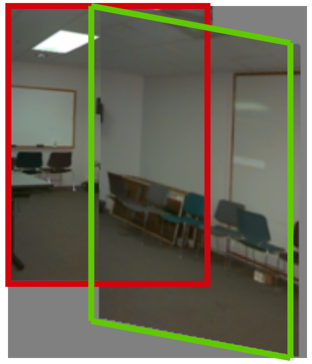


- Substituting into the brightness constancy equation:

$$\boxed{I_x \cdot u + I_y \cdot v + I_t \approx 0}$$

Affine motion

$$u(x, y) = a_1 + a_2 x + a_3 y$$
$$v(x, y) = a_4 + a_5 x + a_6 y$$



- Substituting into the brightness constancy equation:

$$\boxed{I_x (a_1 + a_2 x + a_3 y) + I_y (a_4 + a_5 x + a_6 y) + I_t \approx 0}$$

- Each pixel provides 1 linear constraint in 6 unknowns

- Least squares minimization:

$$\boxed{Err(\vec{a}) = \sum \left[I_x (a_1 + a_2 x + a_3 y) + I_y (a_4 + a_5 x + a_6 y) + I_t \right]^2}$$

Errors in Lukas-Kanade

What are the potential causes of errors in this procedure?

- Suppose A^TA is easily invertible
- Suppose there is not much noise in the image

When our assumptions are violated

- Brightness constancy is not satisfied
- The motion is not small
- A point does not move like its neighbors
  - window size is too large
  - what is the ideal window size?

Iterative Refinement

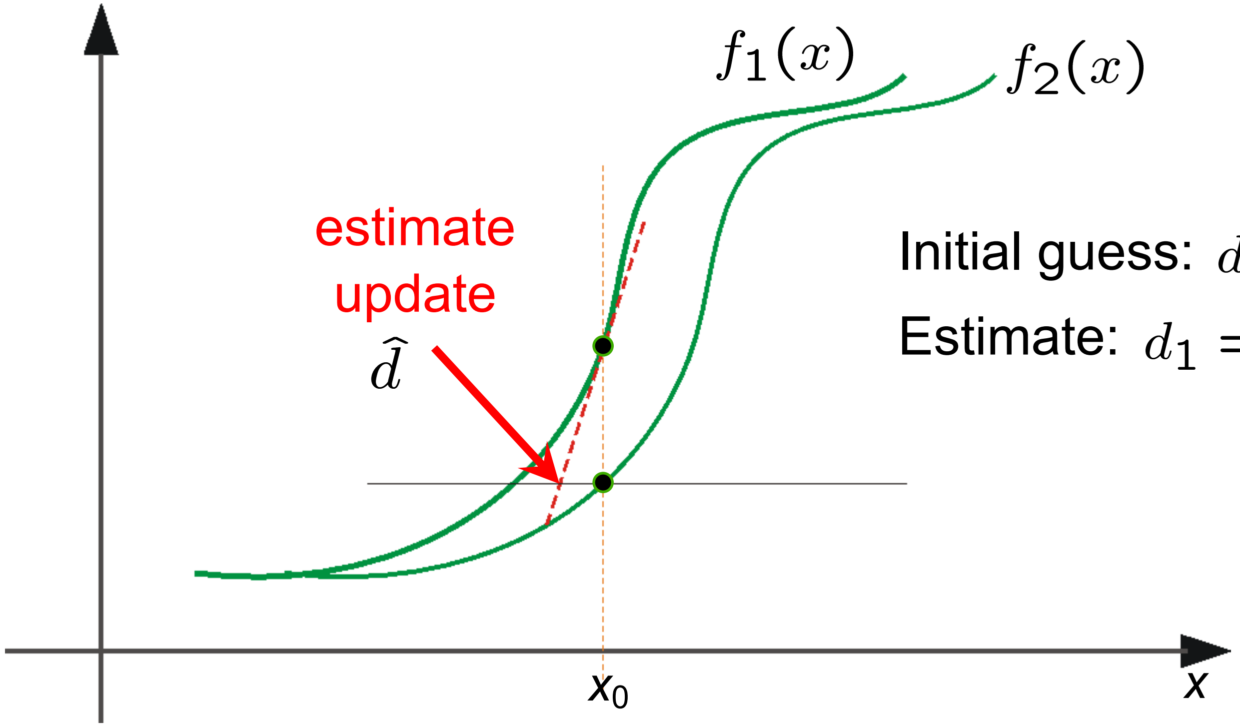
Iterative Lukas-Kanade Algorithm

1. Estimate velocity at each pixel by solving Lucas-Kanade equations
2. Warp H towards I using the estimated flow field

   - use image warping techniques

3. Repeat until convergence

Optical Flow: Iterative Estimation

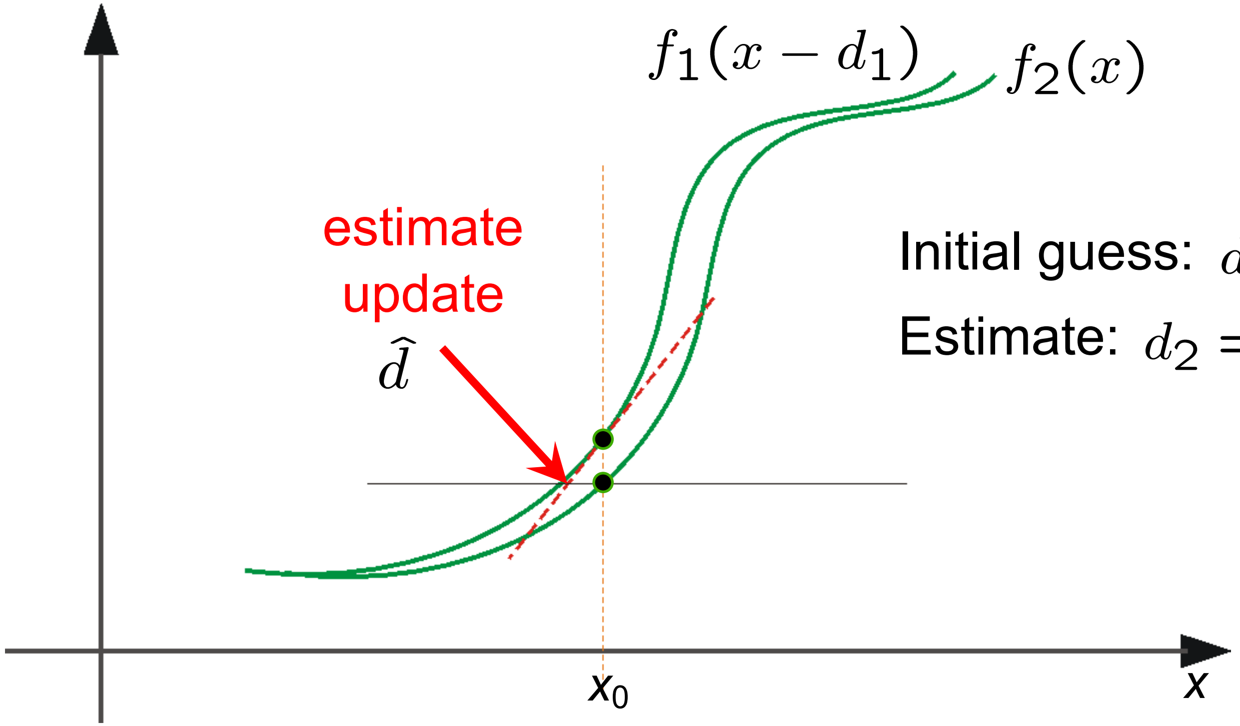


Initial guess: $d_0 = 0$

Estimate: $d_1 = d_0 + \hat{d}$

(using *d* for *displacement* here instead of *u*)

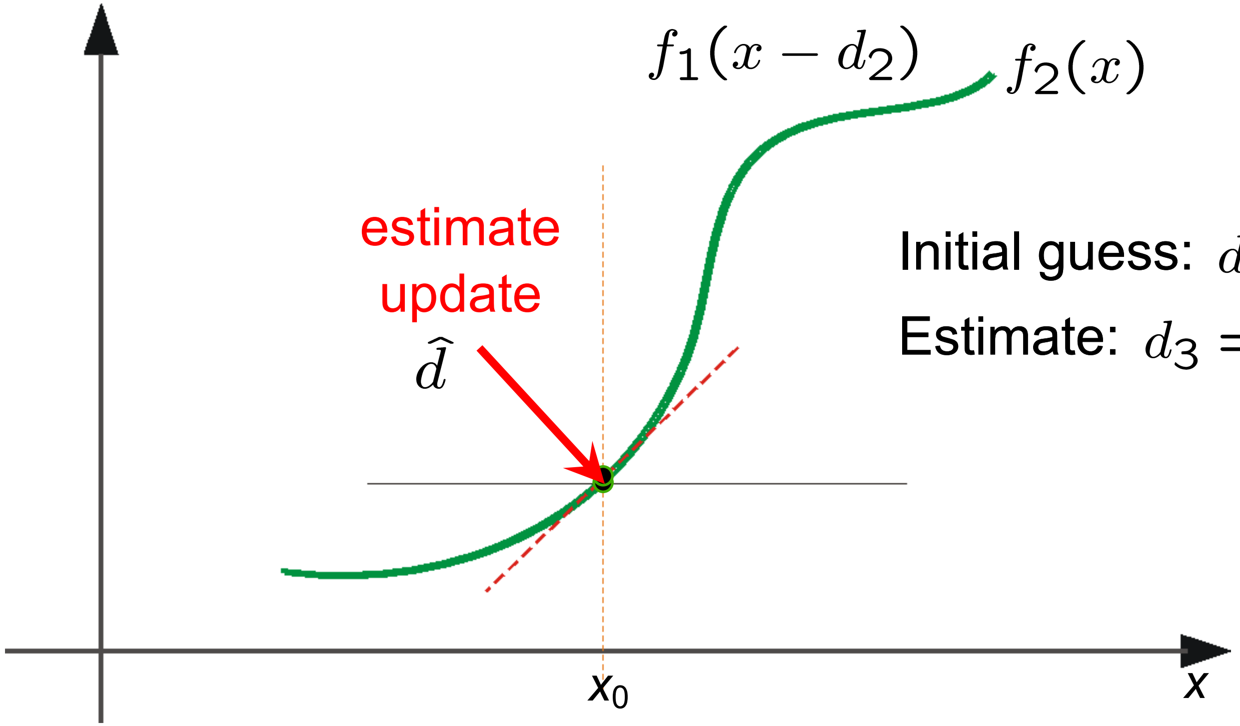
Optical Flow: Iterative Estimation



Initial guess: d_1

Estimate: $d_2 = d_1 + \hat{d}$

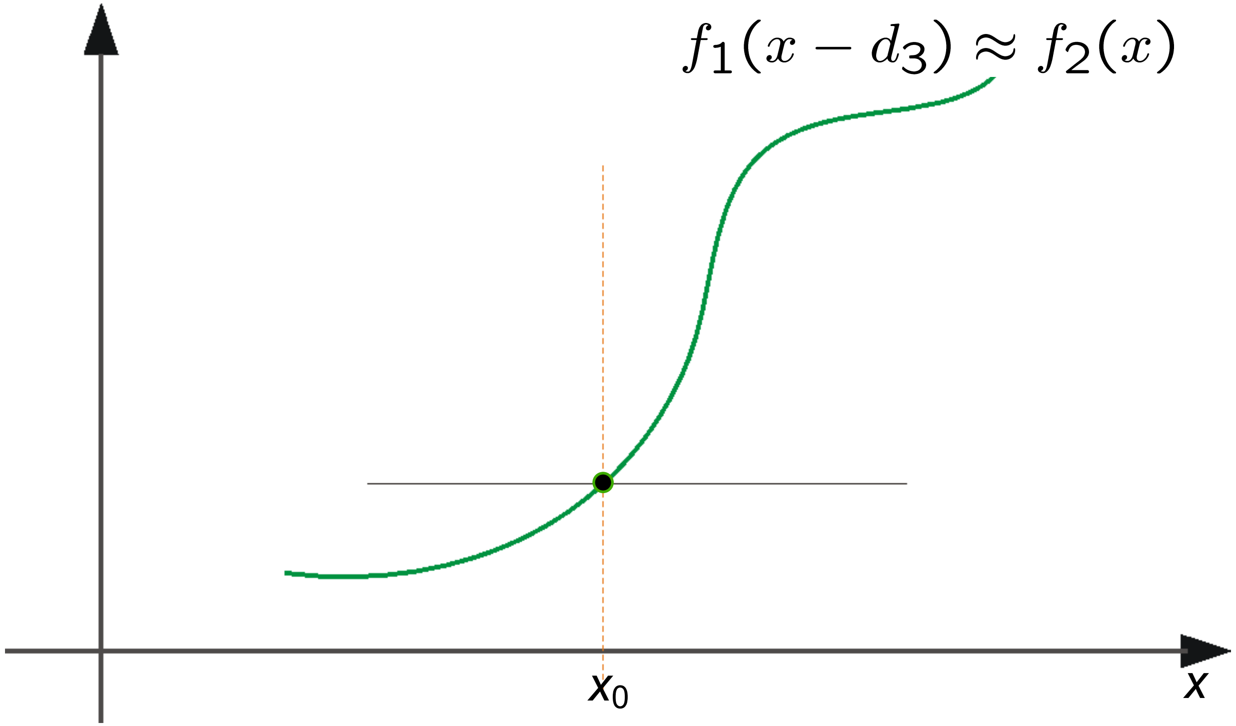
Optical Flow: Iterative Estimation



Initial guess: d_2

Estimate: $d_3 = d_2 + \hat{d}$

Optical Flow: Iterative Estimation



Optical Flow: Iterative Estimation

Some Implementation Issues:

- Warping is not easy (ensure that errors in warping are smaller than the estimate refinement)
- Warp one image, take derivatives of the other so you don’t need to re-compute the gradient after each iteration.
- Often useful to low-pass filter the images before motion estimation (for better derivative estimation, and linear approximations to image intensity)

Revisiting the small motion assumption

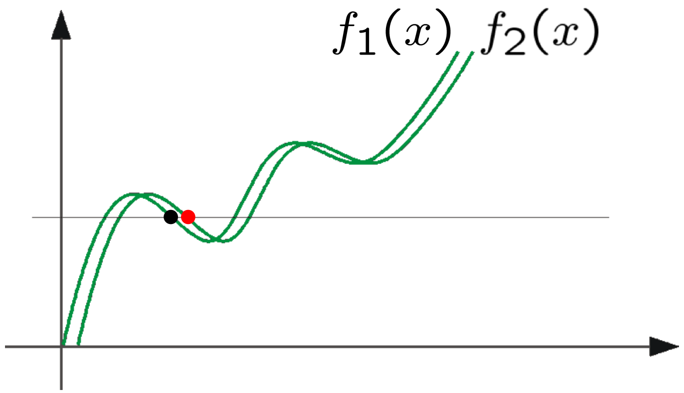
Is this motion small enough?

- Probably not—it's much larger than one pixel (2^{nd} order terms dominate)
- How might we solve this problem?

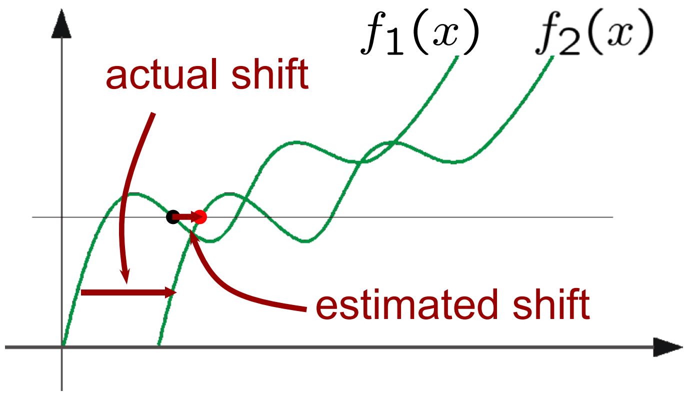
Optical Flow: Aliasing

Temporal aliasing causes ambiguities in optical flow because images can have many pixels with the same intensity.

I.e., how do we know which ‘correspondence’ is correct?



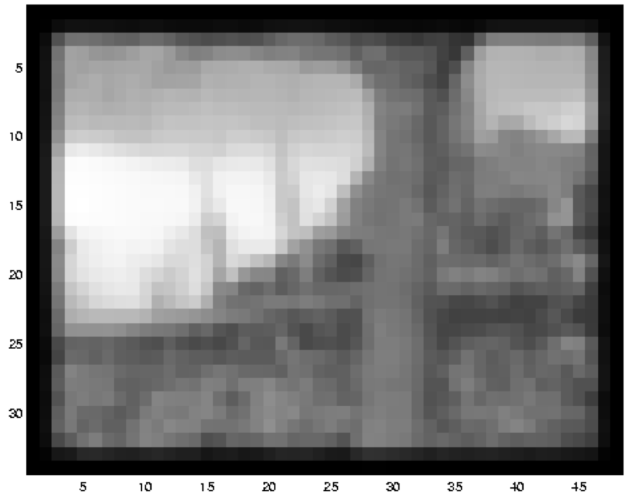
nearest match is correct (no aliasing)

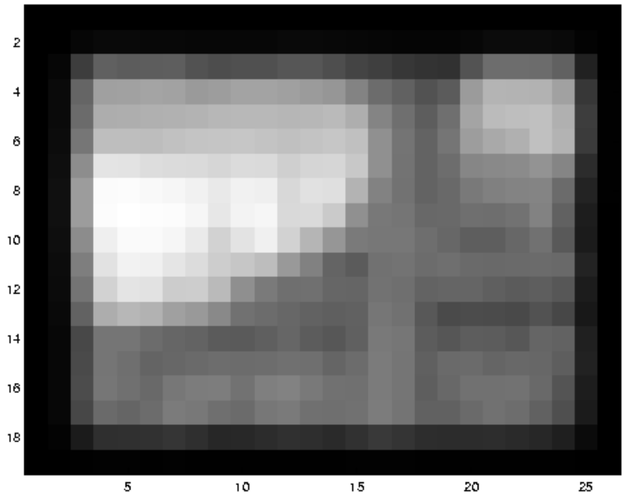


nearest match is incorrect (aliasing)

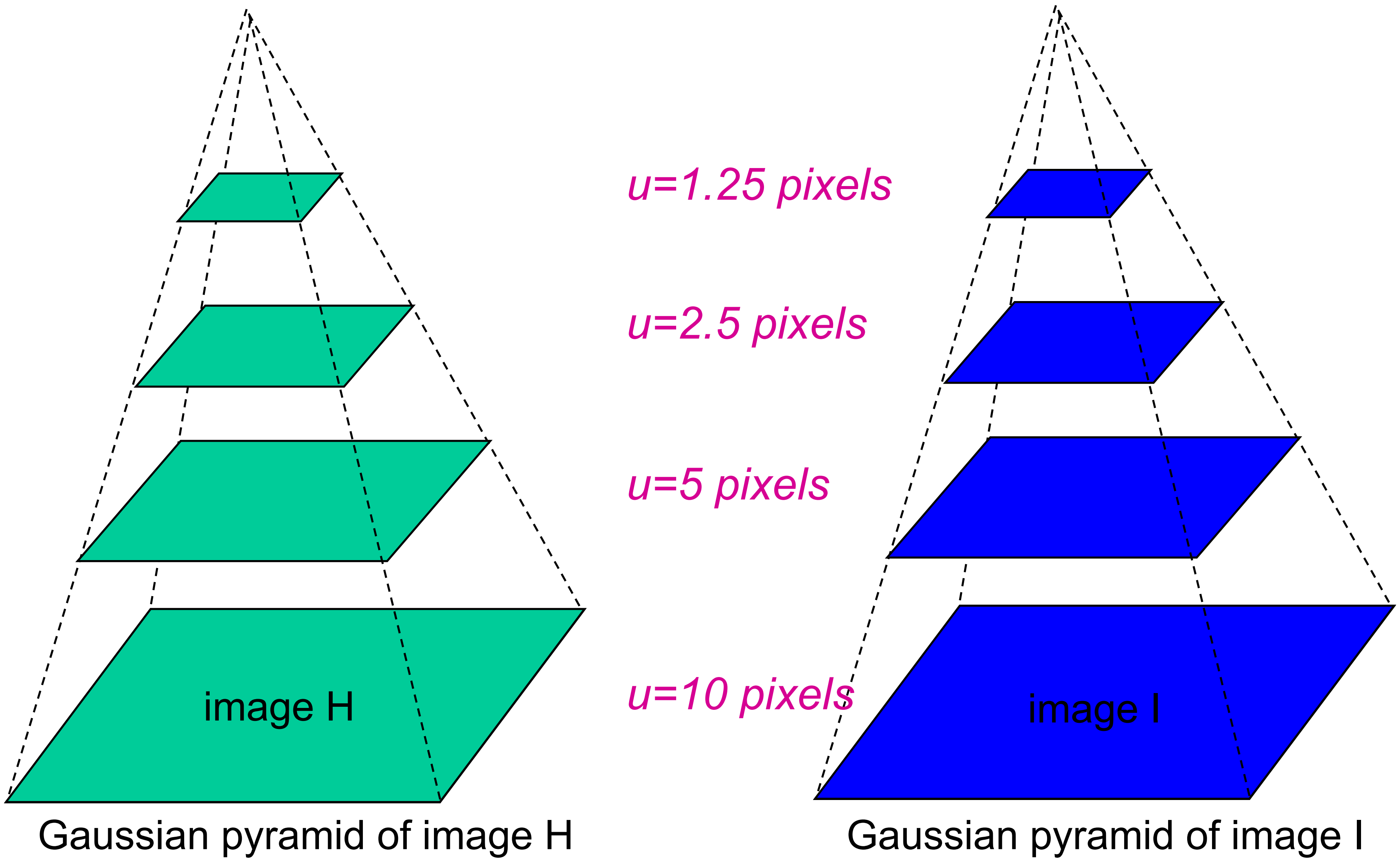
To overcome aliasing: coarse-to-fine estimation.

Reduce the resolution!

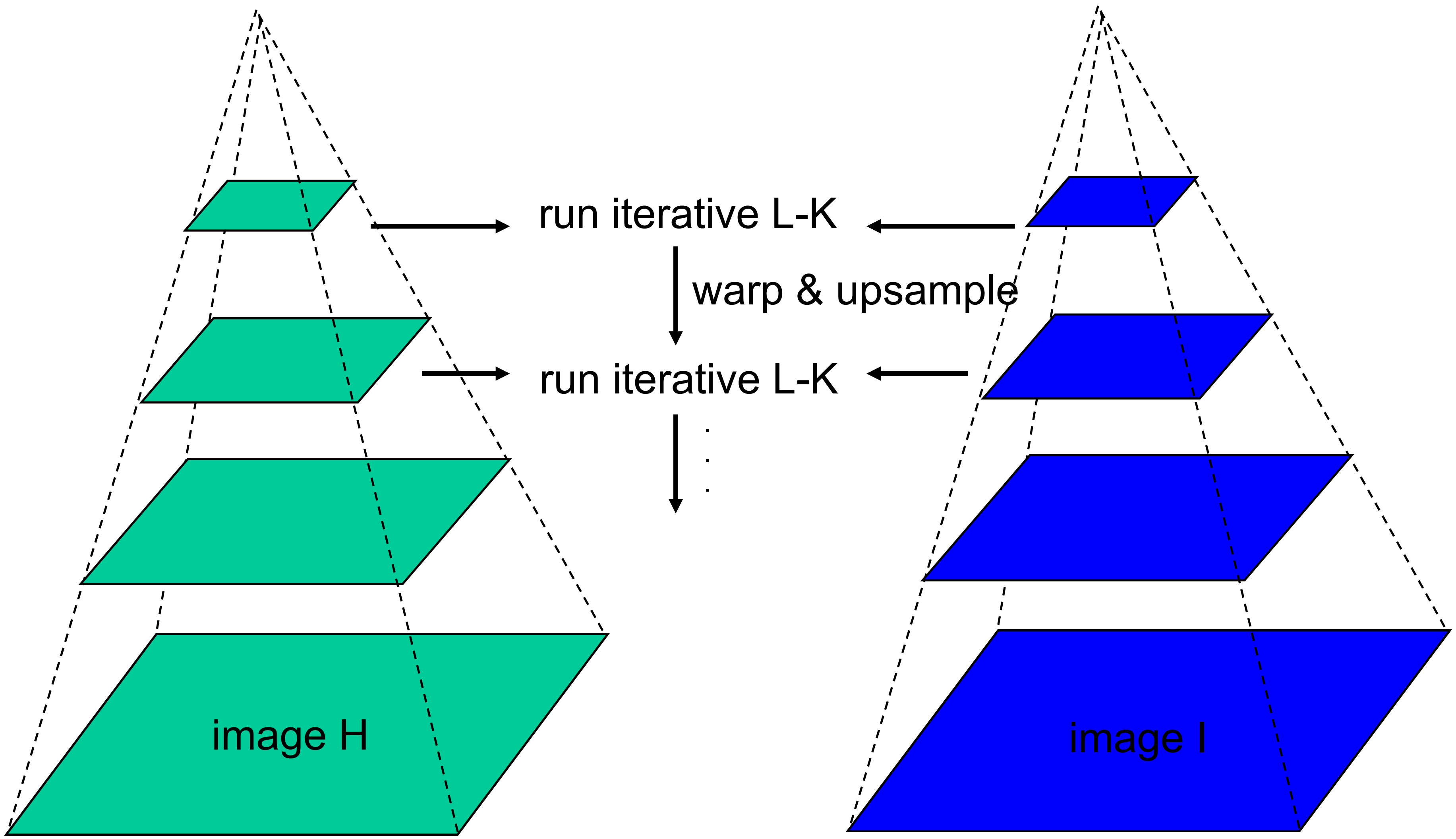




Coarse-to-fine optical flow estimation



Coarse-to-fine optical flow estimation



Gaussian pyramid of image H

Gaussian pyramid of image I

Recap: Classes of Techniques

Direct-methods (e.g. optical flow)

- Directly recover image motion from spatio-temporal image brightness variations
- Global motion parameters directly recovered without an intermediate feature motion calculation
- Dense motion fields, but more sensitive to appearance variations
- Suitable for video and when image motion is small (< 10 pixels)

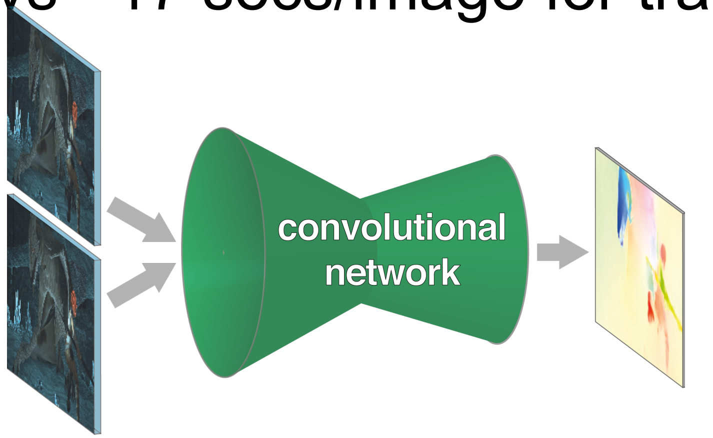
Feature-based methods (e.g. SIFT+Ransac+regression) [To be covered]

- Extract visual features (corners, textured areas) and track them over multiple frames
- Sparse motion fields, but possibly robust tracking
- Suitable especially when image motion is large (10-s of pixels)

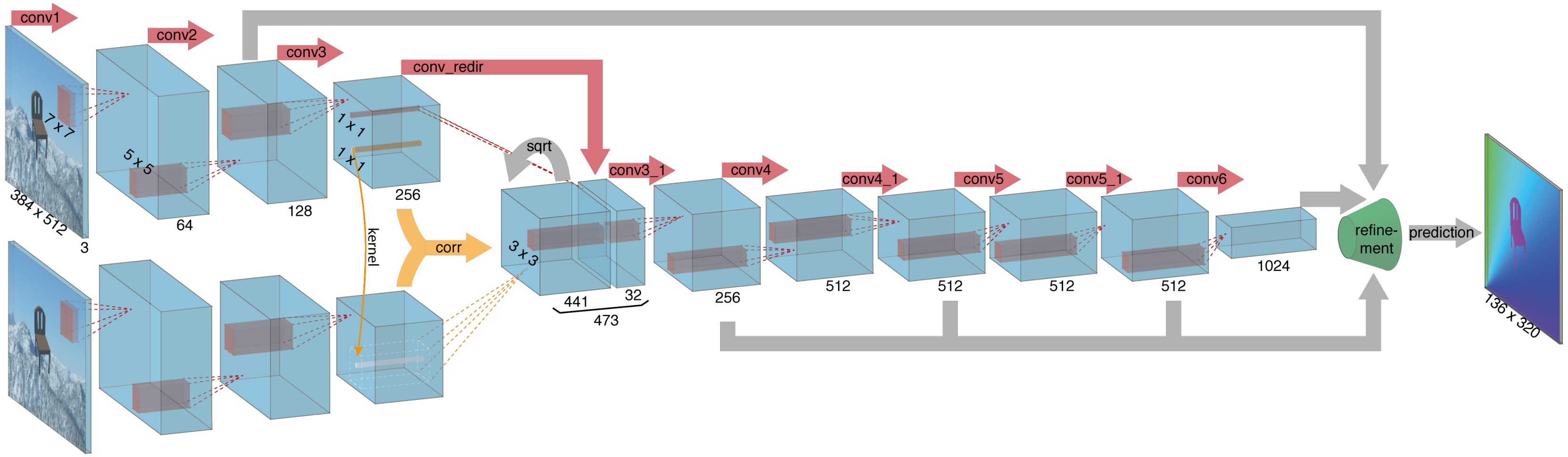
FlowNet

FlowNet: Learning Optical Flow with Convolutional Networks [Fischer et al. 2015]

~ 1sec/image vs ~17 secs/image for traditional optical flow

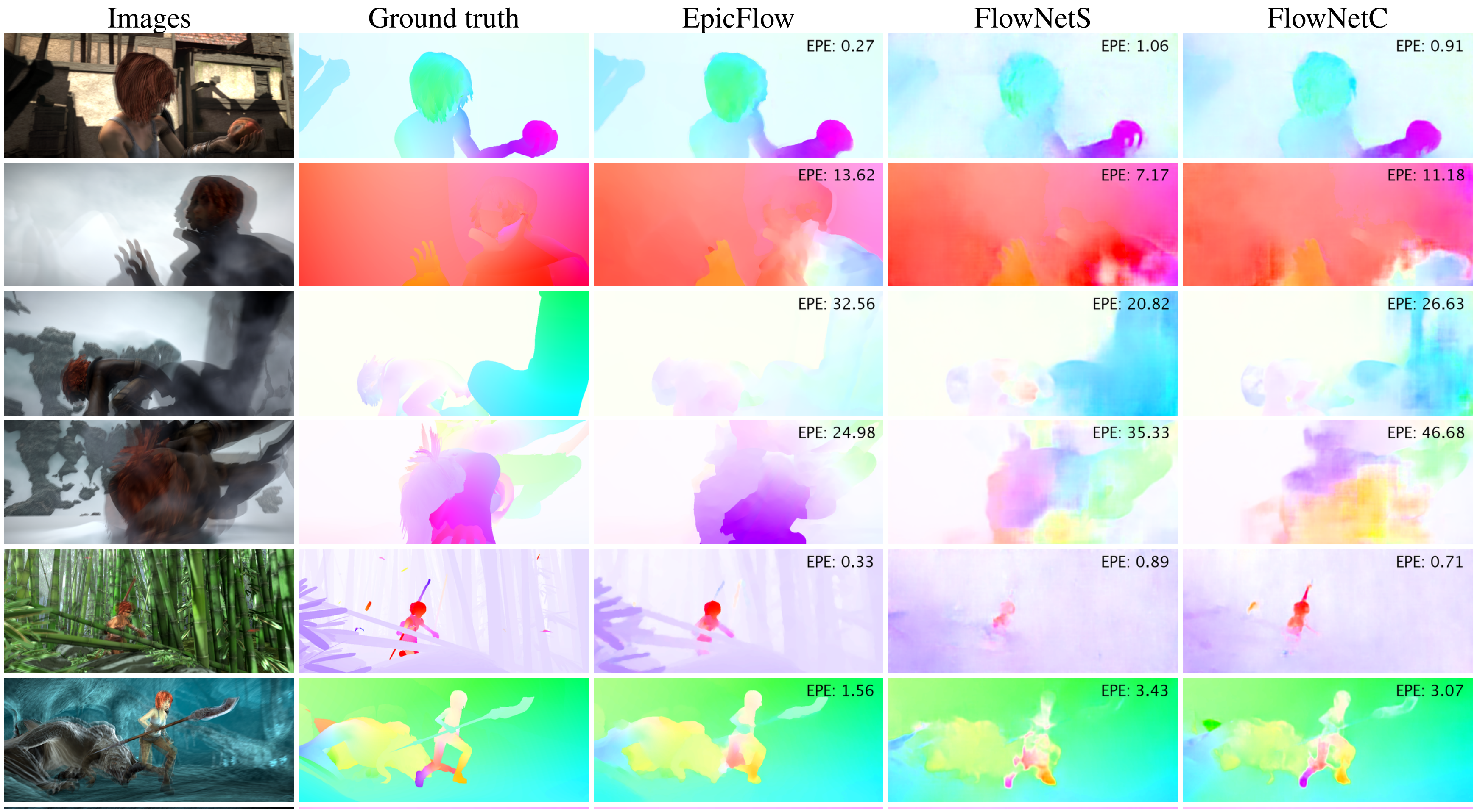


FlowNetCorr

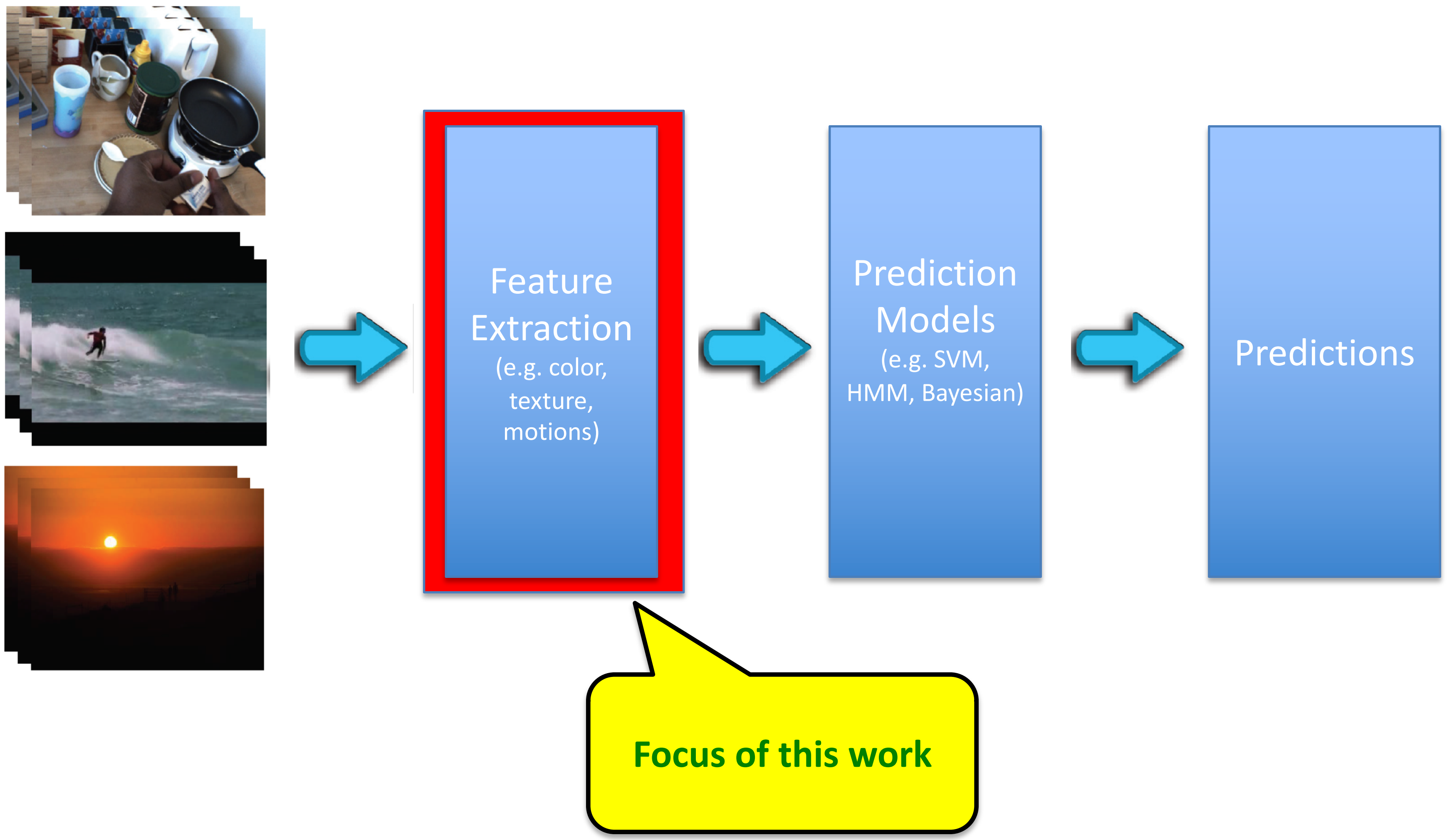


FlowNet

- FlowNet: Learning Optical Flow with Convolutional Networks [Fischer et al. 2015]

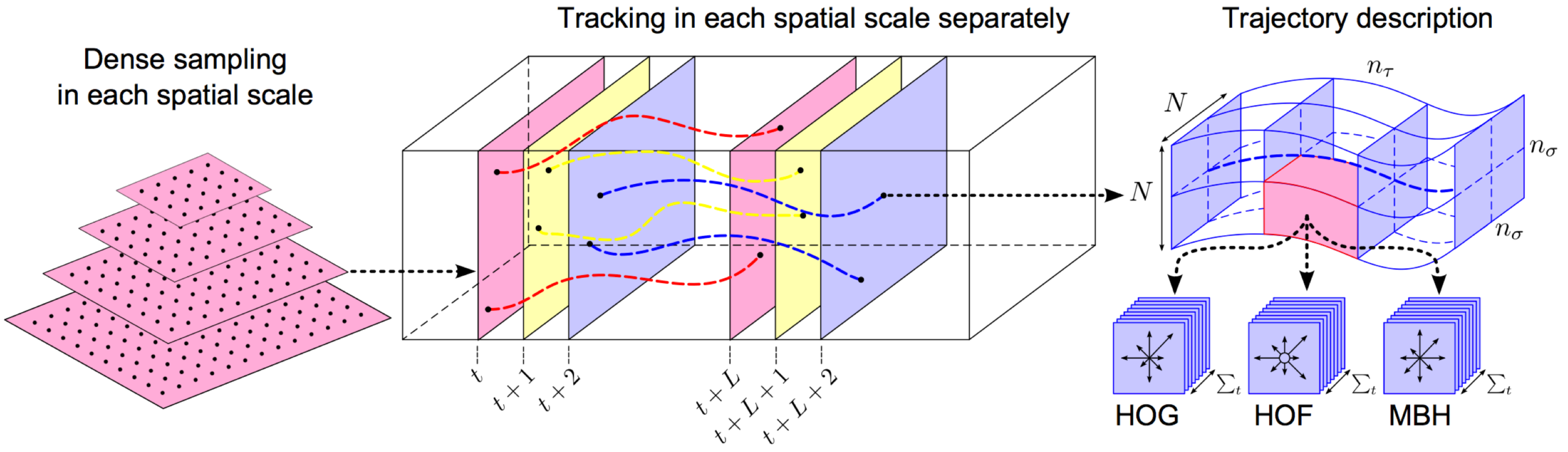


Traditional Computer Vision Pipeline



Best (non-DL) Video Features

- improved Dense Trajectories (iDT)

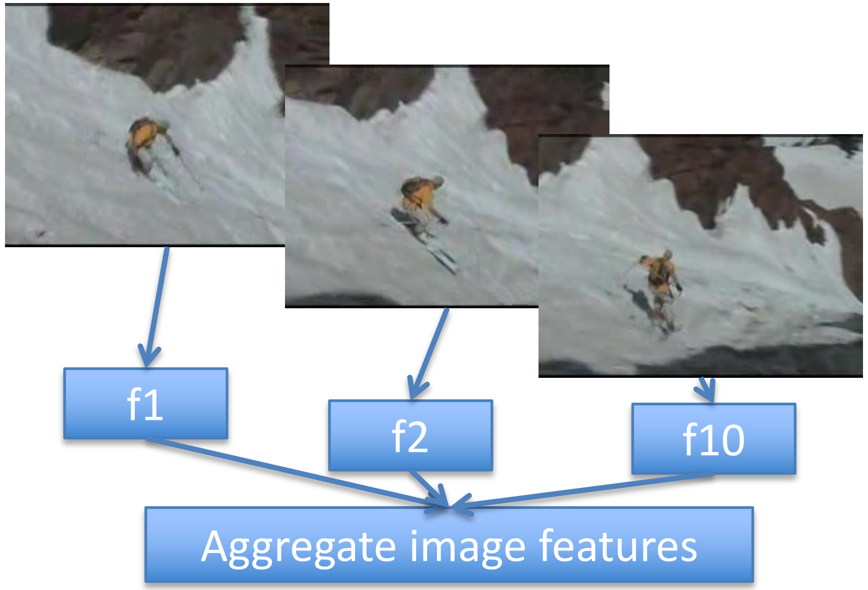


Wang et al. IJCV'13

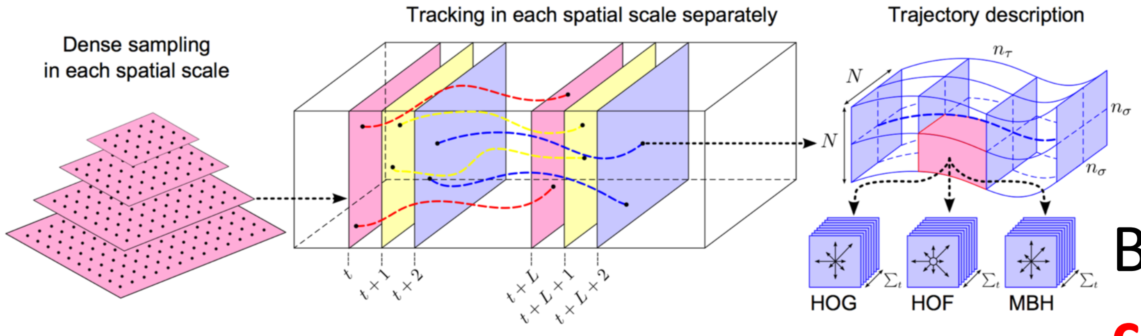
Pros:
- Don't need to learn
- Don't need large-scale training data

Cons:
- Highly hand-crafted
- Computational intensive
- Hard to parallelize

Spatiotemporal Feature Learning



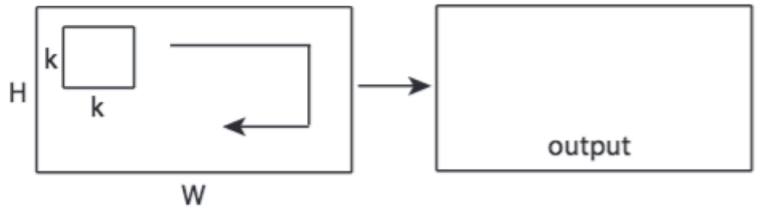
No explicit **motion modeling**



- Wang et al. IJCV'13

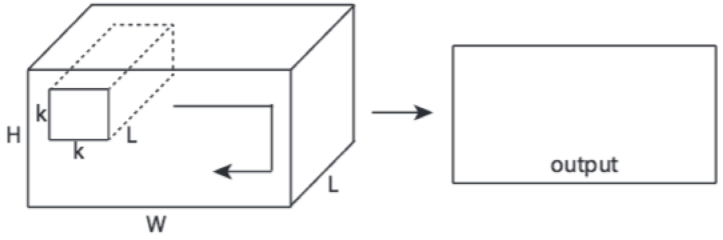
Biased to **human design** &
computationally expensive

Why 3D ConvNets?



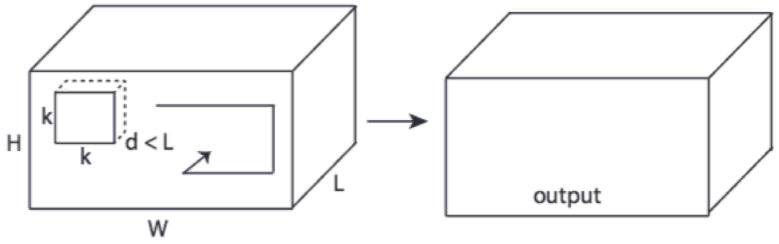
2D convolve on an image

-> no motion modeling ✖



2D convolve on multiple images as channels

-> collapse temporal signal after one convolution layer ✖



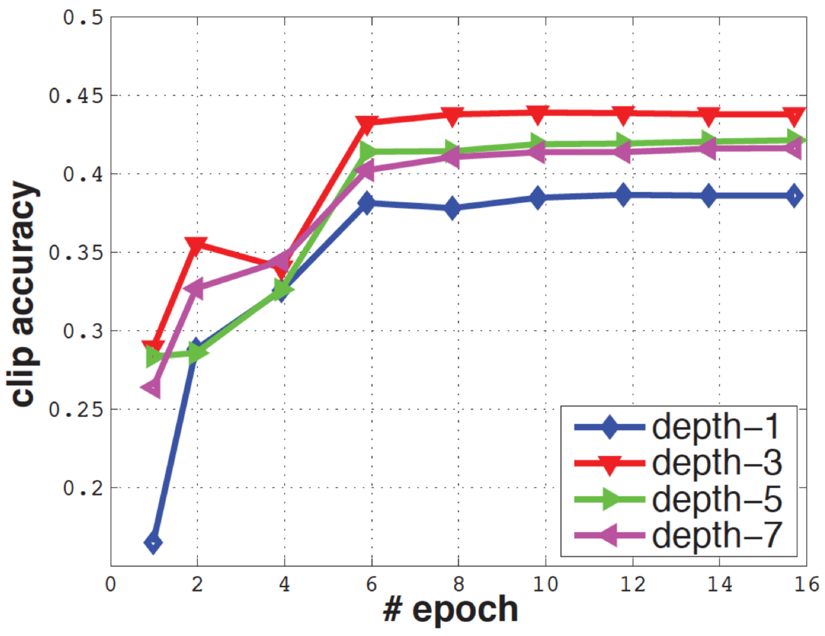
Spatial-temporally convolve on multiple frames

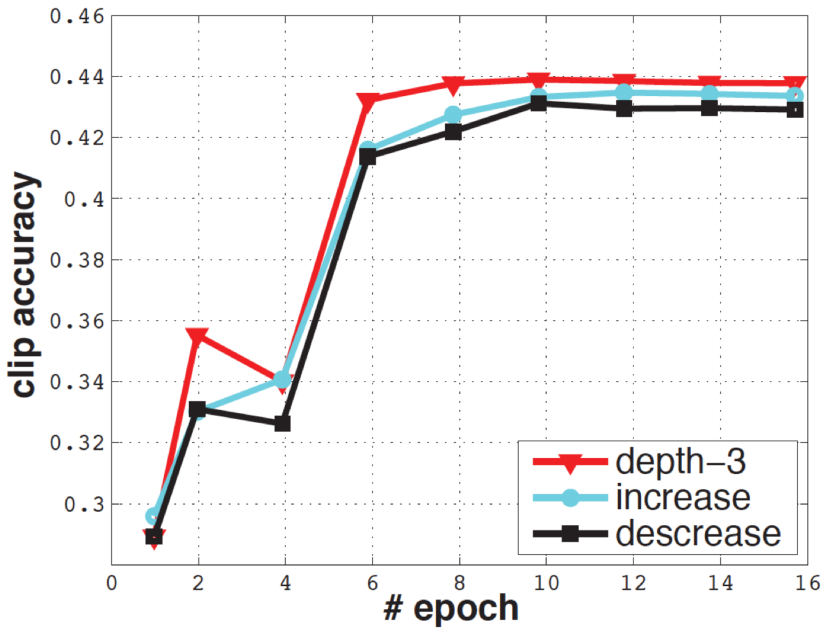
-> hierarchically group temporal signal ✔

What is a Good Architecture for 3D ConvNets?

D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, ***Learning Spatiotemporal Features with 3D Convolutional Networks***, ICCV15.

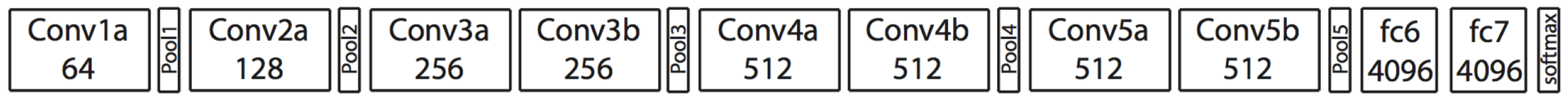
- Dataset: UCF101
- Use VGG-similar architecture, varying kernel temporal length





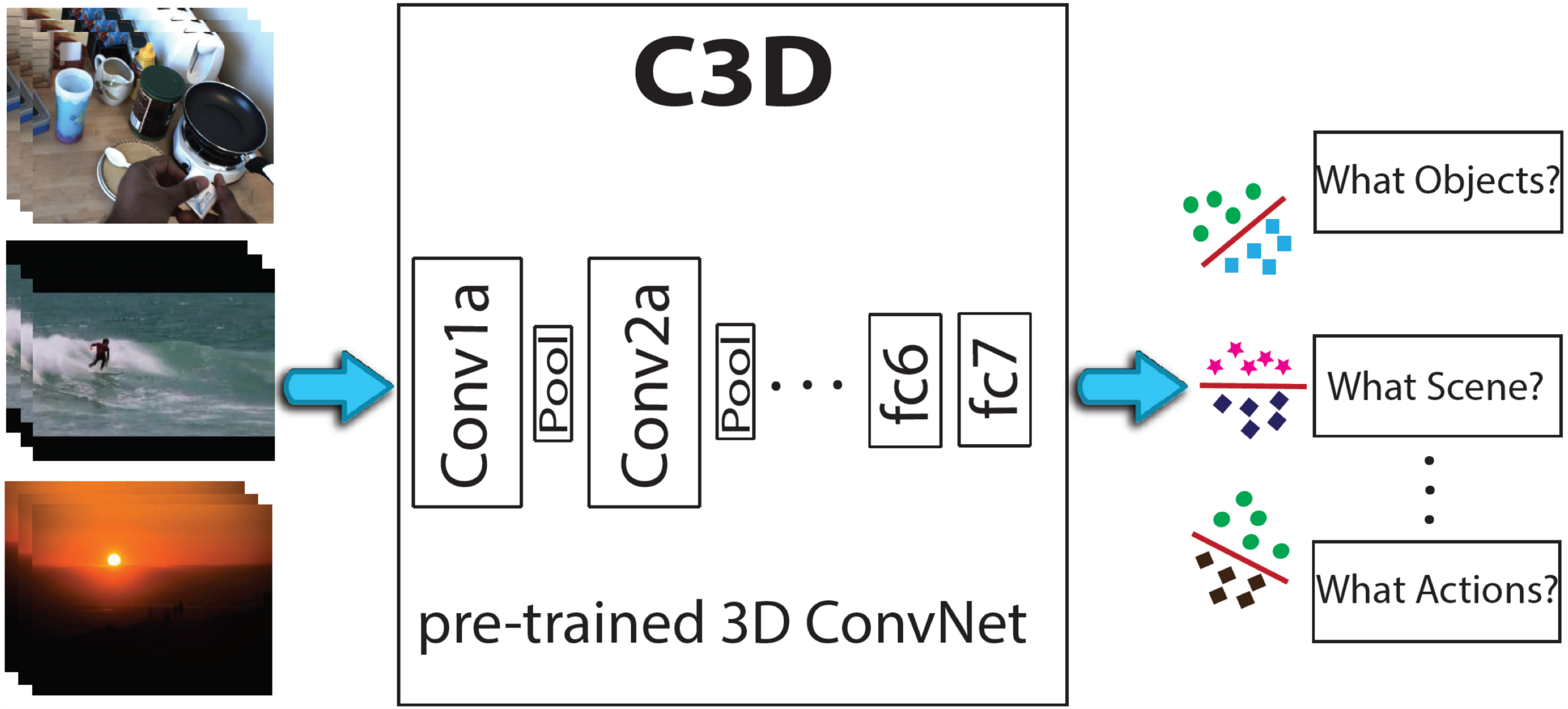
What is a Good Architecture for 3D ConvNets?

D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, ***Learning Spatiotemporal Features with 3D Convolutional Networks***, ICCV15.



- C3D architecture
  - 8 convolution, 5 pool, 2 fully-connected layers
  - 3x3x3 convolution kernels
  - 2x2x2 pooling kernels
- Dataset: Sports-1M [Karpathy et al. CVPR14]
  - 1.1M videos of 487 different sport categories
  - Train/test splits are provided

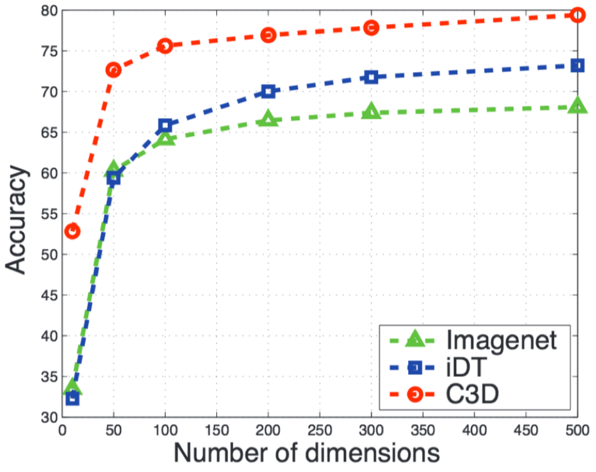
C3D as Generic Features

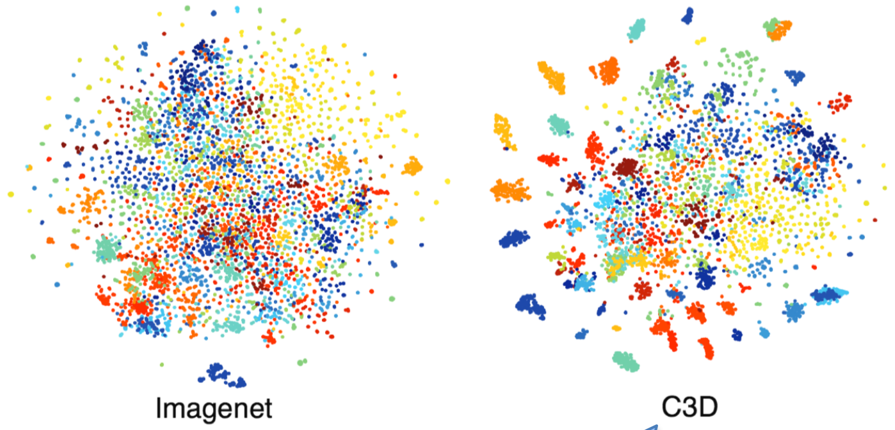


Simple recipe: C3D + linear SVM = good performance

Video Classification with C3D

Dataset Task	Sport1M action recognition	UCF101 action recognition	ASLAN action similarity labeling	YUPENN scene classification	UMD scene classification	Object object recognition
Method	[19]	[39]([26])	[31]	[10]	[10]	[32]
Result	80.2	75.8 (89.1)	68.7	96.2	77.7	12.0
C3D	**85.2**	**85.2 (90.4)**	**78.3**	**98.1**	**87.7**	**22.3**
Δ	5.0	9.4 (1.3)	9.6	1.9	10.0	10.3

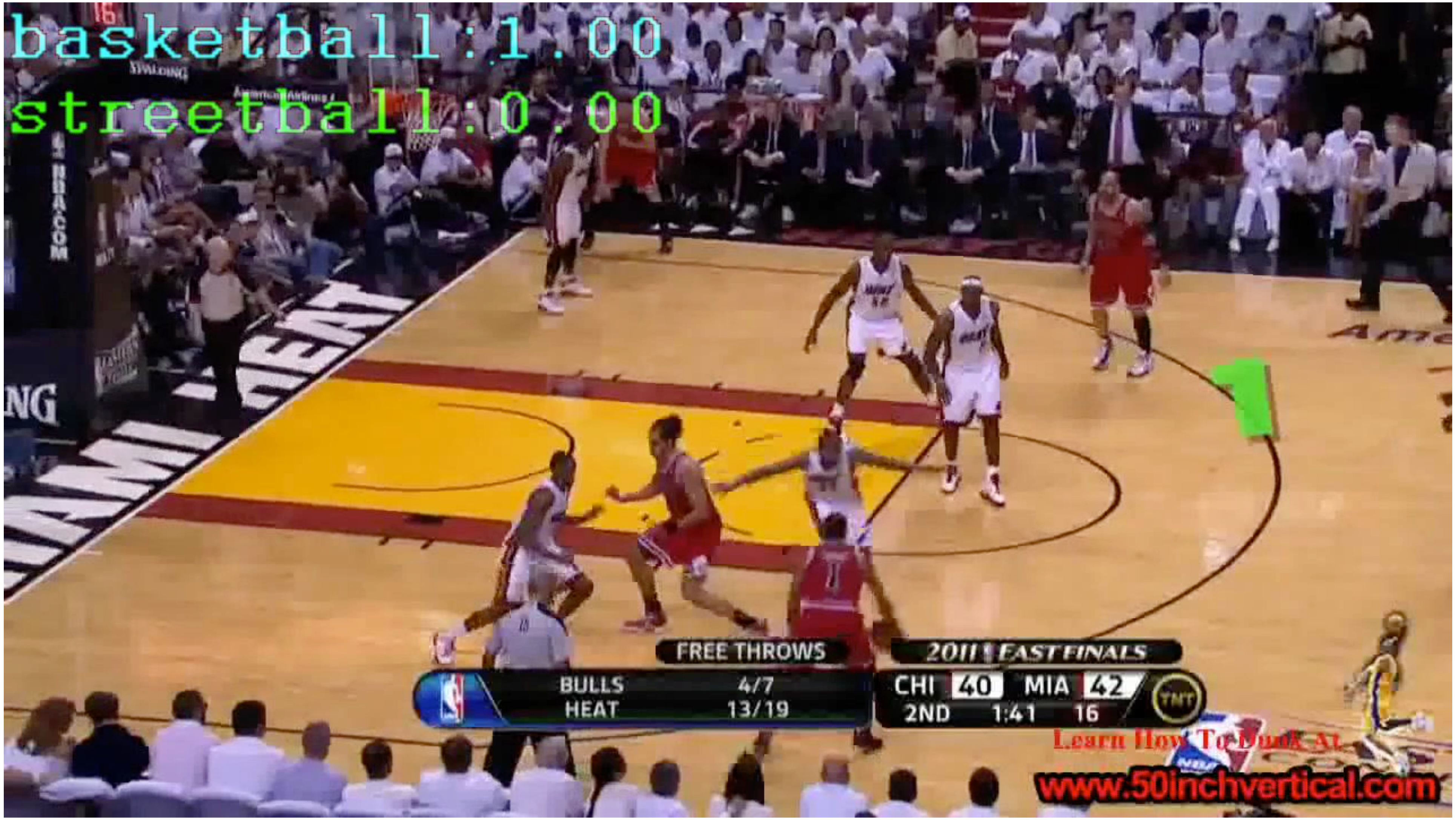




C3D is discriminative and compact!

C3D code/model is publicly available

Action Recognition Task
UCF101



Overview

- Optical Flow
- ConvNets for Video

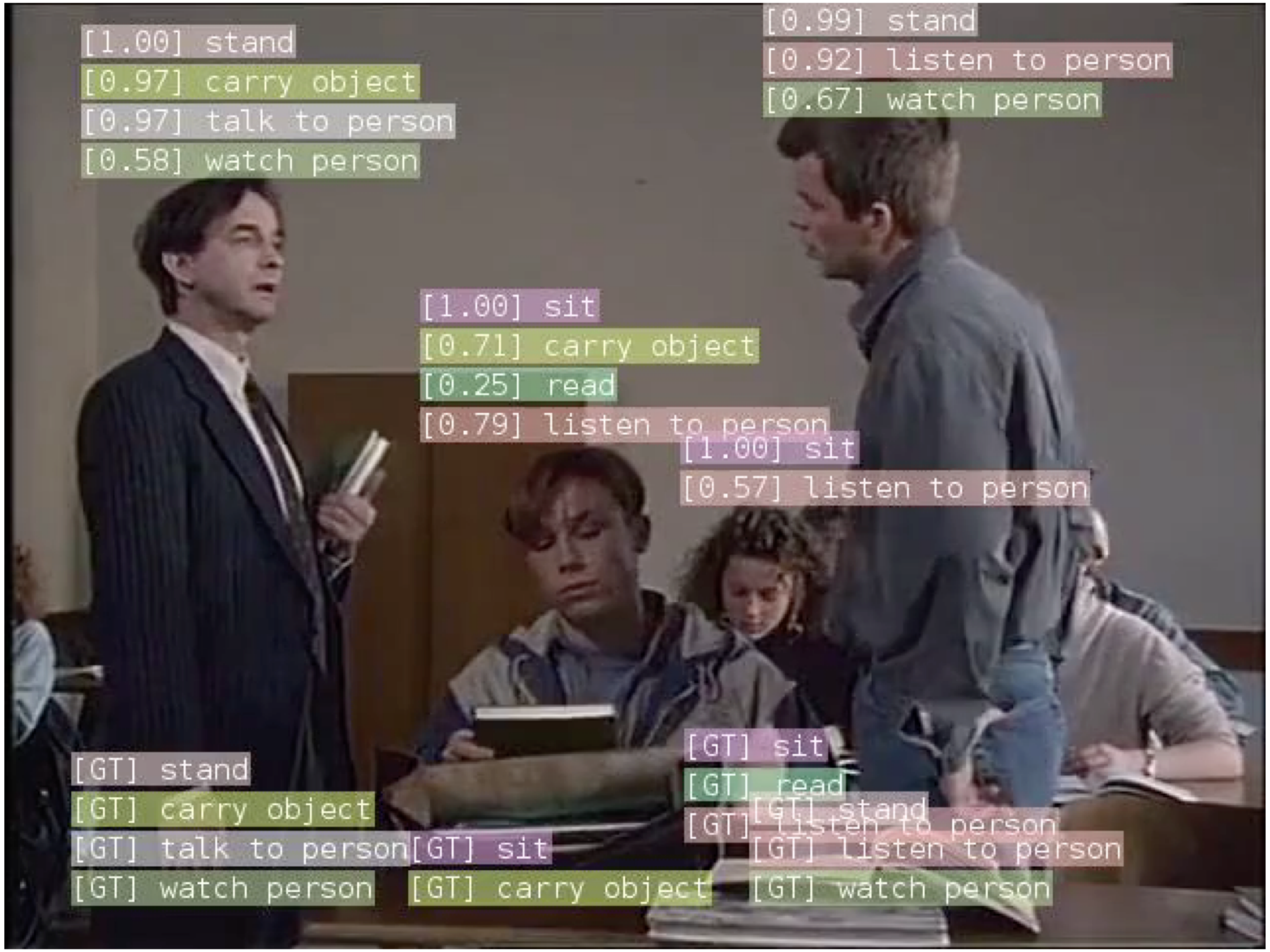
Recognition in Video

ICCV 2019 Tutorial

Christoph Feichtenhofer

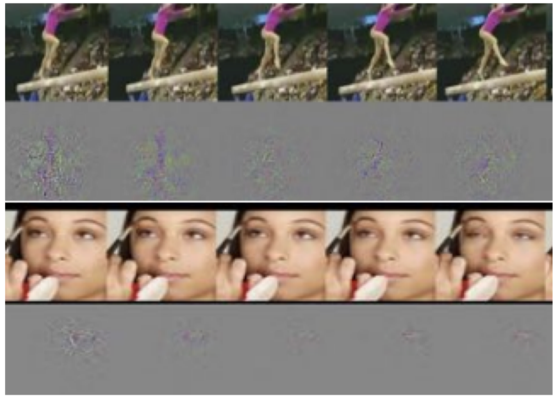
Facebook AI Research (FAIR)

Task: Human action classification & detection

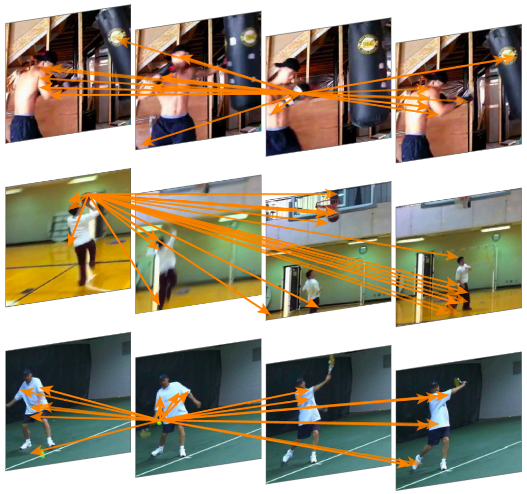


C. Feichtenhofer, H. Fan, J. Malik, K. He SlowFast Networks for Video Recognition. ICCV 2019

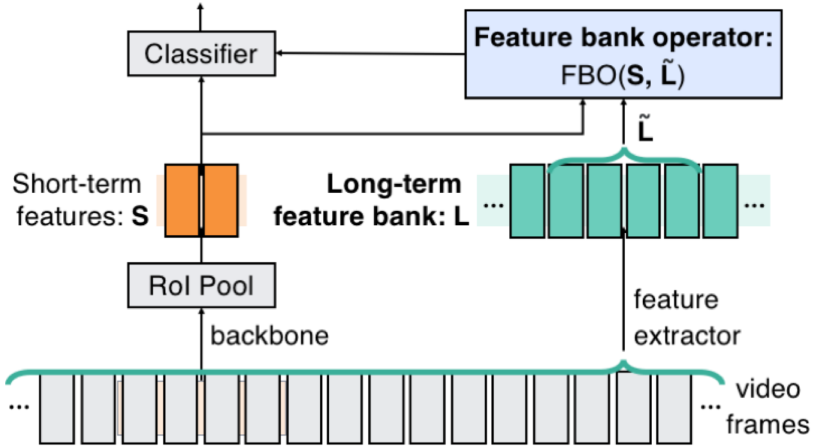
Outline: Components for state-of-the-art video understanding



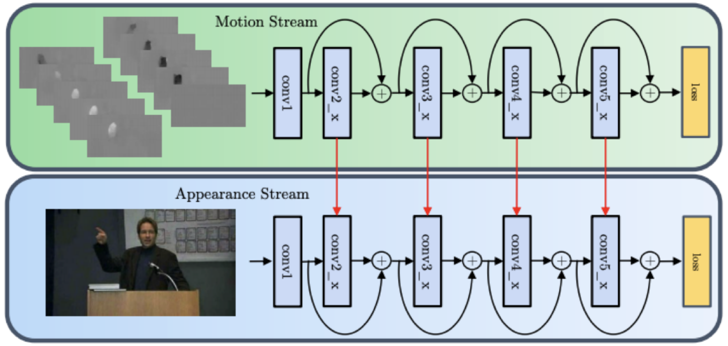
3D ConvNets
[Taylor et al. 2010, Karpathy et al. 2014, Tran et al. 2015,...]



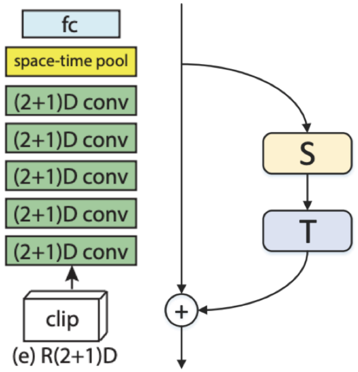
Attention-based models, Non-local network blocks, [Wang et al., 2018 2019, Girdhar et al. 2019 ,...]



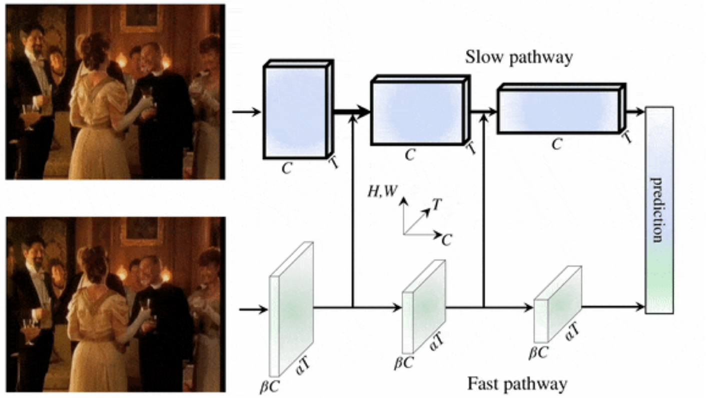
Long-term Models [Varol et al. 2017, Wu et al. 2019, ...]



Two-stream ConvNets (RGB+optical flow)
[Simonyan et al. 2014, Feichtenhofer et al. 2016, Wang et al. 2016, ...]



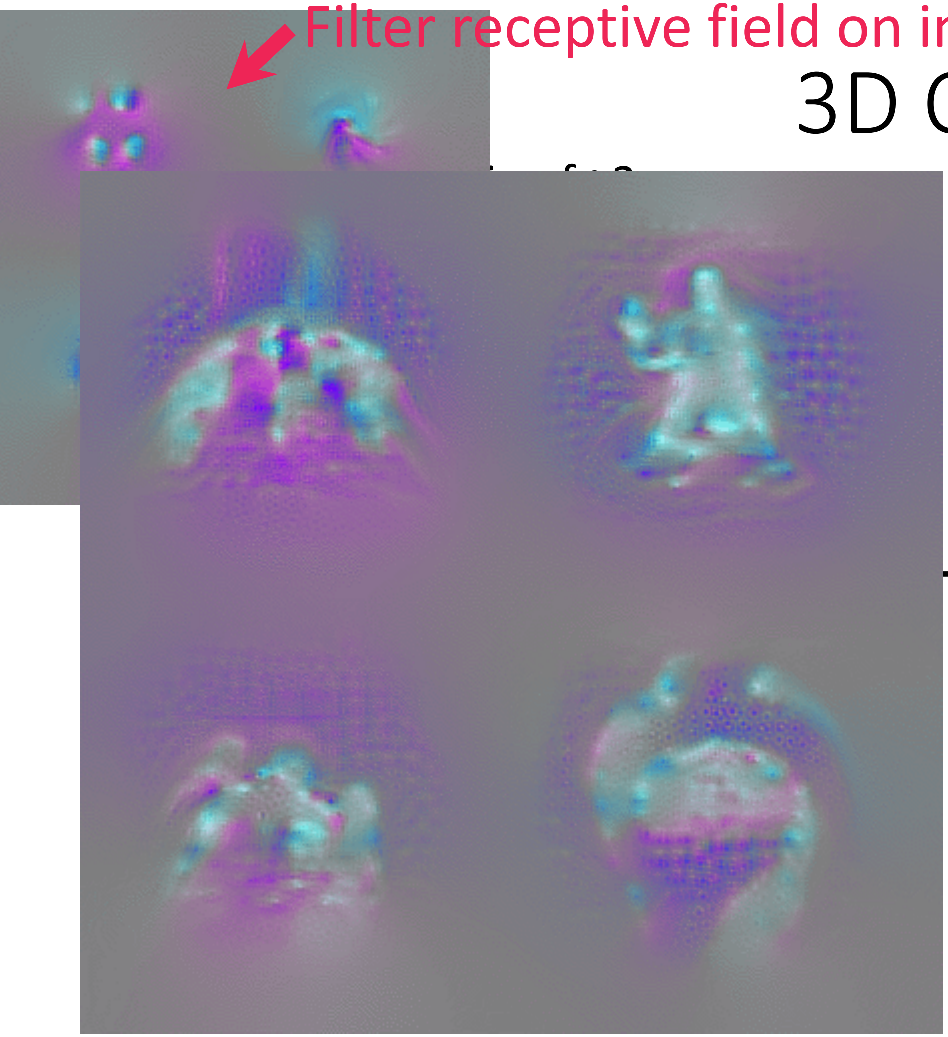
Local decomposition spatial and temporal information
[Feichtenhofer et al. 2016, Qiu et al. 2017, Tran et al. 2018, Xie et al. 2018, ...]



Global decomposition spatial and temporal information
SlowFast networks contrast features of different framerate and channel capacity
[Feichtenhofer et al. 2019]

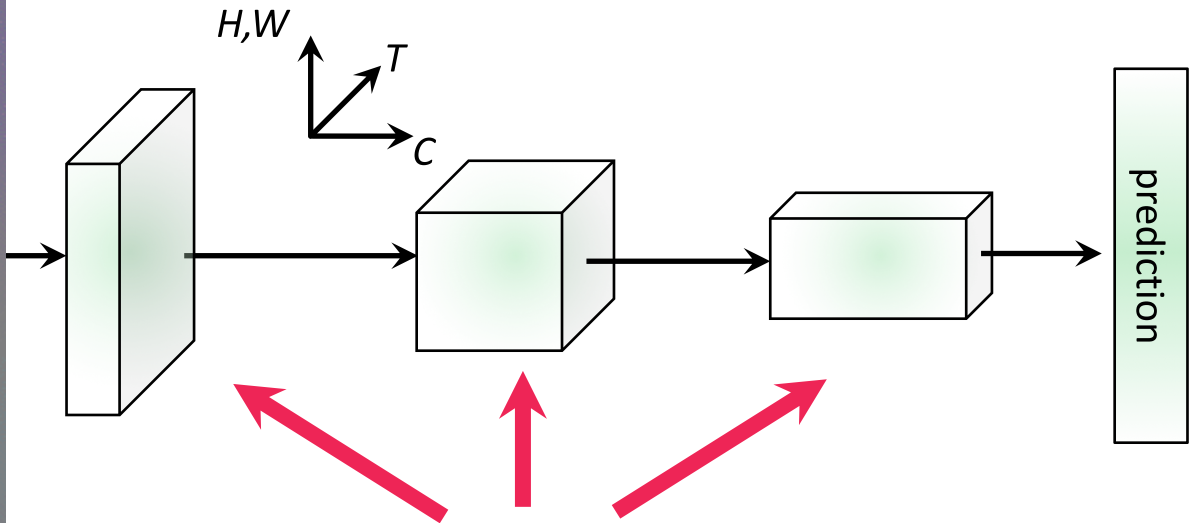
3D Convolutional Networks

Filter receptive field on input



(Kinetics classification annotation)

4D tensors of shape T x H x W x C



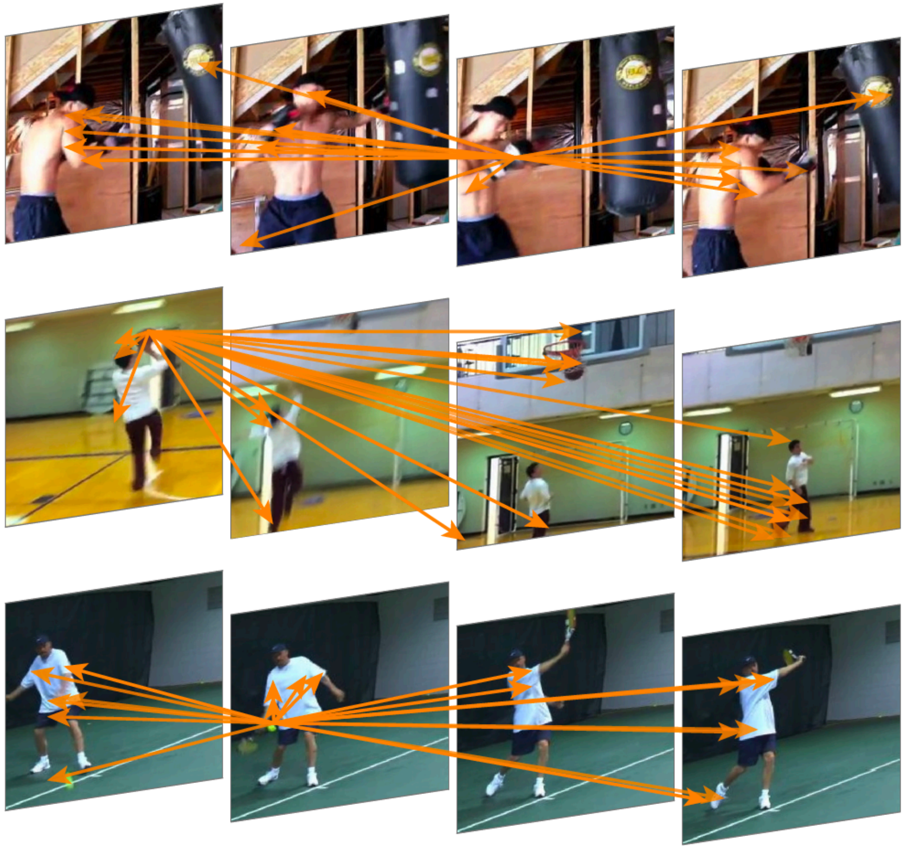
Intermediate filters only capture *local* information (in x,y,t) with a growing receptive field size

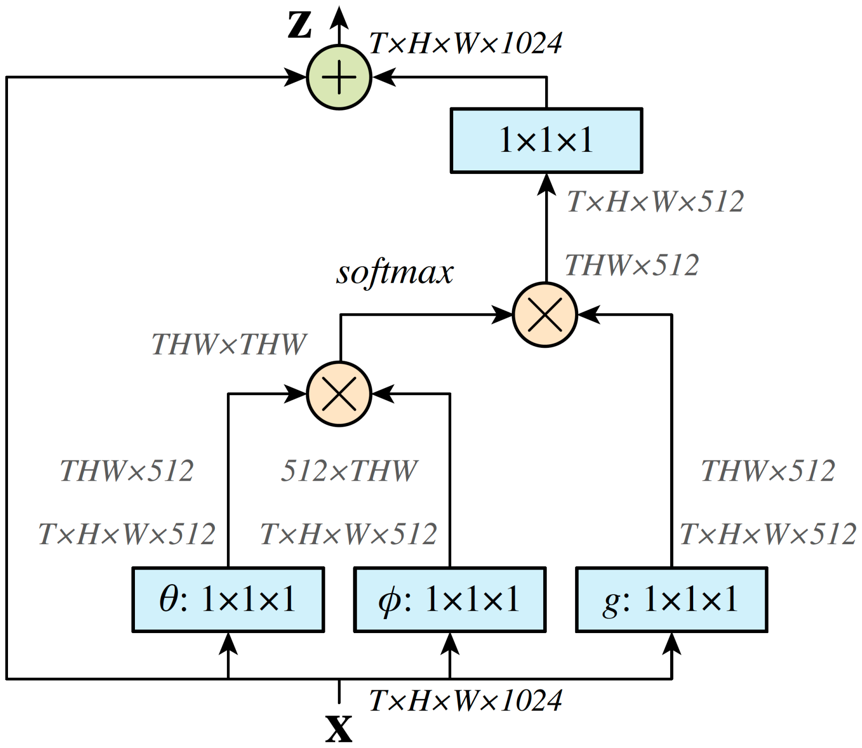
G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. Convolutional learning of spatio-temporal features. In Proc. ECCV, 2010.
D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.
J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In Proc. CVPR, 2017.

Pytorch code now available:

https://github.com/facebookresearch/SlowFast

Non-Local Blocks

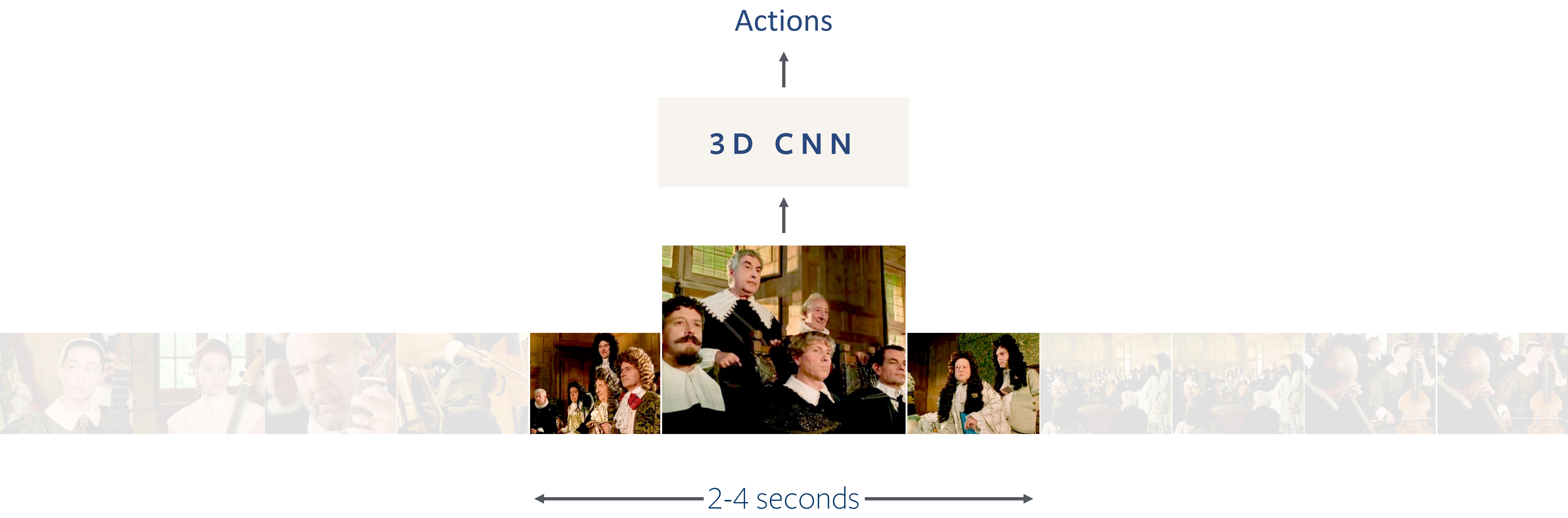




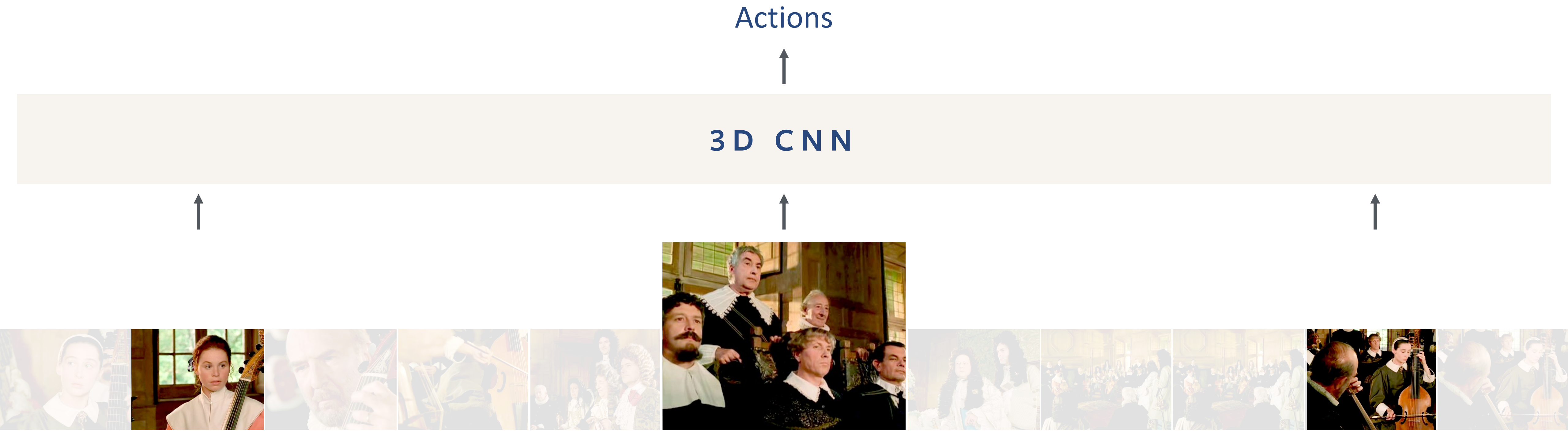
- Self-attention in the spatiotemporal domain allows long-range feature aggregation

X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In Proc. CVPR, 2018.
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., Attention is all you need. NeurIPS 2017

Limited temporal input length of 3D ConvNets

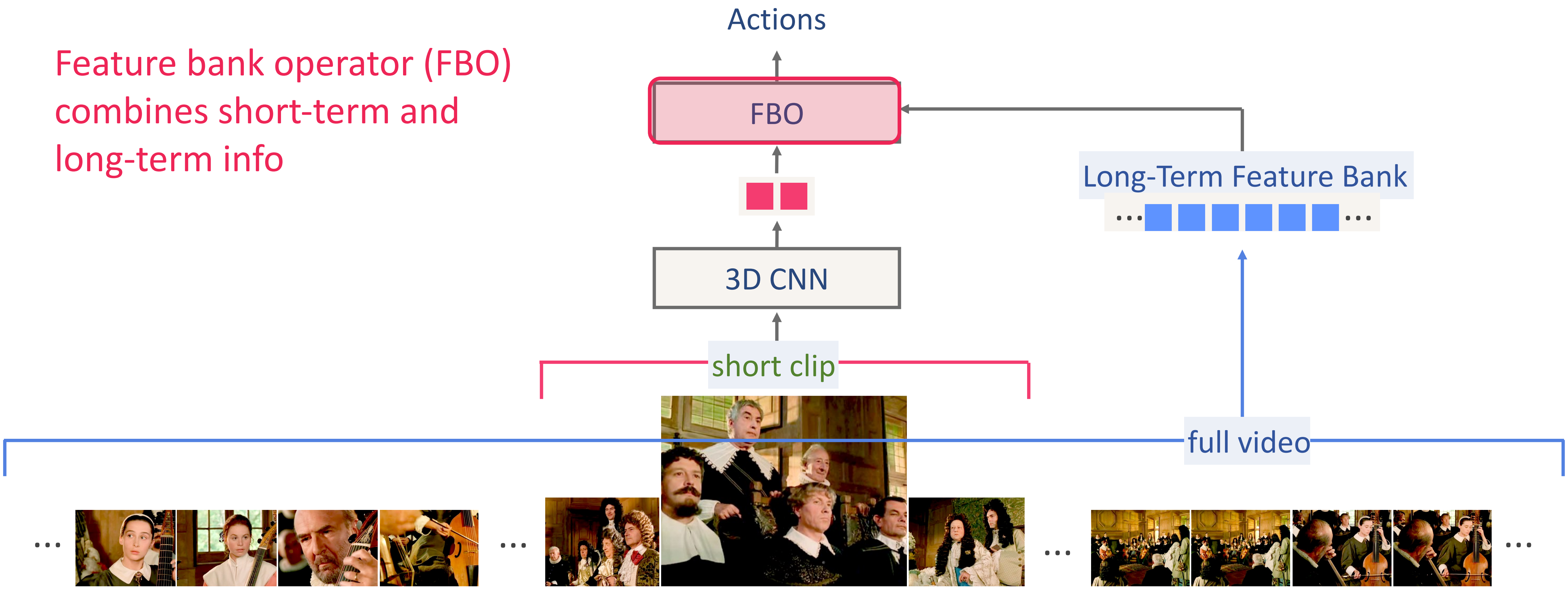


Temporal striding (subsampling)



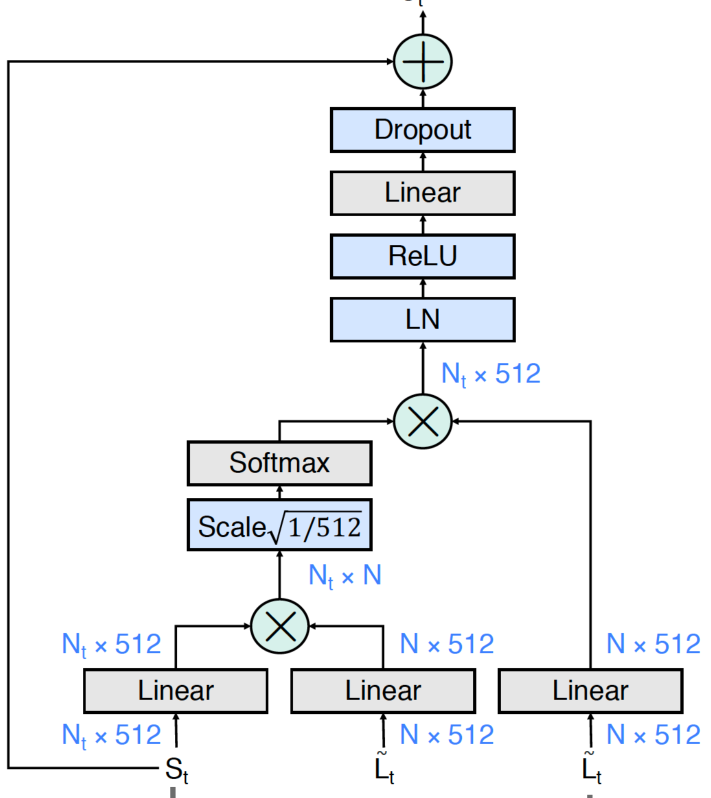
Long-Term Feature Banks for Video Understanding

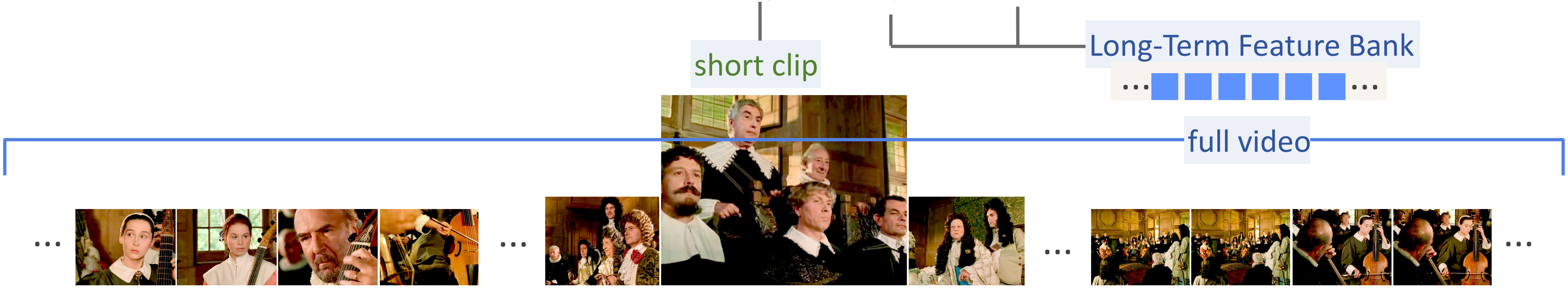
Feature bank operator (FBO) combines short-term and long-term info



CY Wu, C. Feichtenhofer, H. Fan, K. He, P. Krähenbühl, R. Girshick Long-Term Feature Banks for Detailed Video Understanding. In Proc. CVPR, 2019.

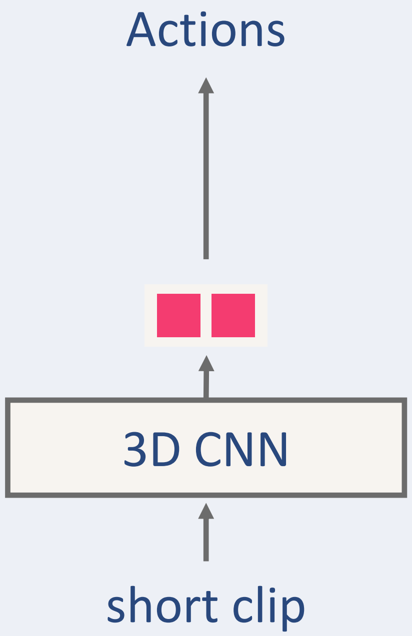
Long-Term Feature … Understanding



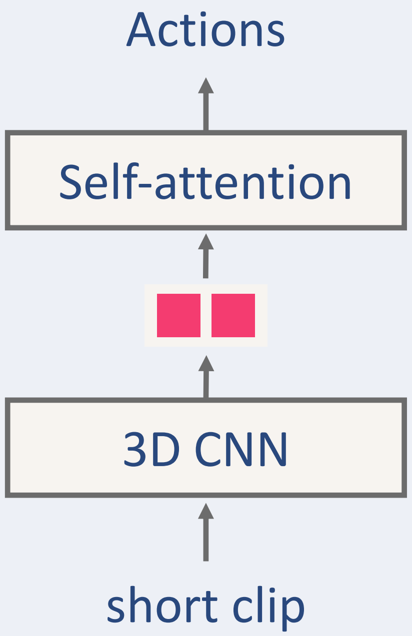


CY Wu, C. Feichtenhofer, H. Fan, K. He, P. Krähenbühl, R. Girshick Long-Term Feature Banks for Detailed Video Understanding. In Proc. CVPR, 2019.

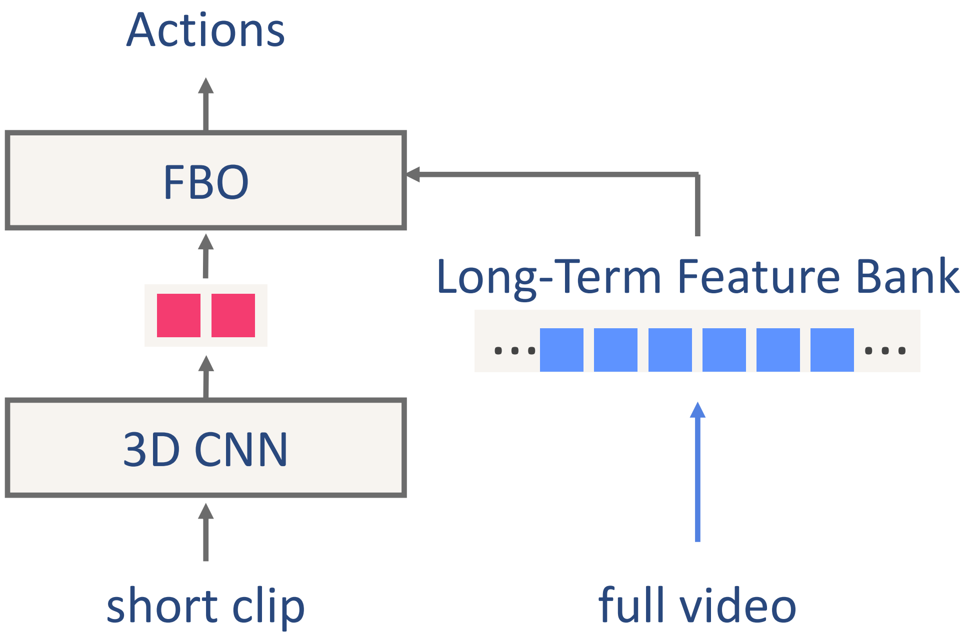
Baseline 1: 3D CNN x2



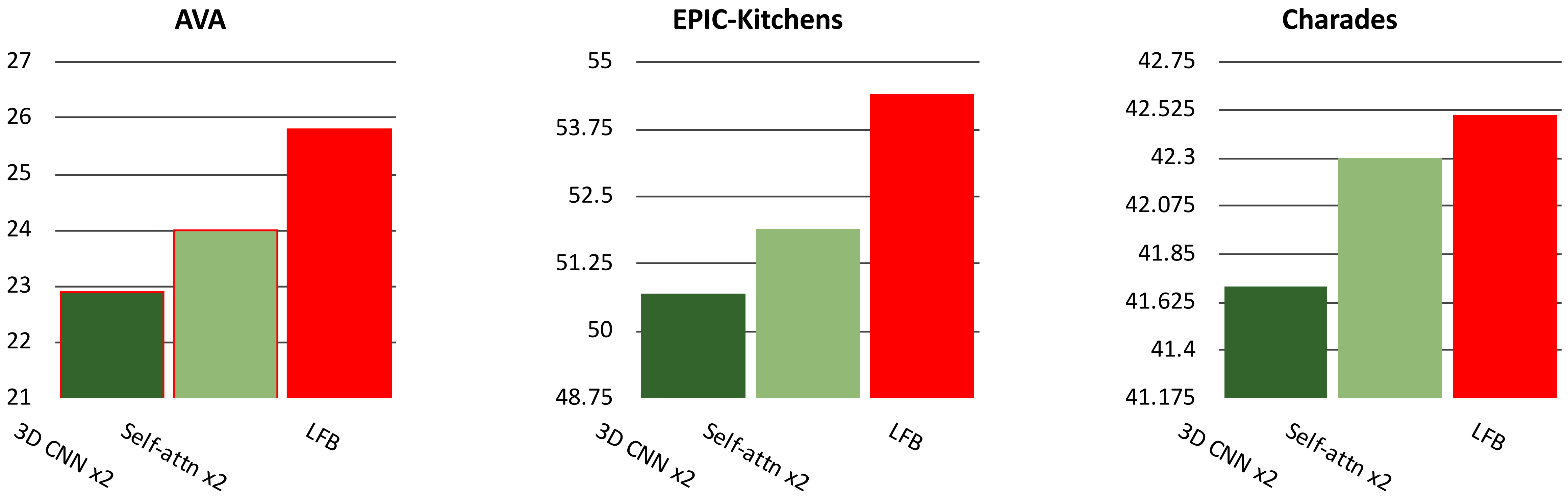
Baseline 2: Self-attention x2



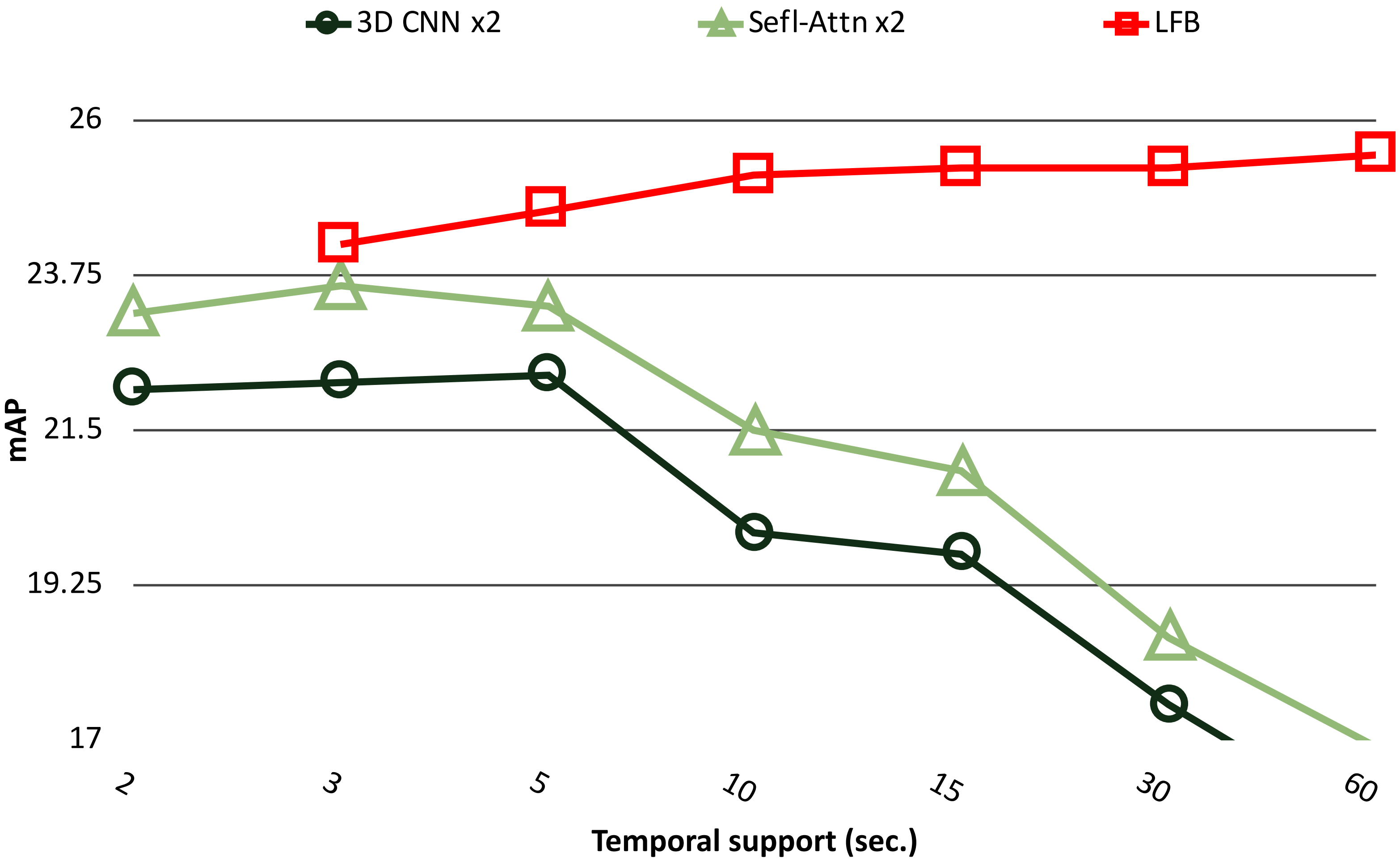
LFB



Ablation study on short-term vs. long-term

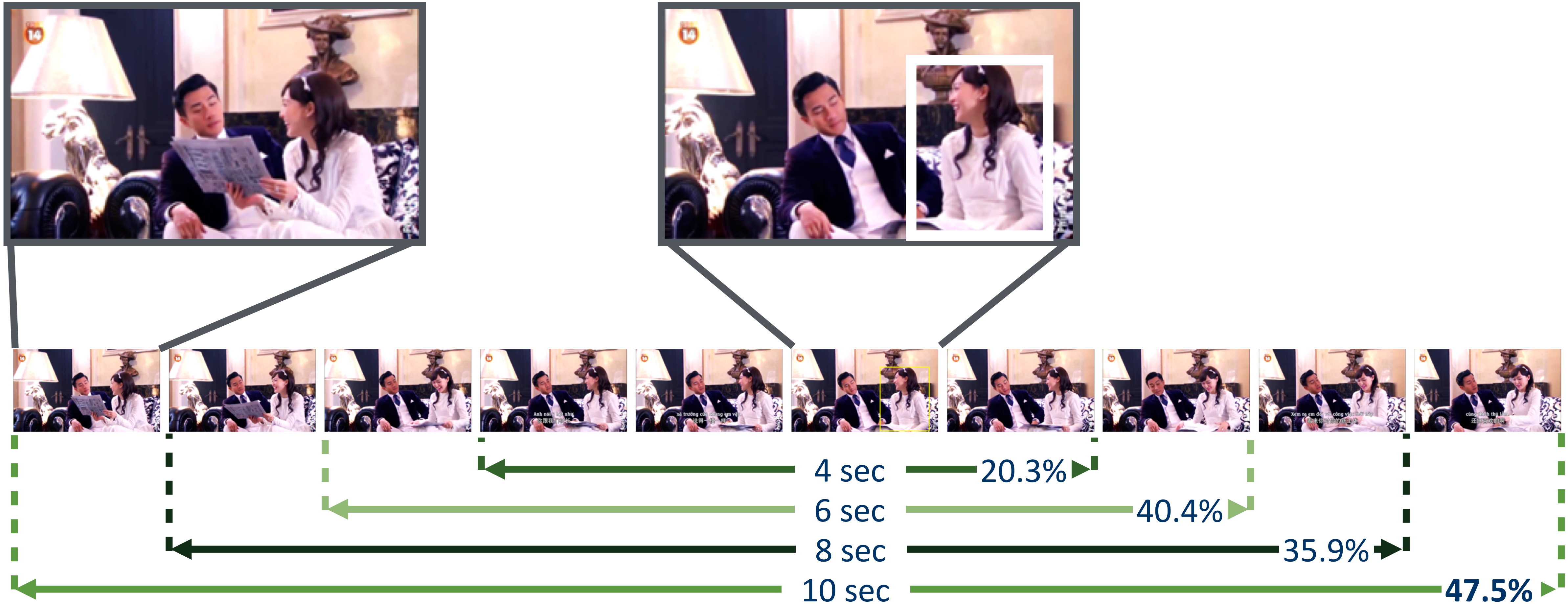


Ablation on input duration: subsampling vs LFB



Code/models:

https://github.com/facebookresearch/video-long-term-feature-banks



P(**holding an object**)?

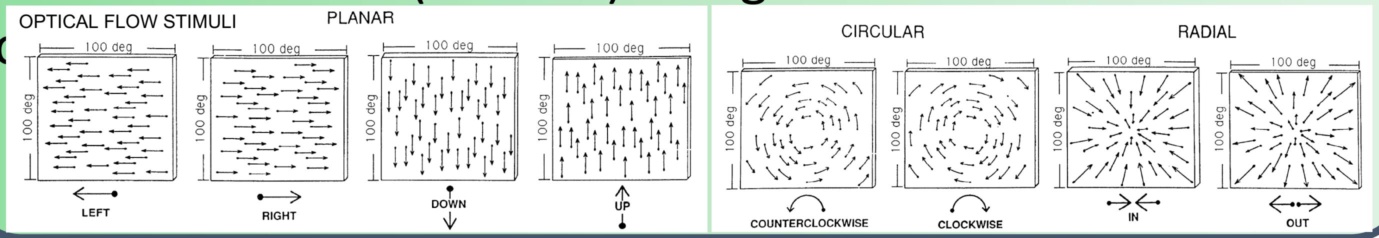
Johanssoı otion

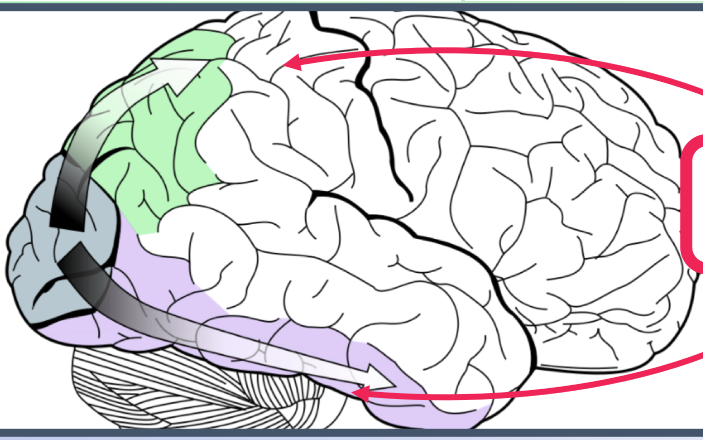
➔ **Amazing what a human brain can do without appearance information**

Sources: Johansson, G. "Visual perception of biological motion and a model for its analysis." Perception & Psychophysics. 14(2):201-211. 1973.

Motivation: Separate visual pathways in nature

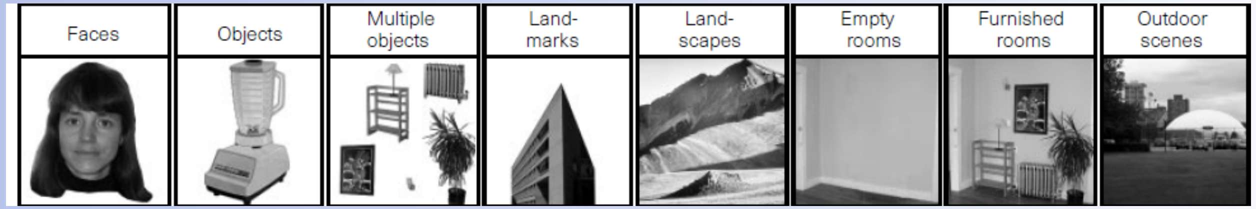
➔ Dorsal stream ('where') recognizes motion and locates





➔ **"Interconnection"**
e.g. in STS area

➔ Ventral ('what') stream performs object recognition



Sources: *"Sensitivity of MST neurons to optic flow stimuli. I. A continuum of response selectivity to large-field stimuli." Journal of neurophysiology 65.6 (1991).*
"A cortical representation of the local visual environment", Nature. 392 (6676): 598–601, 2009
https://en.wikipedia.org/wiki/Two-streams_hypothesis

Two-Stream Convolutional Networks

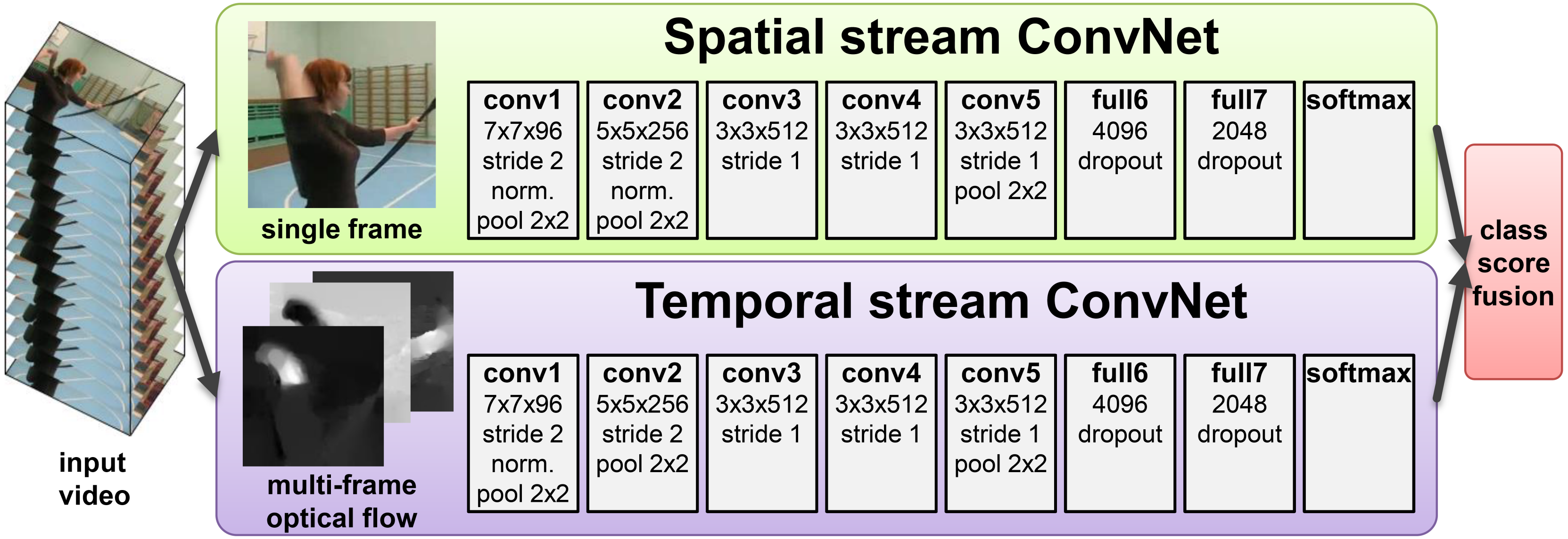


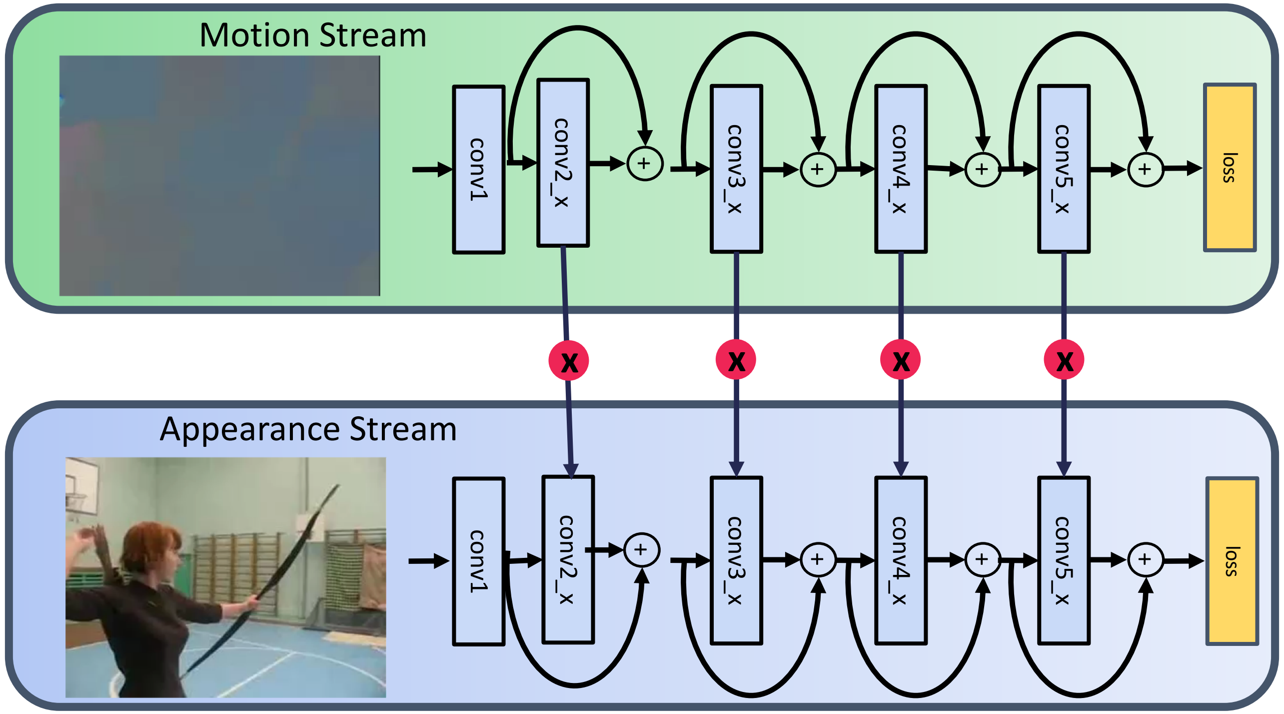
Figure 1: **Two-stream architecture for video classification.**

Individual processing of spatial and temporal information

- Using a separate **2D (x,y)** ConvNet recognition stream for each
- Late fusion via softmax score averaging

K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In NIPS, 2014

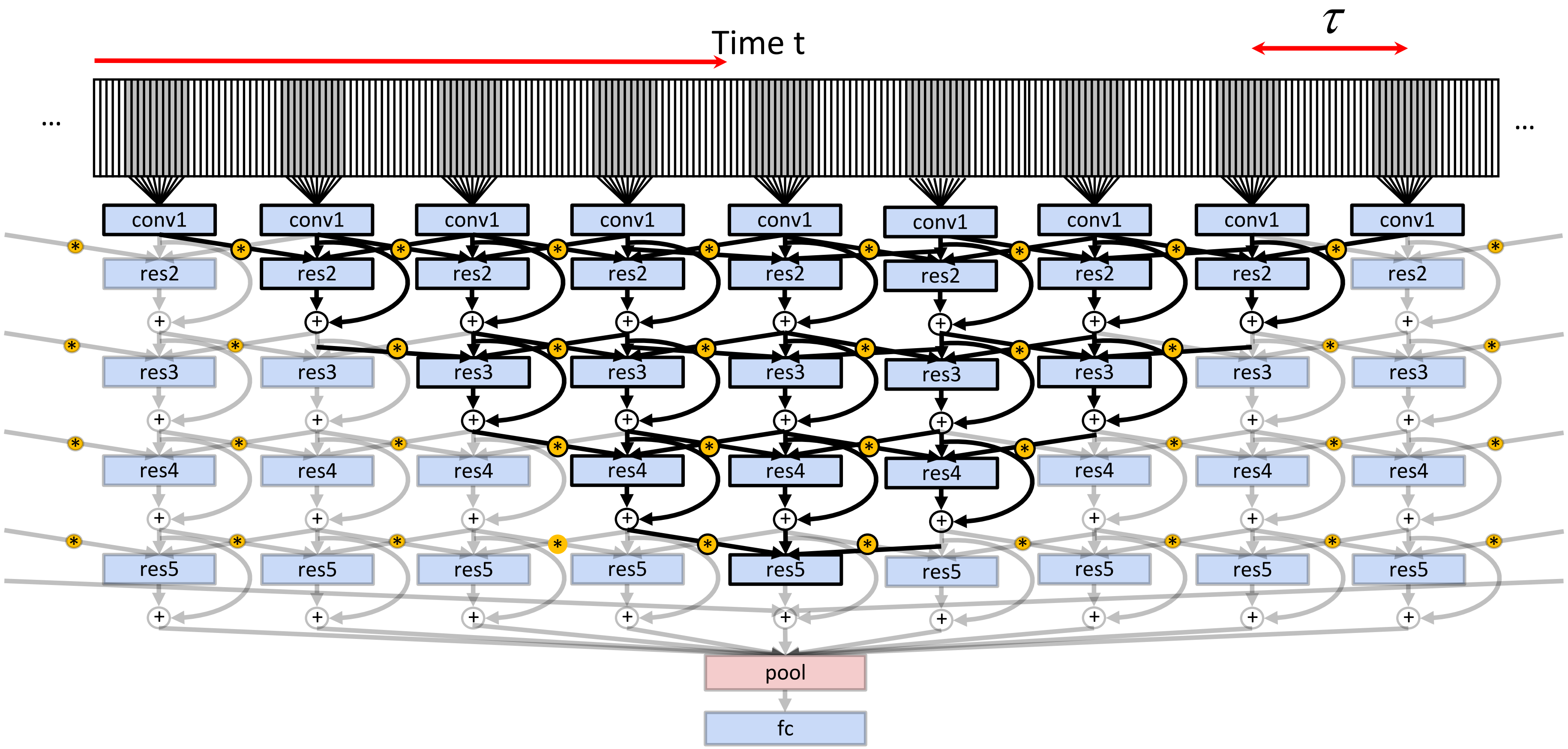
Two-Stream Network Fusion and Long-term Two-Stream networks



- ST-ResNet allows the hierarchical learning of spacetime features by connecting the appearance and motion channels of a two-stream architecture.

C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In Proc. CVPR, 2016
Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X. and Van Gool, L., Temporal segment networks: Towards good practices for deep action recognition. ECCV 2016
C. Feichtenhofer, A. Pinz, and R. Wildes. Spatiotemporal residual networks for video action recognition. In NIPS, 2016.

Long-term Two-Stream networks and transforming filters by Inflation



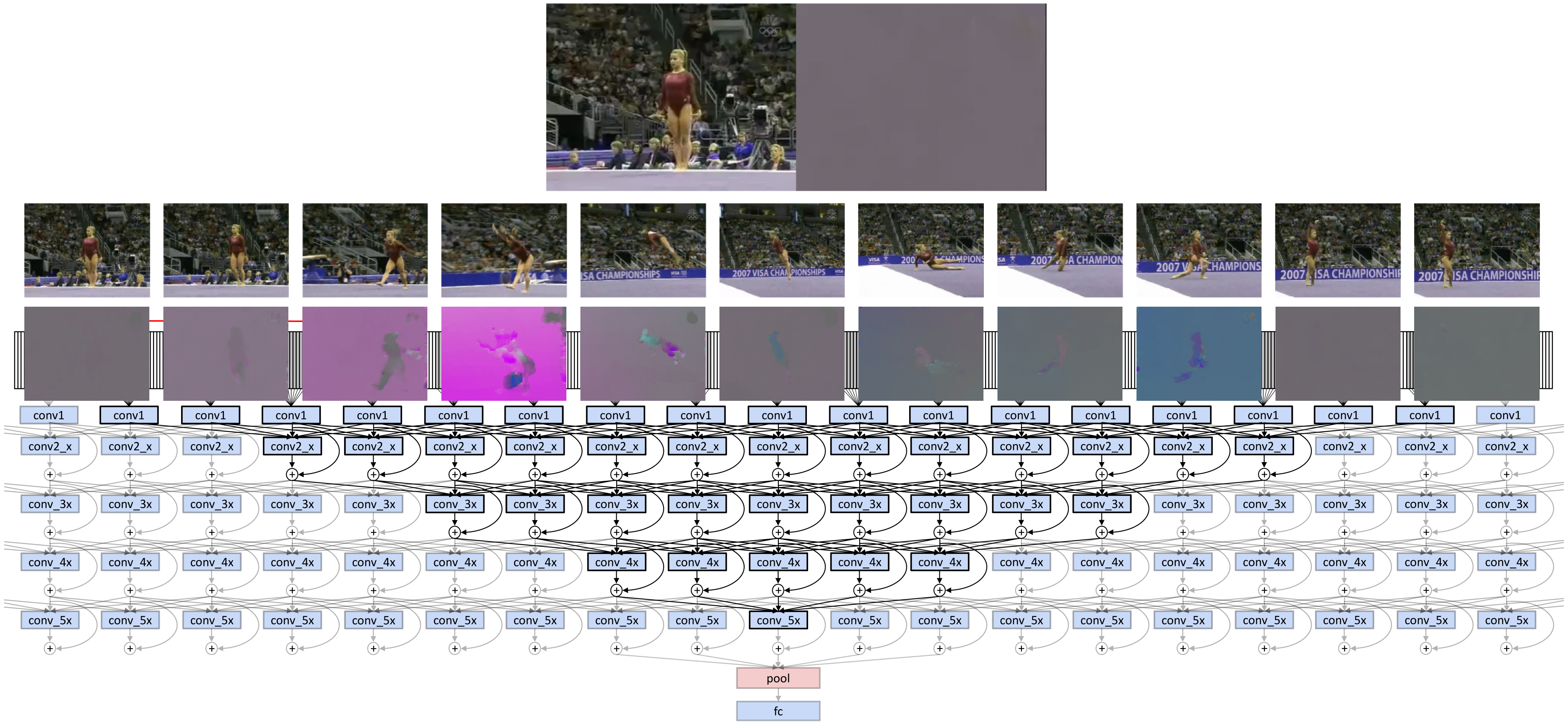
- Inflation allows to transform spatial filters to spatiotemporal ones (3D or 2D spatial +1D temporal)

C. Feichtenhofer, A. Pinz, and R. Wildes. Spatiotemporal residual networks for video action recognition. In NIPS, 2016.

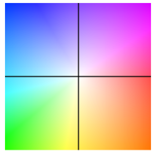
Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X. and Van Gool, L., Temporal segment networks: Towards good practices for deep action recognition. ECCV 2016

J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In Proc. CVPR, 2017.

Two-Stream Network Fusion and Long-term Two-Stream networks



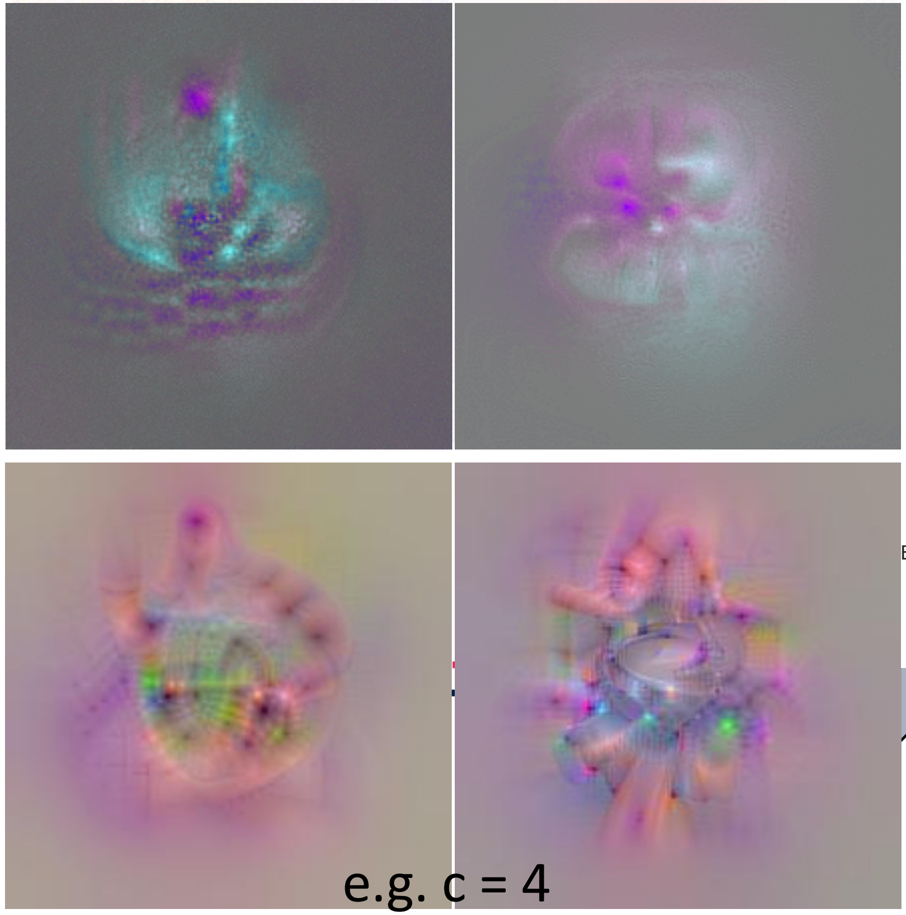
C. Feichtenhofer, A. Pinz, and R. Wildes. Spatiotemporal residual networks for video action recognition. In NIPS, 2016.
Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X. and Van Gool, L., Temporal segment networks: Towards good practices for deep action recognition. ECCV 2016
J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In Proc. CVPR, 2017.

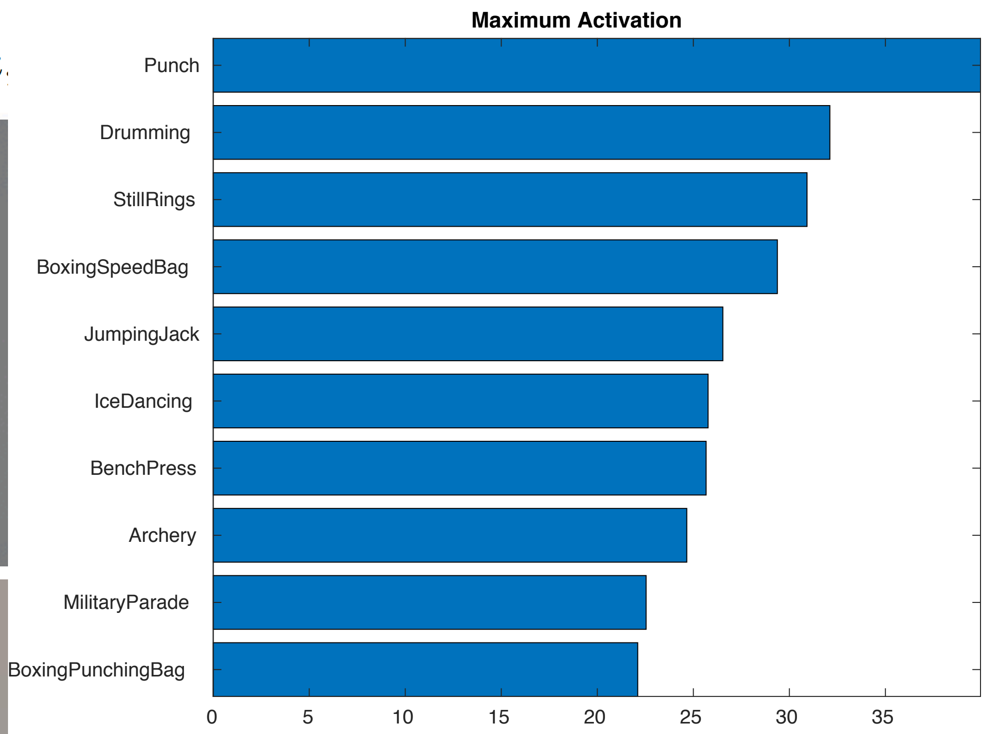


Visualizing the learned representation

Slow motion
(high temporal reg.)

Fast motion
(low temporal reg.)





e.g. c = 4

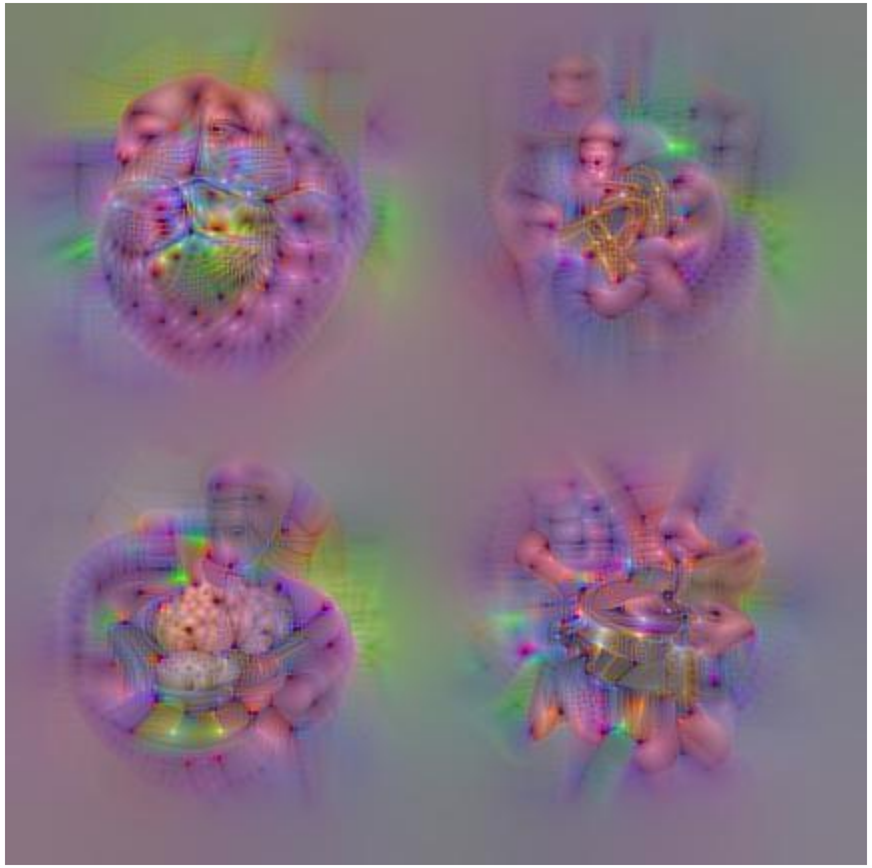
C. Feichtenhofer, A. Pinz, and R. Wildes, A. Zisserman. What have we learned … ition?. In CVPR, 2018.
C. Feichtenhofer, A. Pinz, and R. Wildes, A. Zisserman. Deep insights into convolutional networks for video recognition?. In IJCV, 2019.

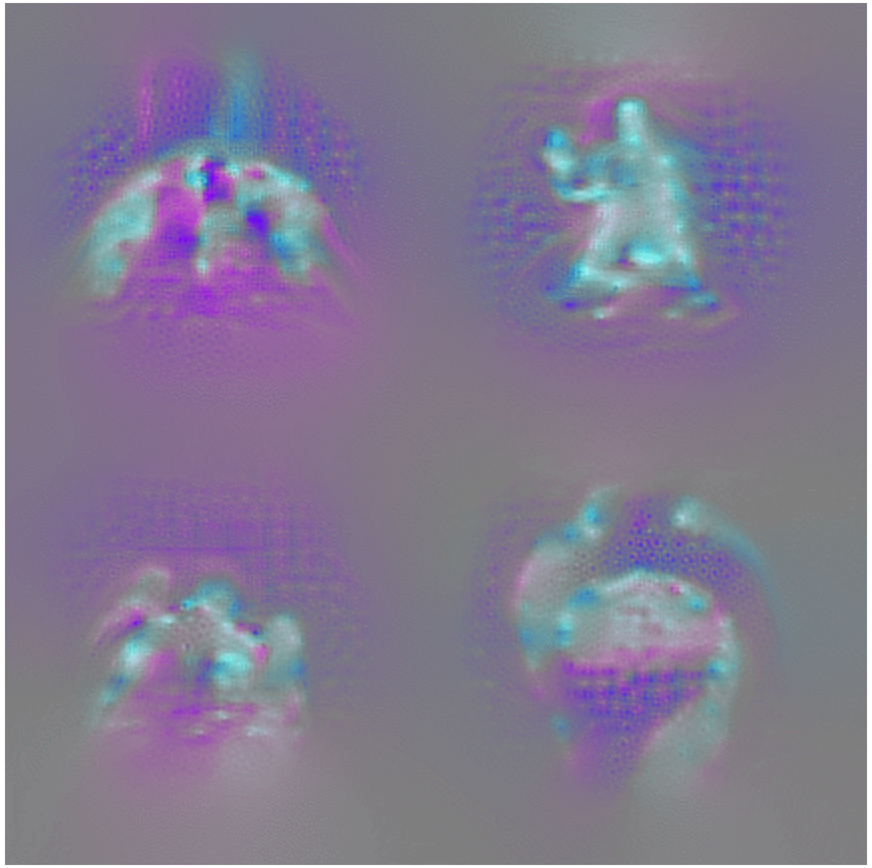
Going through the conv layers of VGG-16 (first four filters of each layer are shown)

Appearance

conv5_3 f1-4

Motion

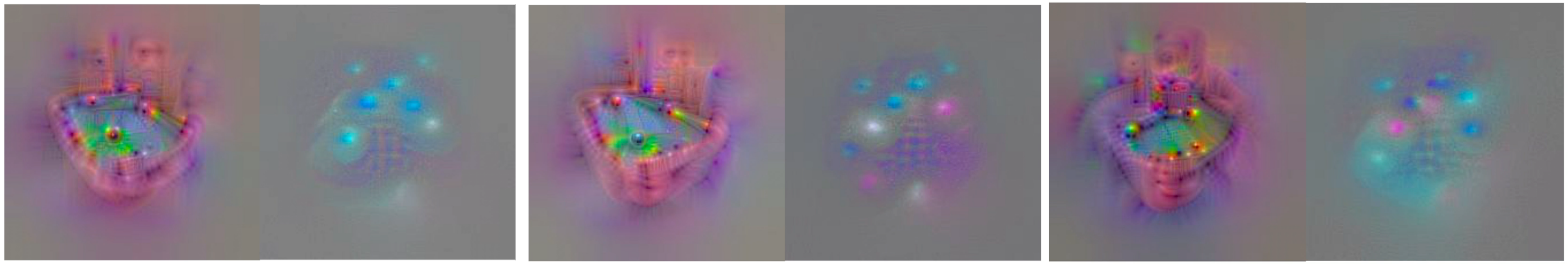


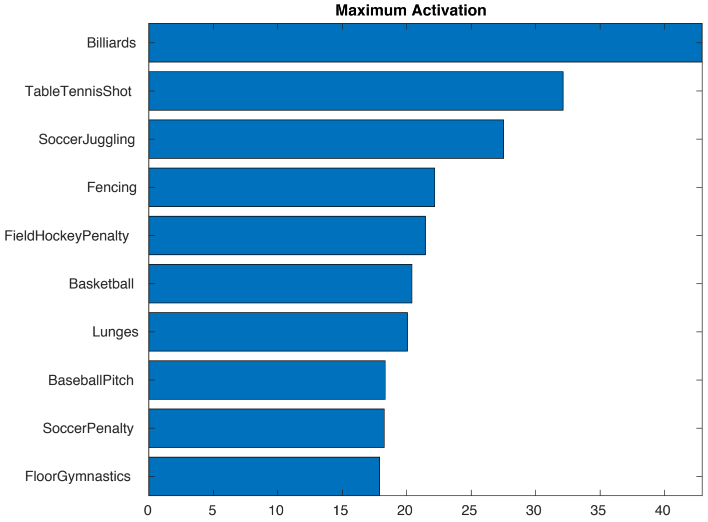


Feichtenhofer, Pinz, Wildes, Zisserman, CVPR 2018, IJCV 2019

Filter #251 at conv5 fusion – the strongest local Billiards unit

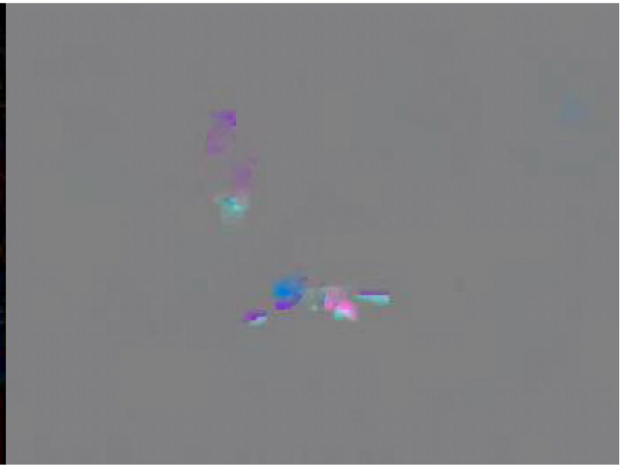
slow medium fast





(c) test set activity

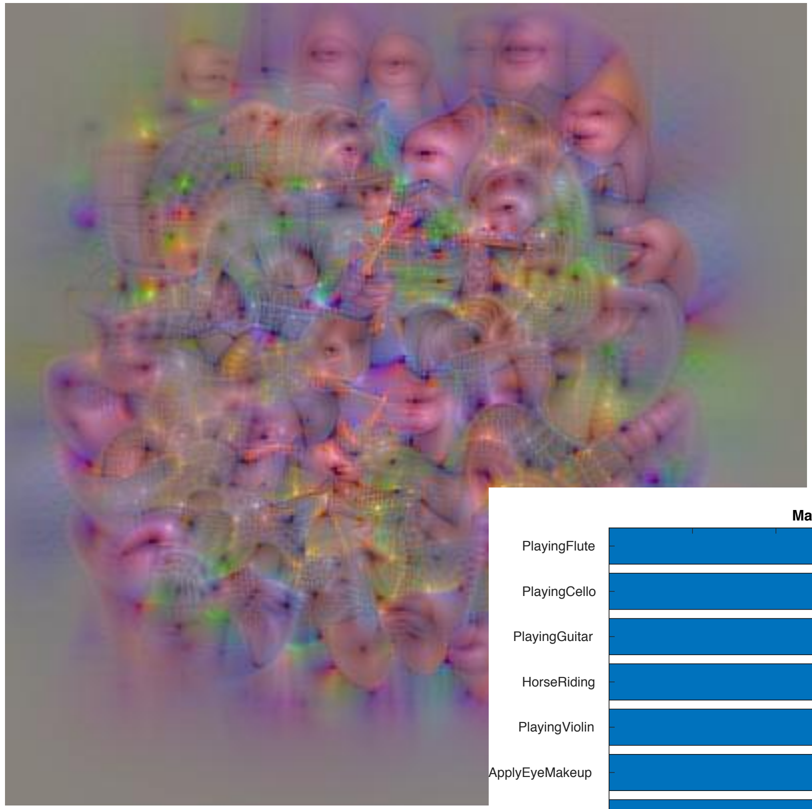
(d)



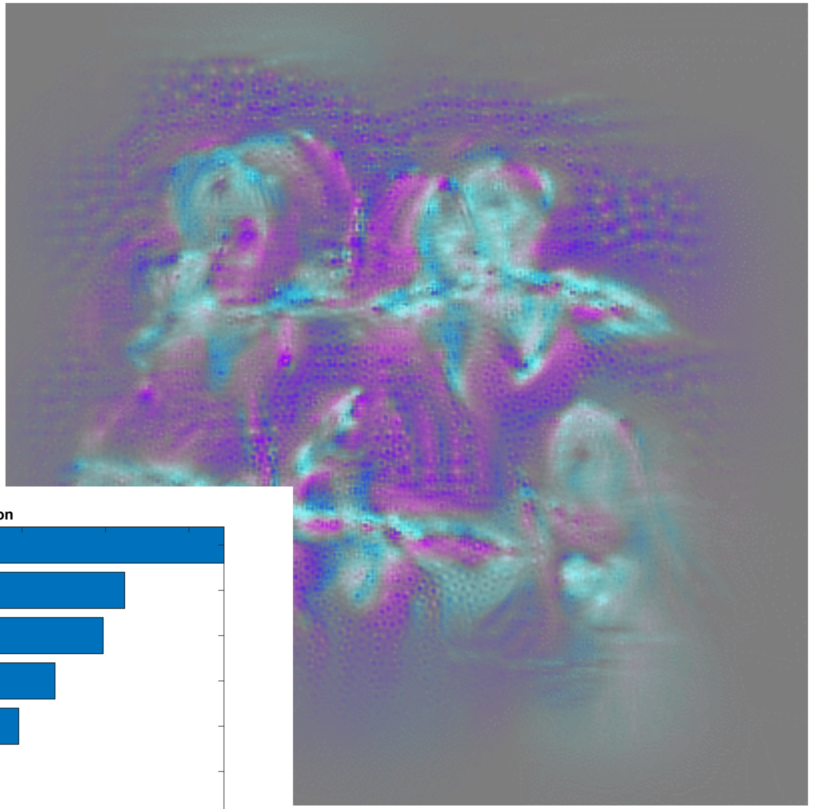
(e)

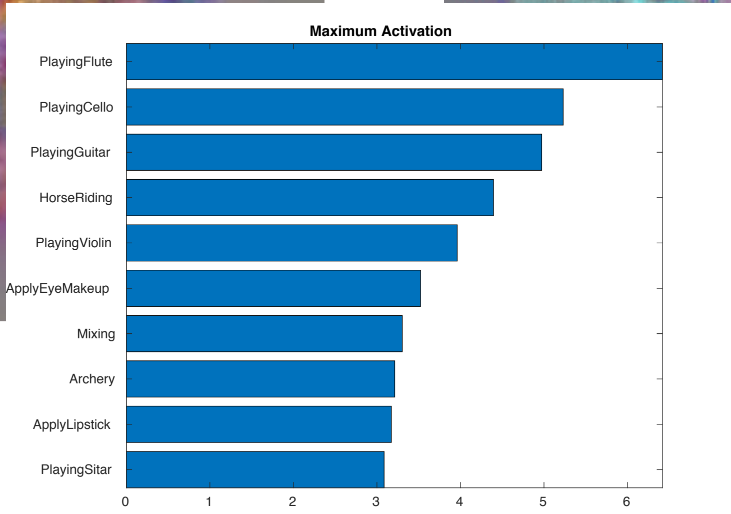
FC 6 (4096 features; RF 404x404)

Appearance



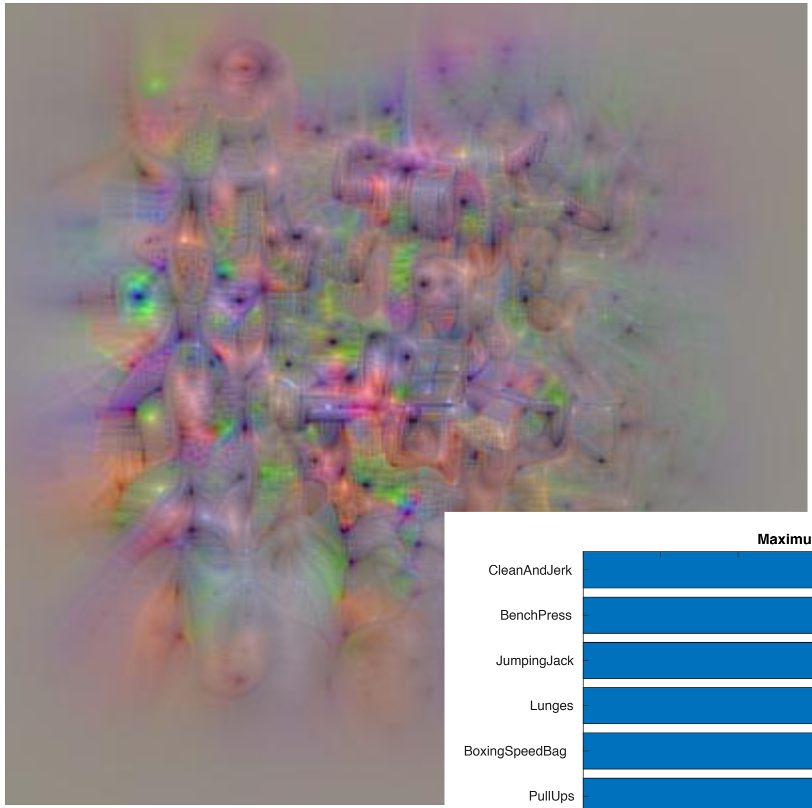
Slow motion



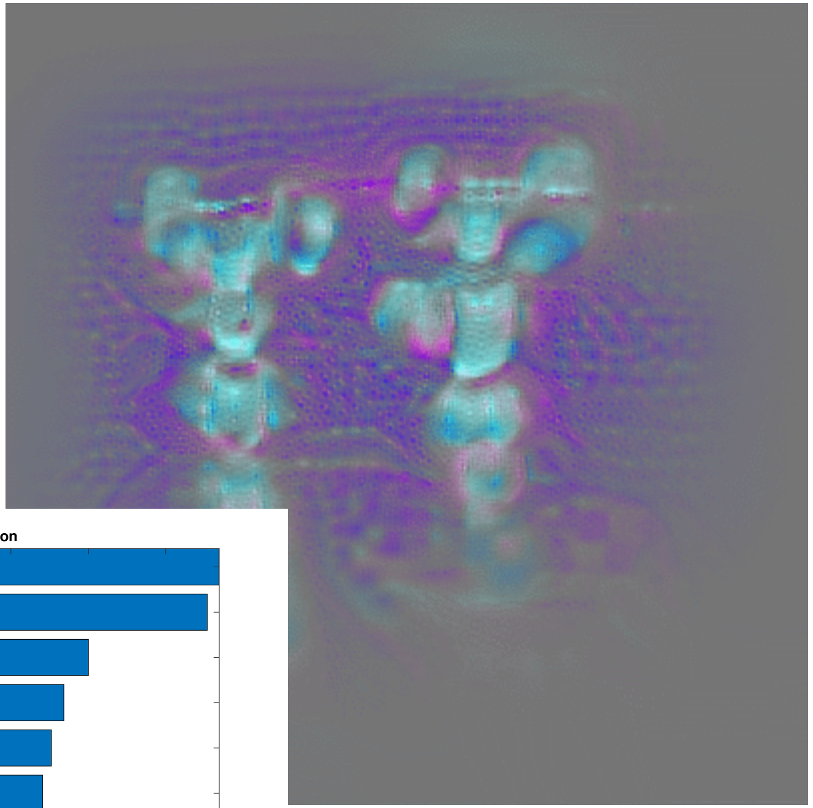


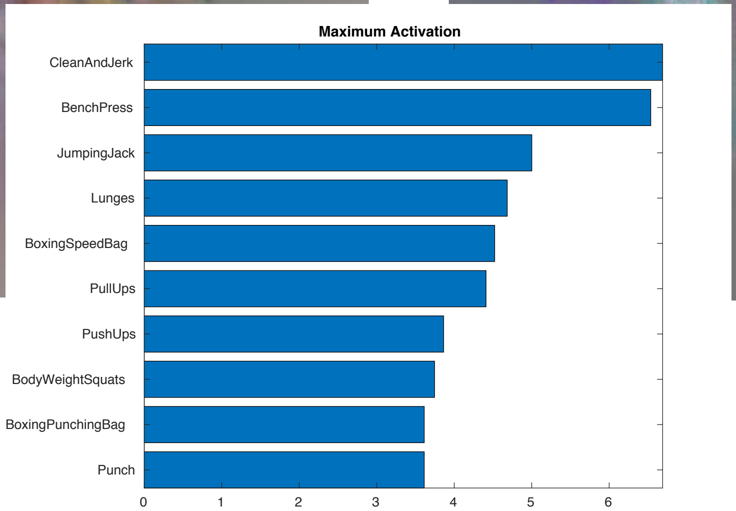
FC 7 (4096 features; RF 404x404)

Appearance



Slow motion





Last layer

→
"CleanAndJerk"

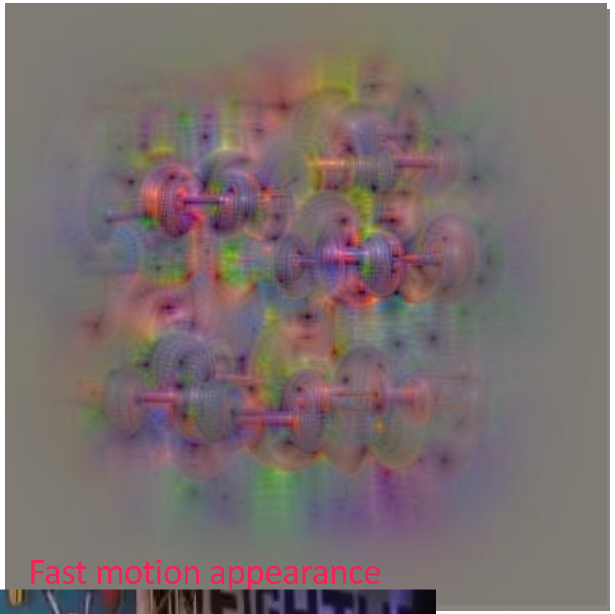
Appearance

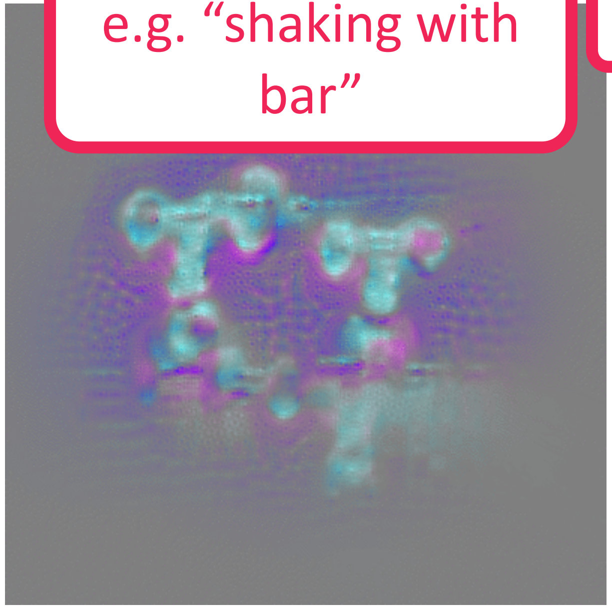
Slow motion

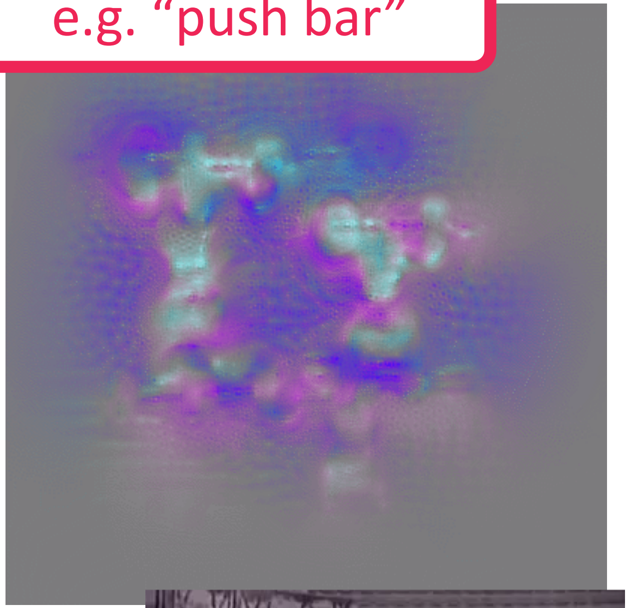
Fast motion

e.g. "shaking with bar"

e.g. "push bar"

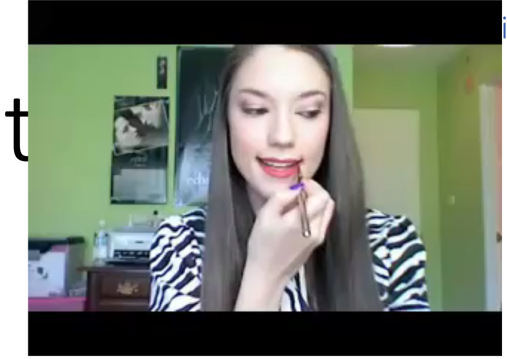






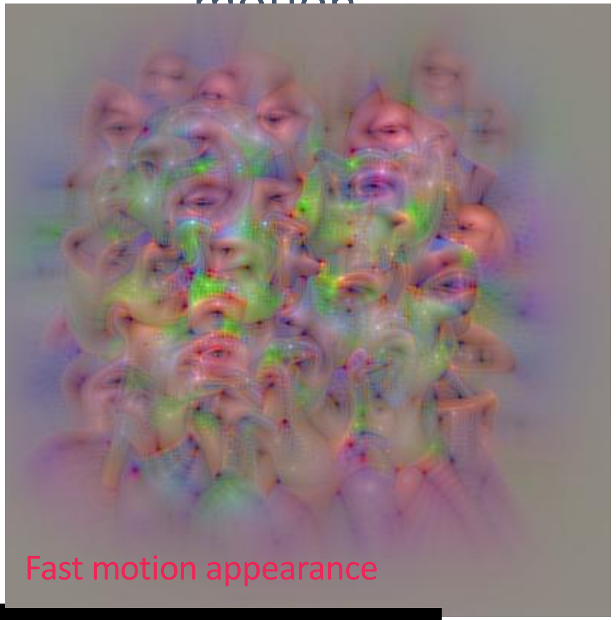
Fast motion appearance

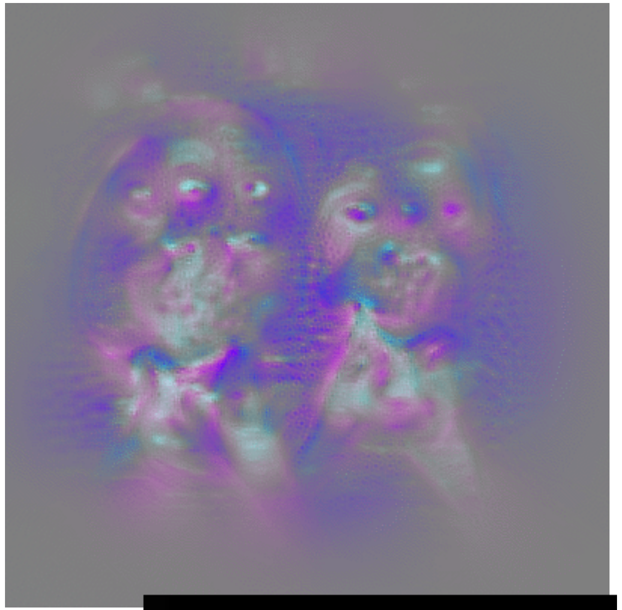
ng idiosyncracies in dat

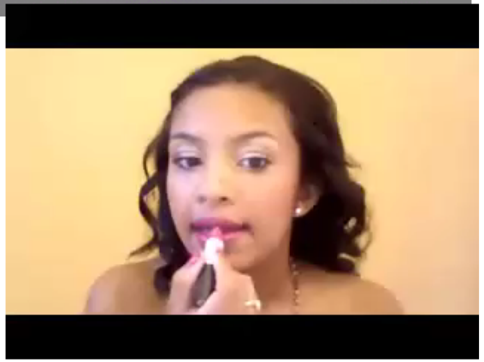


➔ "ApplyLipstick"

Appearance motion | Slow motion | Fast







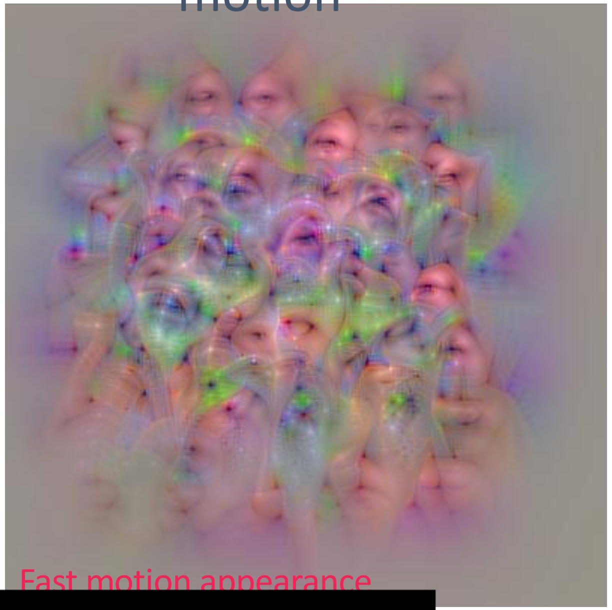
Revealing idiosyncracies in dat

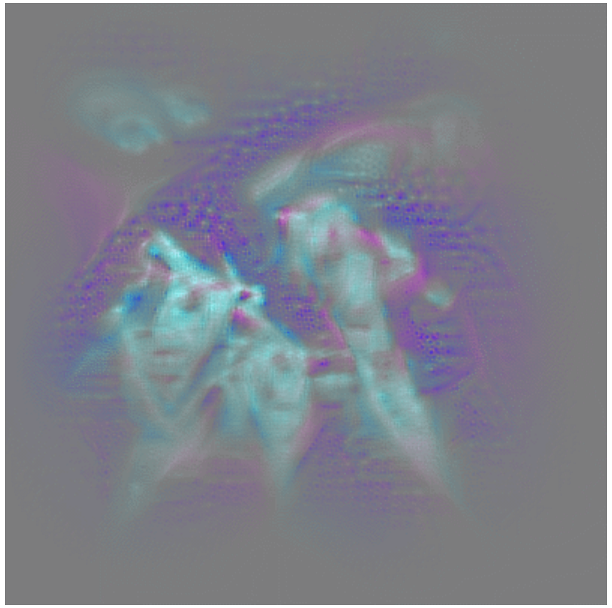
→ "ApplyEyeMakeup"

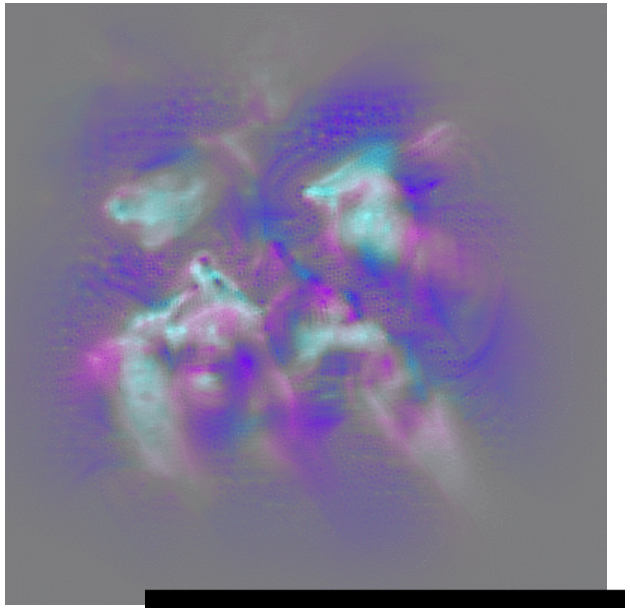
Appearance motion

Slow motion

Fast





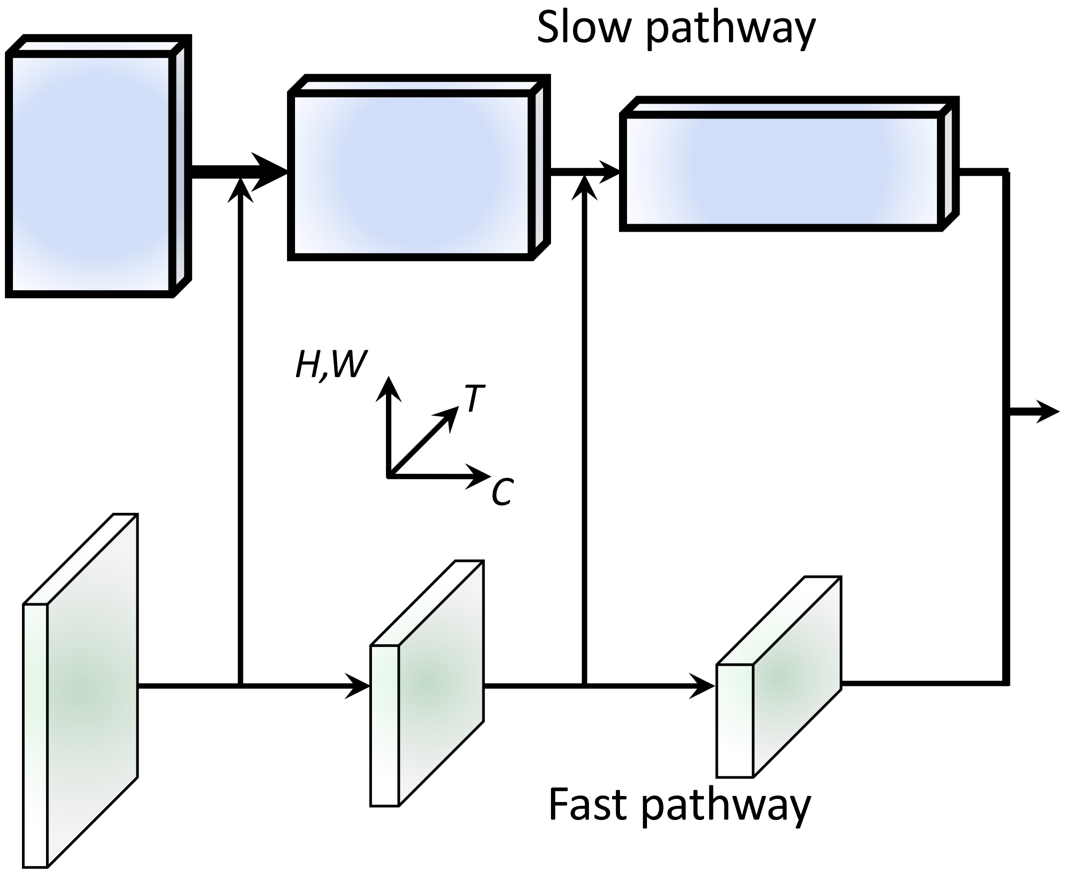


Fast motion appearance

SlowFast Networks for Video Recognition

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik and Kaiming He

- New backbone network for human action classification & detection



C. Feichtenhofer, H. Fan, J. Malik, K. He SlowFast Networks for Video Recognition. Proc. ICCV 2019

Slow frame rate

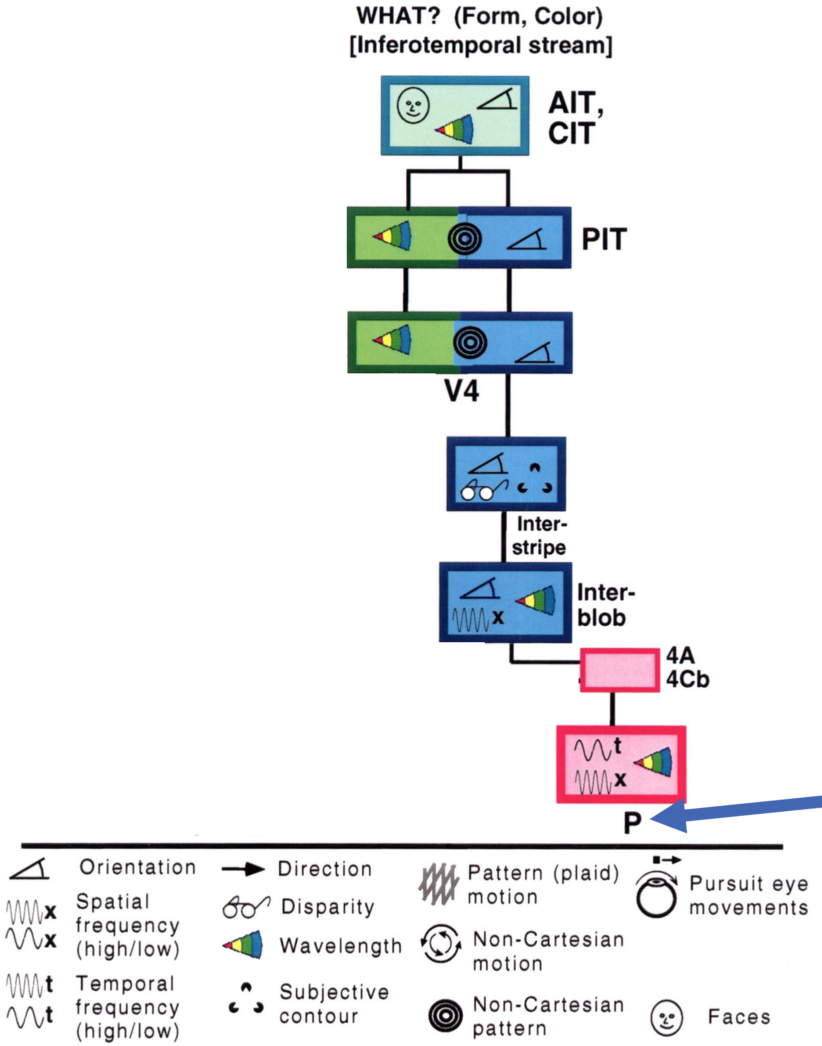
Fast frame rate

“Hand-clap”
(action detection annotation)

Human brain: Separate visual pathways

- Minority: ≈20%
- Fast conduction rate (more myelin)
- Grayscale
- Processes information about depth & motion
- Large receptive field

Magno cells



- Majority: ≈80%
- Slow conduction rate (less myelin)
- Color
- Processes information about color & detail
- Small receptive field

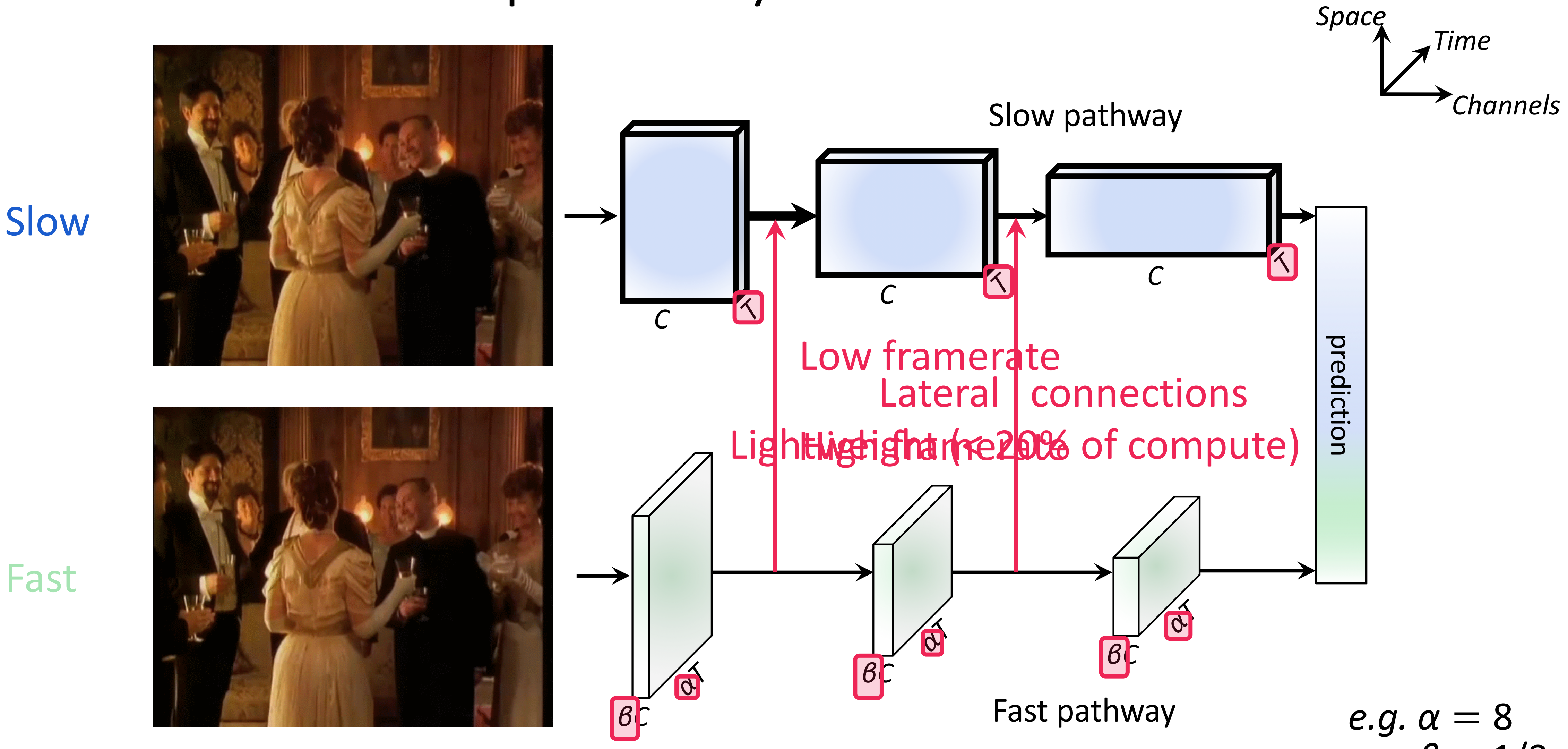
Parvo cells

David C. Van Essen, Jack L. Gallant, Neural mechanisms of form and motion processing in the primate visual system, Neuron, Volume 13, Issue 1, July 1994, Pages 1-10, ISSN 0896-6273

Basic idea: Two pathways

- Slow pathway
  - Low frame rate
  - Capturing spatial semantics
- Fast pathway
  - High frame rate
  - Capturing motion information

Basic idea: Two pathways



$e.g.\ \alpha = 8$
$\beta = 1/8$

Example instantiation of a SlowFast network

- Dimensions are $\{T\times S^2, C\}$
- Strides are {temporal, spatial^2}
- The backbone is ResNet-50
- Residual blocks are shown by brackets
- Non-degenerate temporal filters are underlined
- Here the speed ratio is $\alpha = 8$ and the channel ratio is $\beta = 1/8$
- Orange numbers mark fewer channels, for the Fast pathway
- Green numbers mark higher temporal resolution of the Fast pathway
- No temporal *pooling* is performed throughout the hierarchy

stage	*Slow* pathway	
raw clip	-	
data layer	stride 16, 1^2	
conv_1	1×7^2, 64 stride 1, 2^2	
pool_1	1×3^2 max stride 1, 2^2	
res_2	$\begin{bmatrix} 1\times1^2, 64 \\ 1\times3^2, 64 \\ 1\times1^2, 256 \end{bmatrix}\times3$	
res_3	$\begin{bmatrix} 1\times1^2, 128 \\ 1\times3^2, 128 \\ 1\times1^2, 512 \end{bmatrix}\times4$	
res_4	$\begin{bmatrix} \underline{3\times1^2}, 256 \\ 1\times3^2, 256 \\ 1\times1^2, 1024 \end{bmatrix}\times6$	
res_5	$\begin{bmatrix} \underline{3\times1^2}, 512 \\ 1\times3^2, 512 \\ 1\times1^2, 2048 \end{bmatrix}\times3$	
	global average pool, concate, fc	# classes

SlowFast ablations: Individual paths

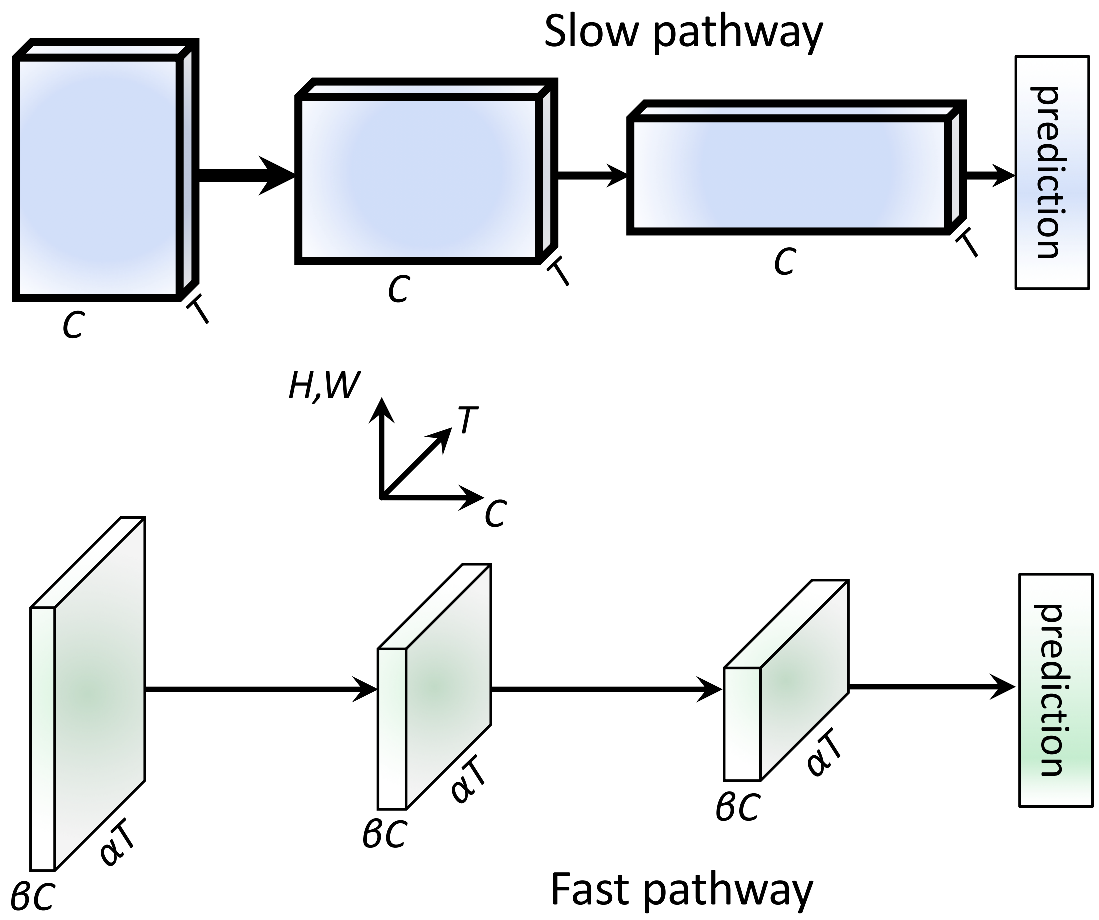
- Kinetics action classification dataset has 240k training videos and 20k validation videos in 400 classes

model	$T \times \tau$	t-reduce	top-1	top-5	GFLOPs
3D R-50	8×8	2^1	73.5	90.8	28.1
3D R-50	8×8	1	**74.6**	**91.5**	44.9
our Slow-only, R-50	4×16	1	72.6	90.3	**20.9**
our Fast-only, R-50	32×2	1	51.7	78.5	**4.9**

(b) **Individual pathways**: Training our Slow-only or Fast-only pathway alone, using the structure specified in Table 1. "t-reduce" is the total temporal downsampling factor within the network.

$\alpha = 8$

$\beta = 1/8$



SlowFast ablations: Learning curves

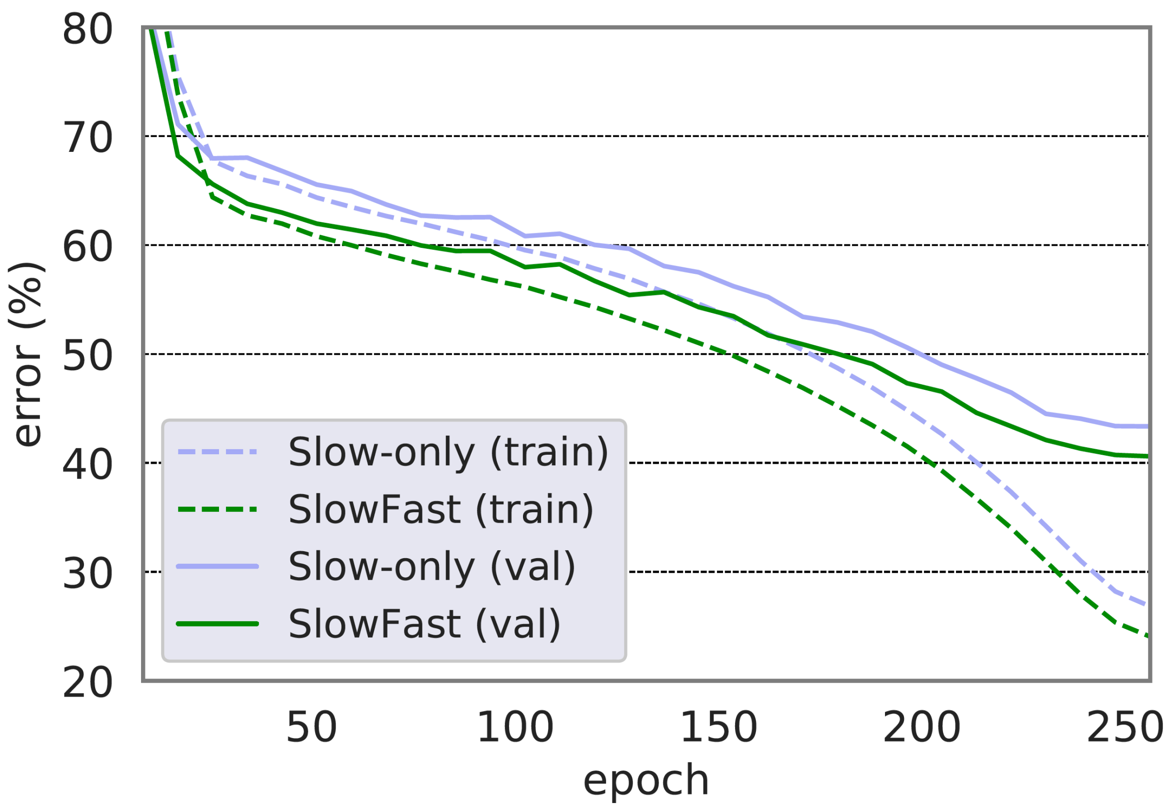
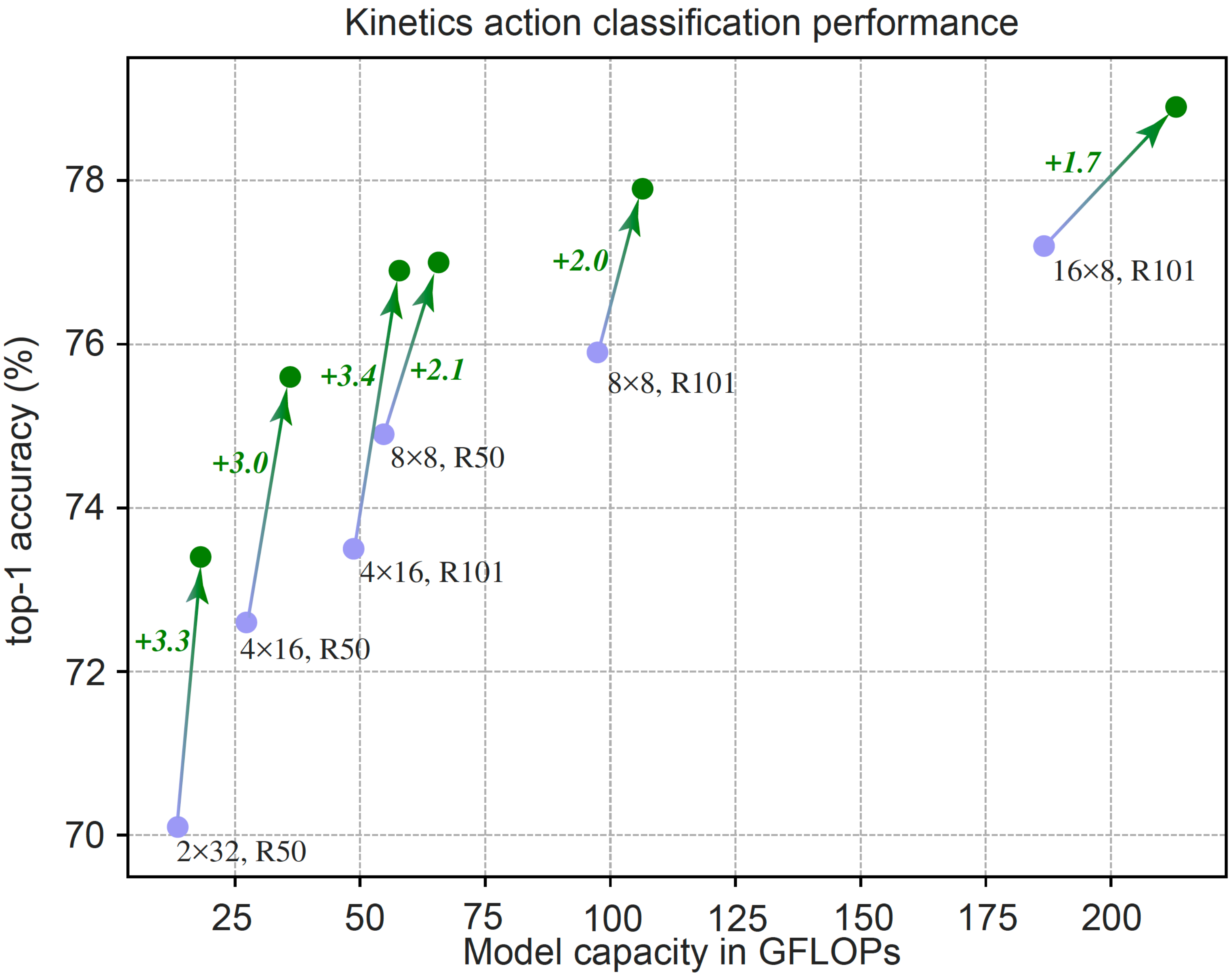
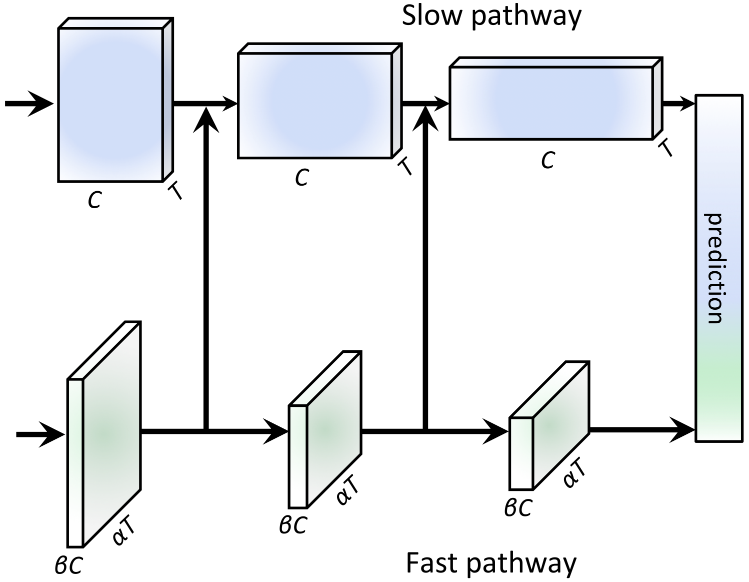


Figure 2. Training procedure on Kinetics for Slow-only (blue) *vs*. SlowFast (green) network. We show the top-1 training error (dash) and validation error (solid). The curves are single-crop *errors*; the video *accuracy* is 72.6% *vs*. 75.6% (see also Table 2c).

SlowFast ablations: Video action classification



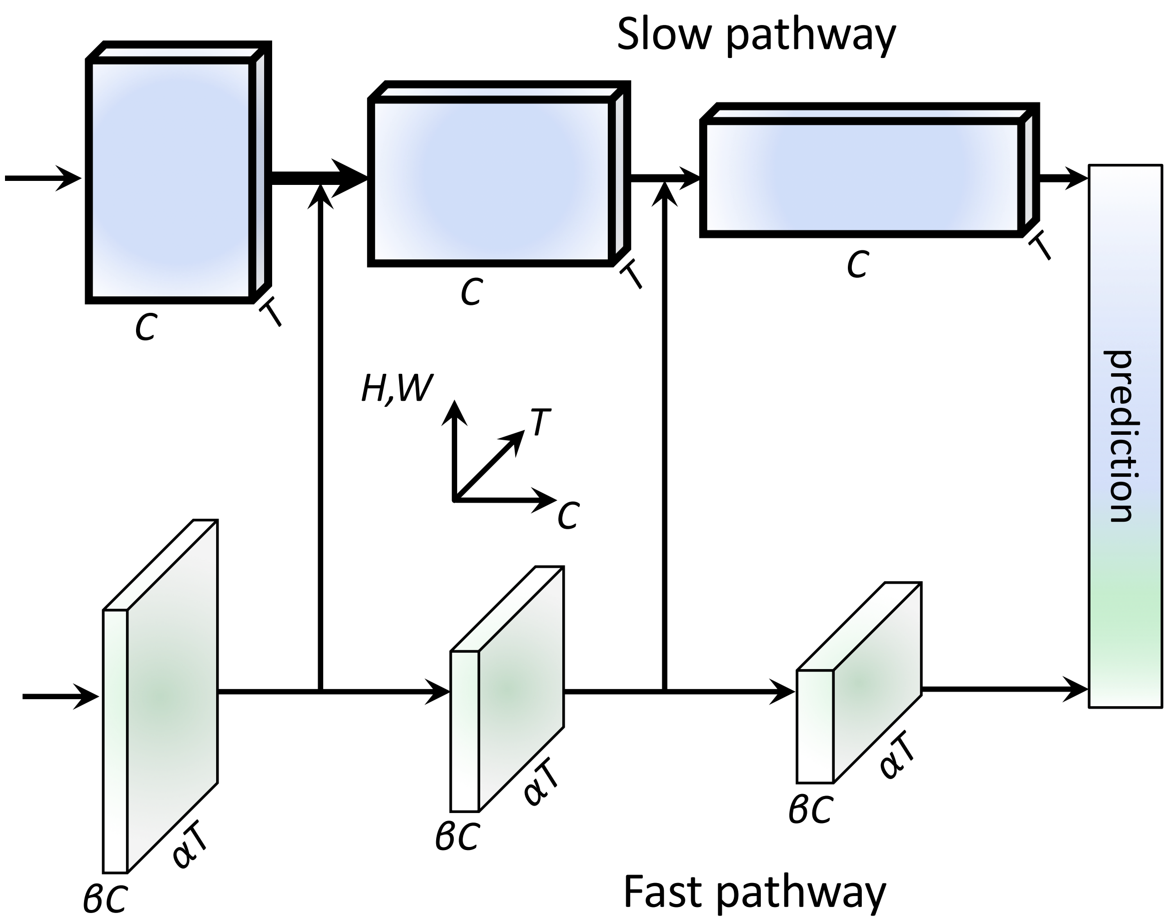


SlowFast ablations: Making the Fast path thin in channel dimension

- Kinetics dataset has 240k training videos and 20k validation videos in 400 classes

	top-1	top-5	GFLOPs
Slow-only	72.6	90.3	20.9
$\beta = 1/4$	75.6	91.7	41.7
1/6	**75.8**	92.0	32.0
1/8	75.6	**92.1**	27.6
1/12	75.2	91.8	25.1
1/16	75.1	91.7	23.4
1/32	74.2	91.3	21.9

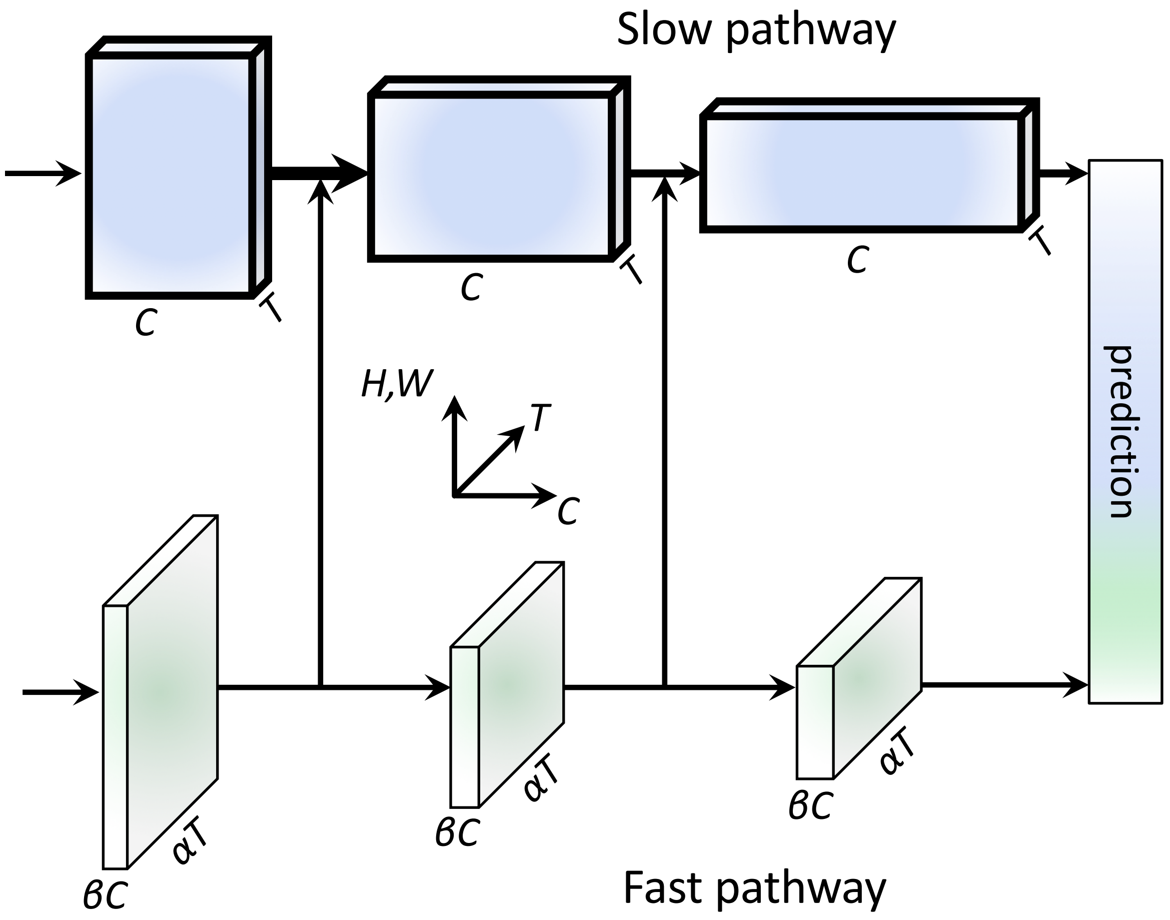
(d) **Channel capacity ratio**: Varying values of β, the channel capacity ratio of the Fast pathway. Backbone: R-50.



SlowFast ablations: Weaken the Fast appearance information

Fast pathway	spatial	top-1	top-5	GFLOPs
RGB	-	**75.6**	**92.1**	27.6
RGB, β=1/4	*half*	74.7	91.8	26.3
gray-scale	-	**75.5**	**91.9**	**26.1**
time diff	-	74.5	91.6	26.2
optical flow	-	73.8	91.3	26.9

(e) **Weaker spatial input to Fast pathway**: Various ways of weakening spatial inputs to the Fast pathway in SlowFast models. β=1/8 unless specified otherwise. Backbone: R-50.

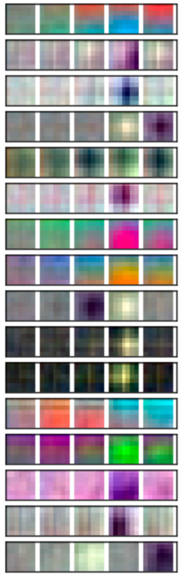


SlowFast ablatios: Weaker input & reduced channels: conv1 filters

Fast pathway	spatial	top-1	top-5	GFLOPs
RGB	-	**75.6**	**92.1**	27.6
gray-scale	-	**75.5**	**91.9**	**26.1**
time diff	-	74.5	91.6	26.2

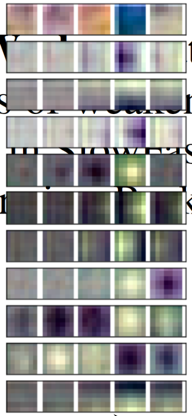
(e) **W... tial input to Fast pathway**: Various ways ... ning spatial inputs to the Fast pathway ... st models. β=1/8 unless specified other... kbone: R-50.

$\beta = 1/4$



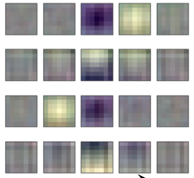
t

$\beta = 1/6$



t

$\beta = 1/16$



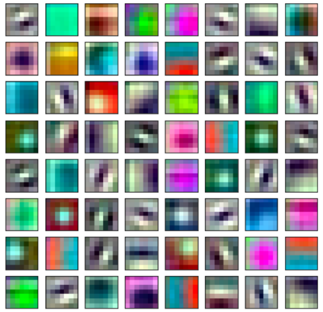
t

$\beta = 1/32$

t

Conv1 filters

Slow



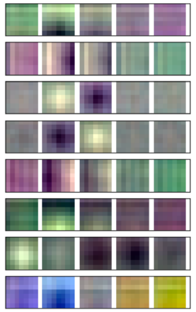
rgb

grayscale

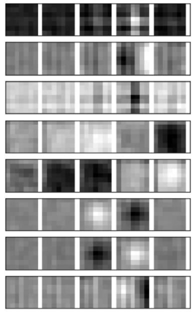
time diff

$\beta = 1/8$

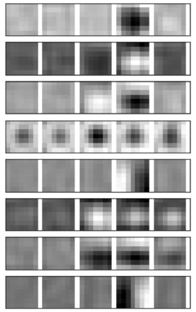
Fast



t



t



t

rgb

grayscale

dt

SlowFast: State-of-the-art comparison on Kinetics

model	flow	pretrain	top-1	top-5	inference GFLOPs×crops
I3D [1]		ImageNet	72.1	90.3	108 × N/A
Two-Stream I3D [1]	✓	ImageNet	75.7	92.0	216 × N/A
S3D-G [6]	✓	ImageNet	74.7	93.4	142.8 × N/A
Nonlocal R-50 [5]		ImageNet	76.5	92.6	282 × 30
Nonlocal R-101 [5]		ImageNet	77.7	93.3	359 × 30
R(2+1)D Flow [3]	✓	-	67.5	87.2	152 × 115
STC [2]		-	68.7	88.5	N/A × N/A
ARTNet [4]		-	69.2	88.3	23.5 × 250
S3D [6]		-	69.4	89.1	66.4 × N/A
ECO [7]		-	70.0	89.4	N/A × N/A
I3D [1]	✓	-	71.6	90.0	216 × N/A
R(2+1)D [3]		-	72.0	90.0	152 × 115
R(2+1)D [3]	✓	-	73.9	90.9	304 × 115
SlowFast, R50 (4×16)		-	75.6	92.1	36.1 × 30
SlowFast, R50		-	77.0	92.6	65.7 × 30
SlowFast, R50 + NL		-	77.7	93.1	80.8 × 30
SlowFast, R101		-	77.9	93.2	106 × 30
SlowFast, R101 + NL		-	**79.0**	**93.6**	115 × 30

+ 5.1% top-1

at 10% of FLOPs

Table 1. **Comparison with the state-of-the-art on Kinetics-400**. In the column of computational cost, we report the cost of a single spacetime crop and the numbers of such crops used. "N/A" indicates the numbers are not available for us. The SlowFast models are the $T \times \tau = 8 \times 8$ versions, unless specified.

[1] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proc. CVPR*, 2017.

[2] A. Diba, M. Fayyaz, V. Sharma, M. M. Arzani, R. Yousefzadeh, J. Gall, and L. Van Gool. Spatio-temporal channel correlation networks for action classification. In *Proc. ECCV*, 2018.

[3] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proc. CVPR*, 2018.

[4] L. Wang, W. Li, W. Li, and L. Van Gool. Appearance-and-relation networks for video classification. In *Proc. CVPR*, 2018.

[5] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *Proc. CVPR*, 2018.

[6] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy. Rethinking spatiotemporal feature learning for video understanding. *arXiv preprint arXiv:1712.04851*, 2017.

[7] M. Zolfaghari, K. Singh, and T. Brox. ECO: efficient convolutional network for online video understanding. In *Proc. ECCV*, 2018.

SlowFast: State-of-the-art comparison Charades[1]

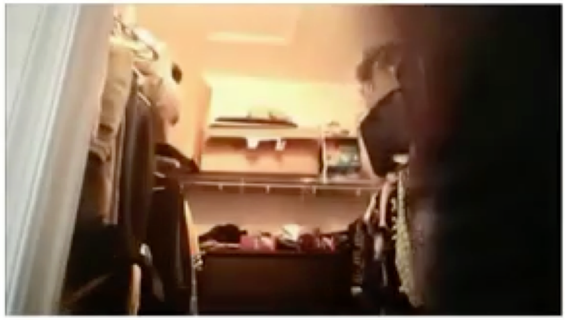
- Charades has 9.8k training videos and 1.8k validation videos in 157 classes
- Multi-label classification setting of longer activities spanning 30 seconds on average

model	pretrain	mAP	inference GFLOPs×views
CoViAR, R-50 [55]	ImageNet	21.9	N/A
Asyn-TF, VGG16 [39]	ImageNet	22.4	N/A
MultiScale TRN [58]	ImageNet	25.2	N/A
Nonlocal, R101 [52]	ImageNet+Kinetics400	37.5	544 × 30
STRG, R101+NL [53]	ImageNet+Kinetics400	39.7	630 × 30
our baseline (Slow-only)	Kinetics-400	39.0	187 × 30
SlowFast	Kinetics-400	41.8	213 × 30
SlowFast, +NL	Kinetics-400	42.5	234 × 30
SlowFast, +NL	Kinetics-600	**45.2**	234 × 30

Table 4. **Comparison with the state-of-the-art on Charades.** All our variants are based on $T \times \tau = 16 \times 8$, R101.

Annotated Actions: (gray if not active)
Turning on a light
Walking through a doorway
Taking a box from somewhere
Holding a box
Opening a box
Taking a pillow from somewhere
Taking something from a box
Closing a box
Holding a pillow
Snuggling with a pillow
Putting something on a shelf
Putting a box somewhere

Video 21 of 50: (3x Speed)



Annotated Objects:
Box, Closet, Doorway, Light, Pillow, Shelf

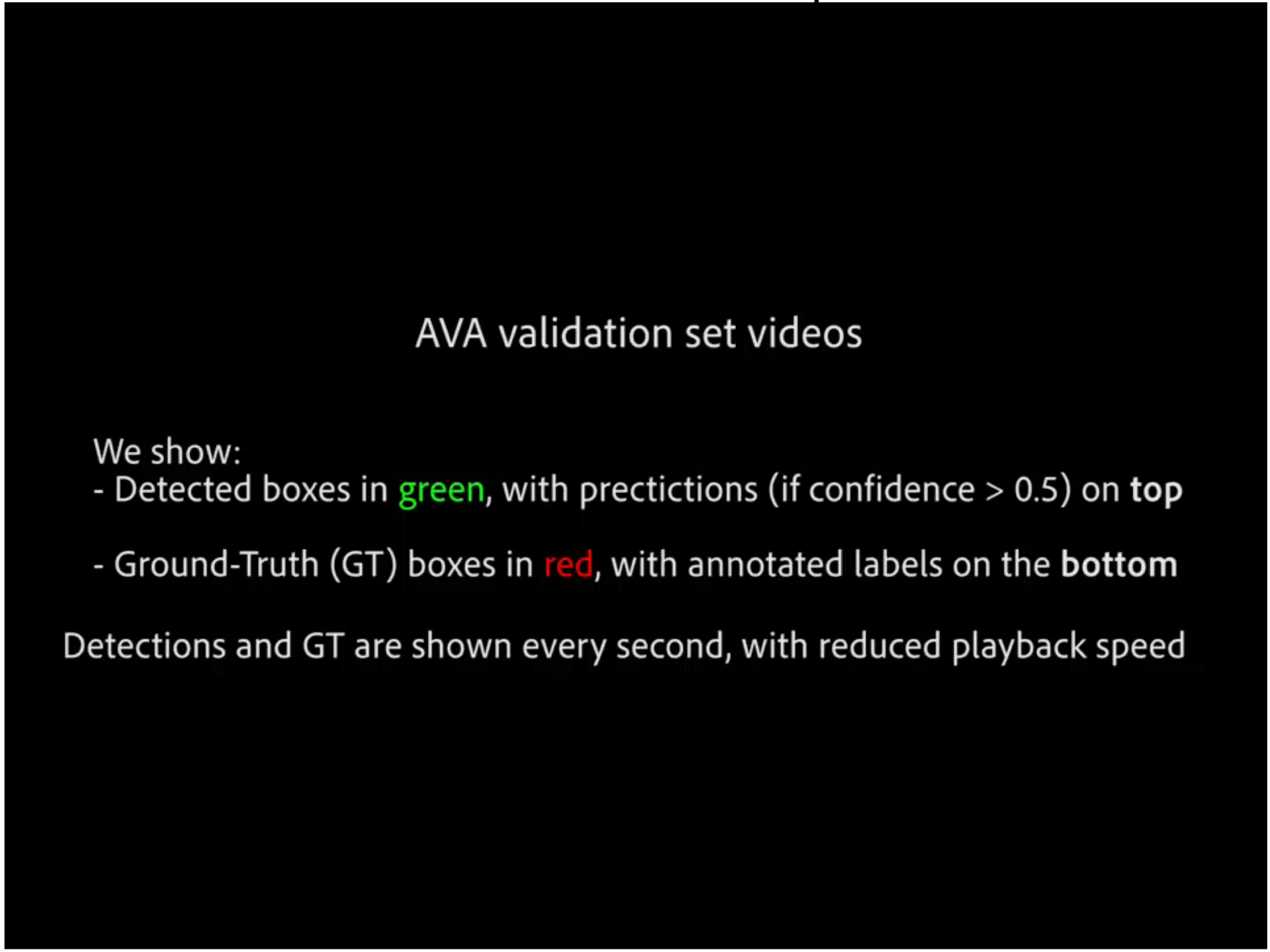
Script:
A person turns on the light in a closet, opens a large container, then grasps a pillow from it.

[1] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In ECCV, 2016. , CVPR 2016

Experiments: AVA[1] Action Detection

- Fine-scale localization of 80 different physical actions
- Data from 437 different movies and spatiotemporal labels are provided in a 1Hz interval
- 211k training and 57k validation video segments
- We follow the standard protocol of evaluating on 60 most freqent classes
- Every person is annotated with a bounding box and (possibly multiple) actions

SlowFast detector output



[1]Gu et al. AVA: A Video Dataset of Spatio-temporally Localized Atomic Visual Actions, CVPR 2018

SlowFast: AVA action detection

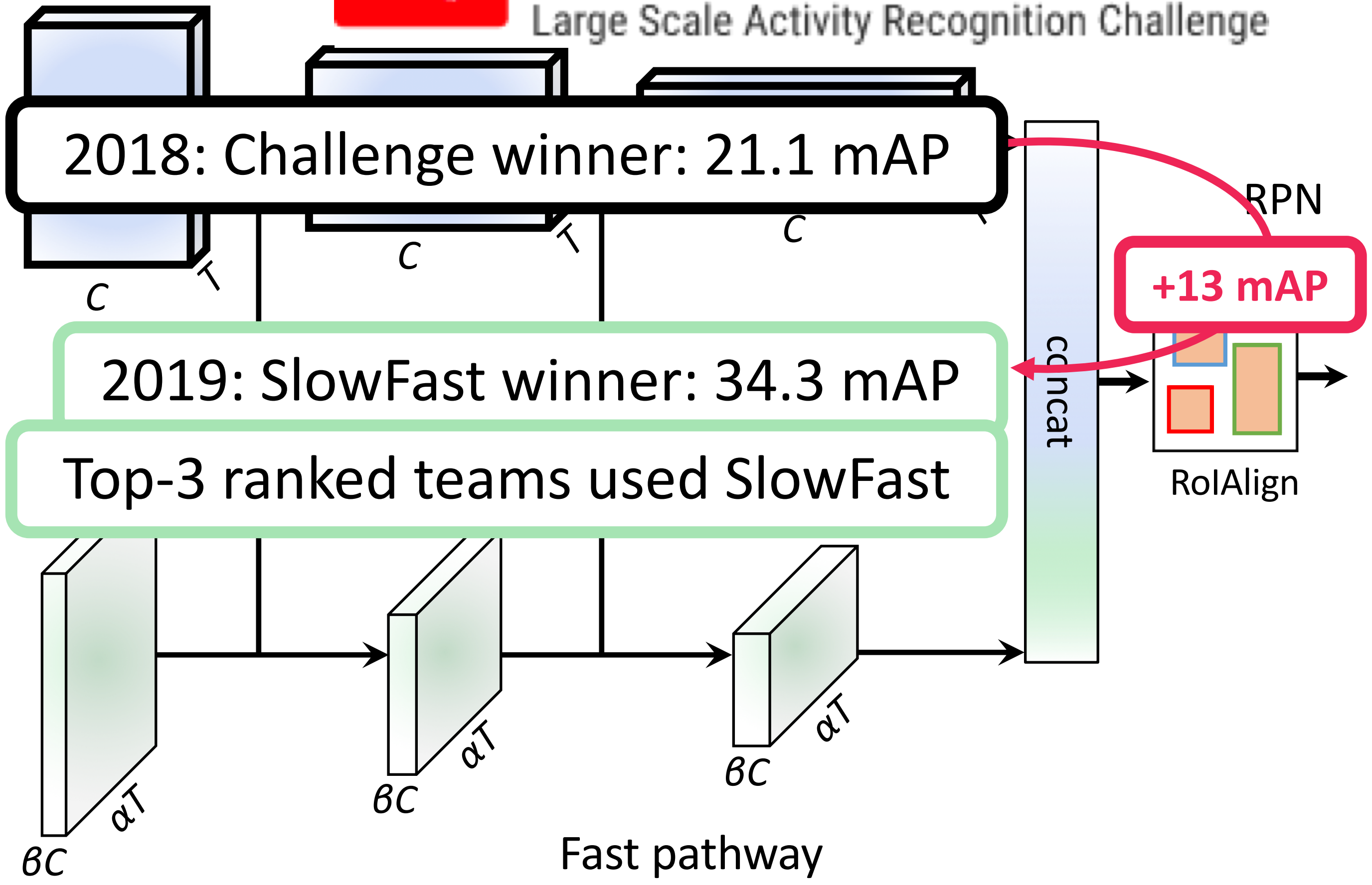
ACTIVITYNET
Large Scale Activity Recognition Challenge

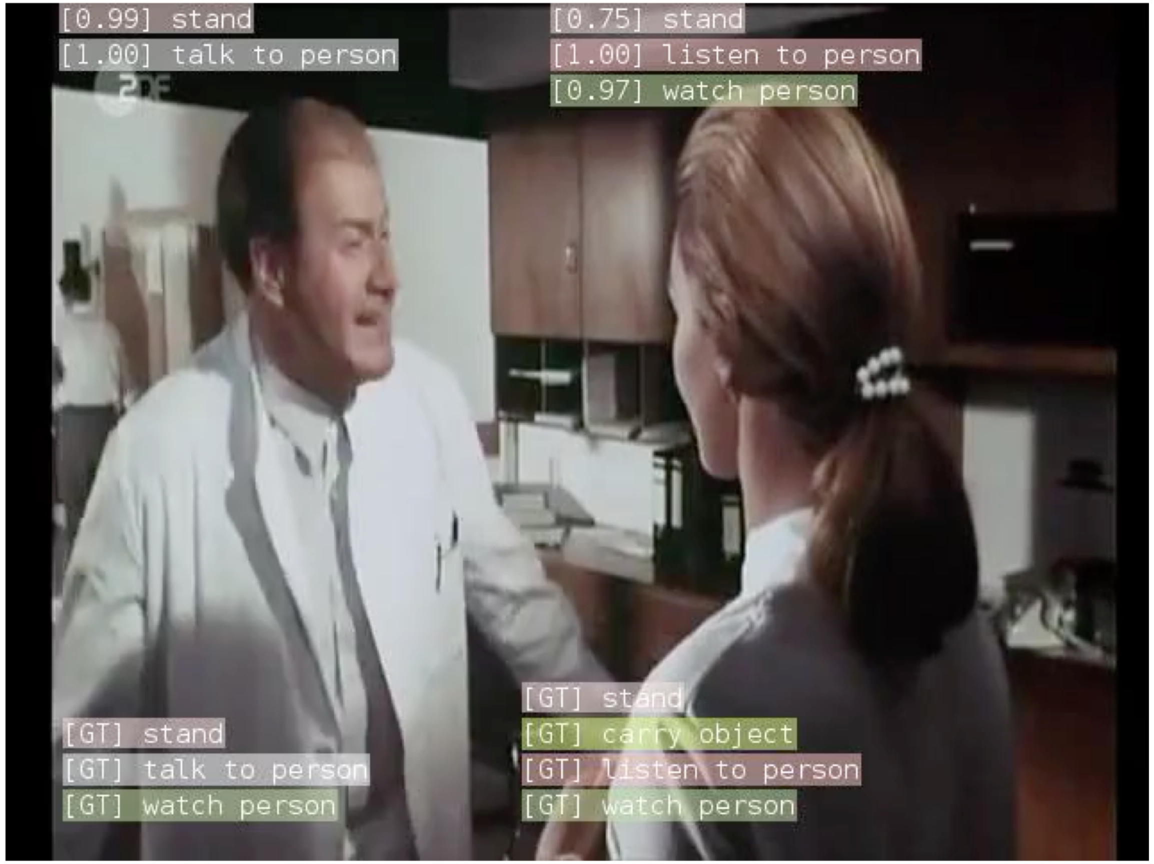
2018: Challenge winner: 21.1 mAP

+13 mAP

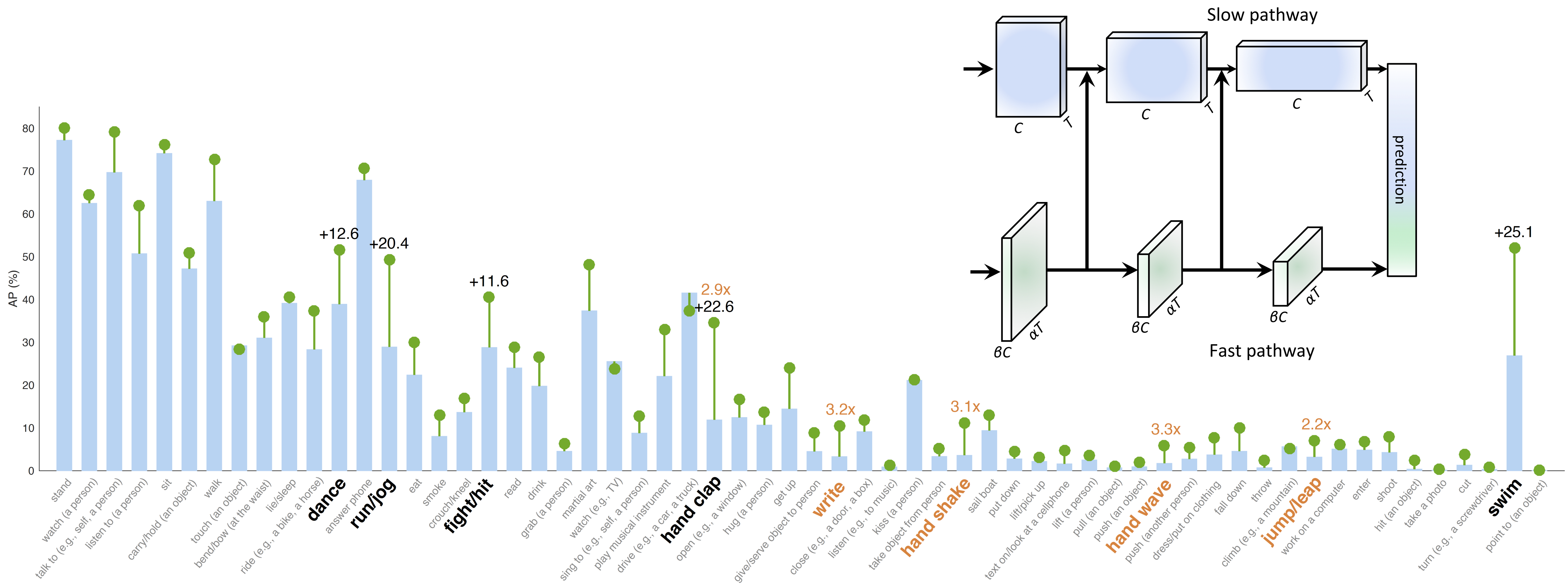
2019: SlowFast winner: 34.3 mAP

Top-3 ranked teams used SlowFast





SlowFast ablations: AVA class level performance



Experiments: AVA Qualitative results

Experiments: AVA Qualitative results

Pytorch code available:

Conclusion

https://github.com/facebookresearch/SlowFast

- The time axis is a special dimension of video
- *3D ConvNets* treat space and time uniformly
- *Non-local networks* and *Long-term feature banks* aggregate long-term spatiotemporal information
- *SlowFast* & *Two-Stream* networks treat space and time differently and share motivation from neuroscience
- The *SlowFast* architecture design focuses on contrasting the speed along the temporal axis
- Given the mutual benefits of jointly modeling video with different temporal speeds, we hope that this concept can foster further research in video analysis