Videos

Saurabh Gupta

CS 543 / ECE 549 Computer Vision

Spring 2020

Outline

- Optical Flow
- Tracking
- Correspondence
- Recognition in Videos

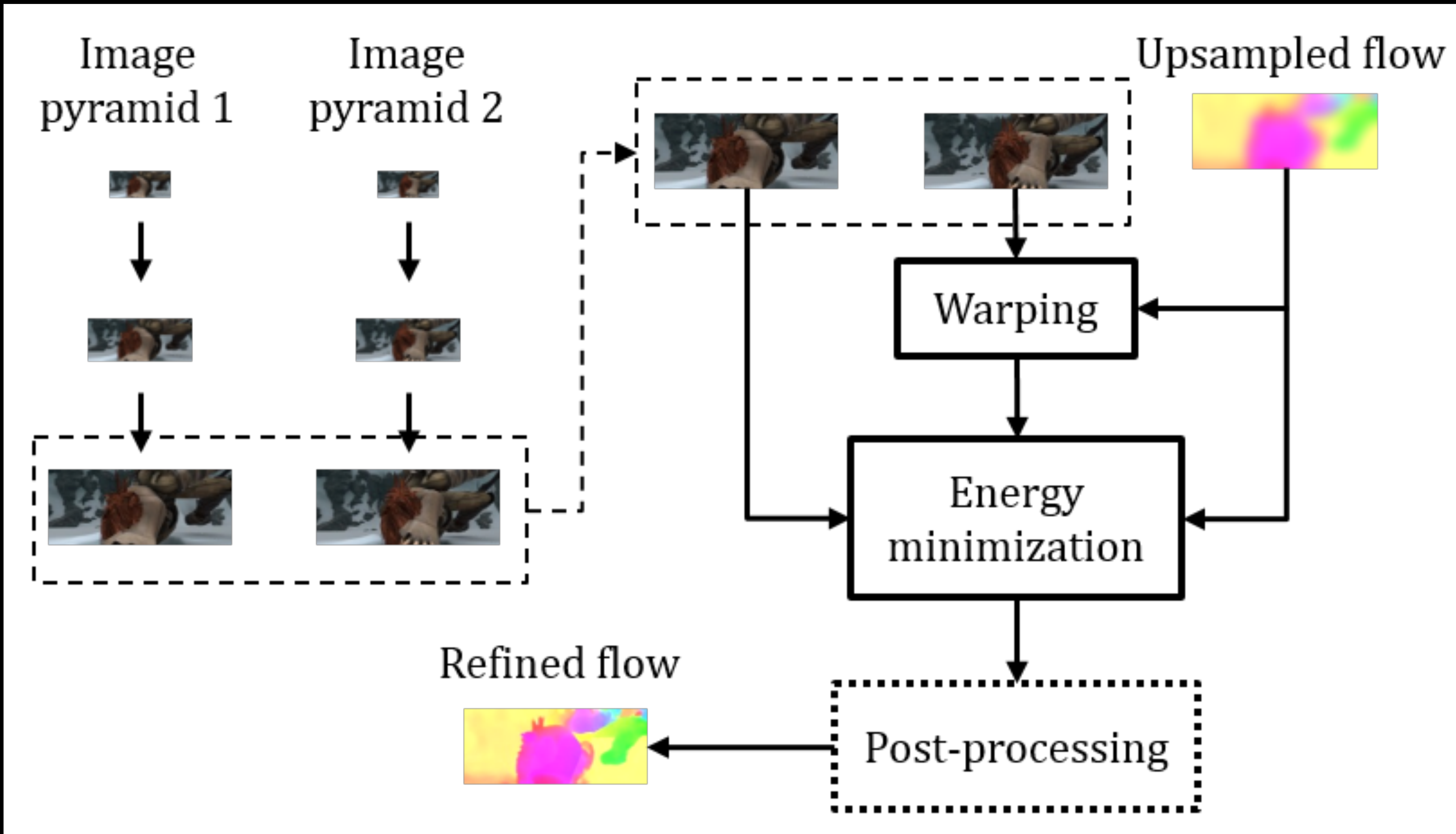
Optical Flow

- Data / Supervision
- Architecture

Datasets

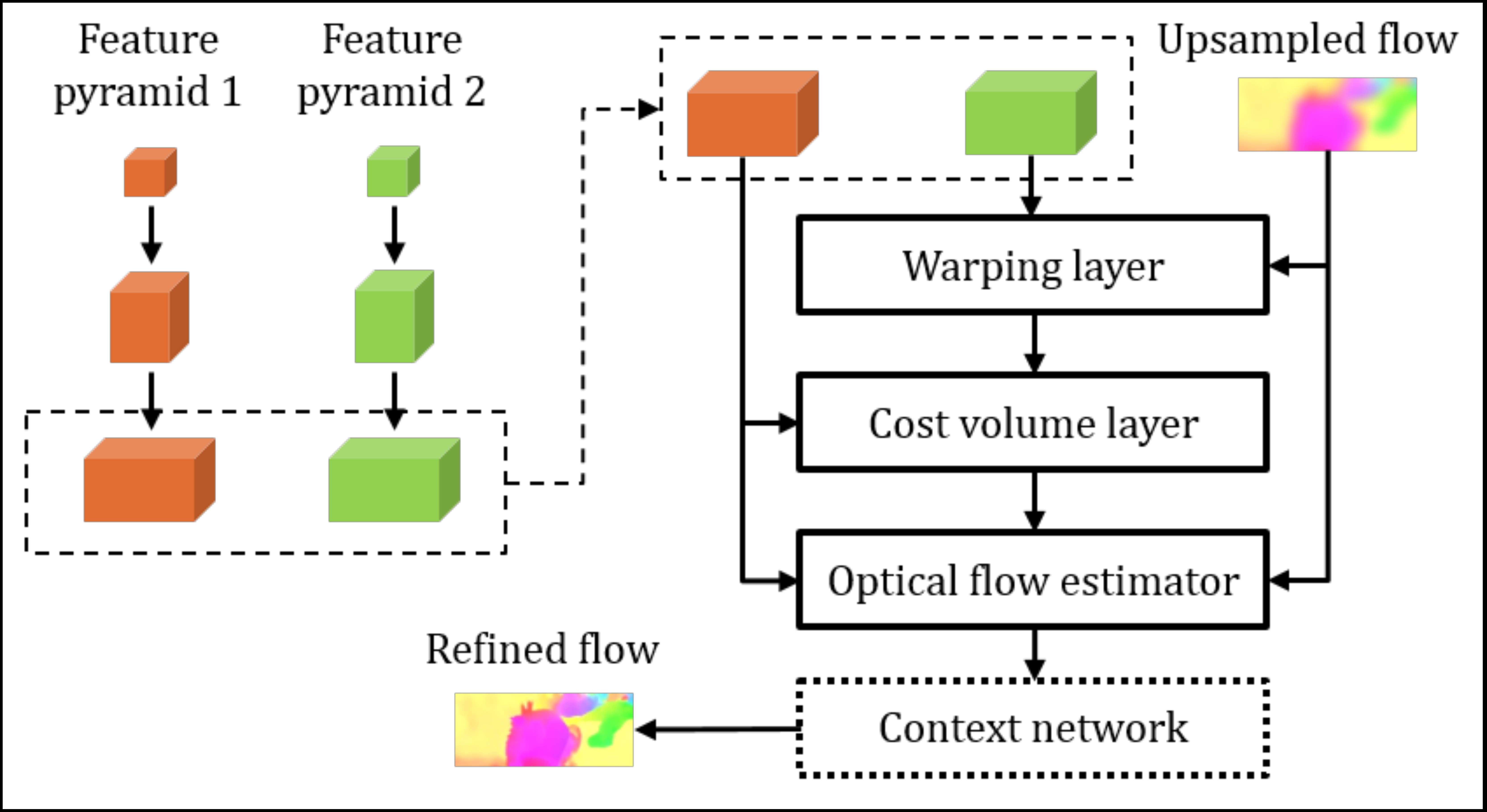
- Traditional datasets: Yosemite, Middlebury
- KITTI: http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=flow
- Sintel: http://sintel.is.tue.mpg.de/
- Synthetic Datasets
  - Flying Chairs et al: https://lmb.informatik.uni-freiburg.de/resources/datasets/FlyingChairs.en.html
- Supervision: from Simulation
- Metrics: End-point Error

“Classical Optical Flow Pipeline”



PWC Net

$$\mathbf{cv}^l(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{N}\left(\mathbf{c}_1^l(\mathbf{x}_1)\right)^\top \mathbf{c}_w^l(\mathbf{x}_2),$$



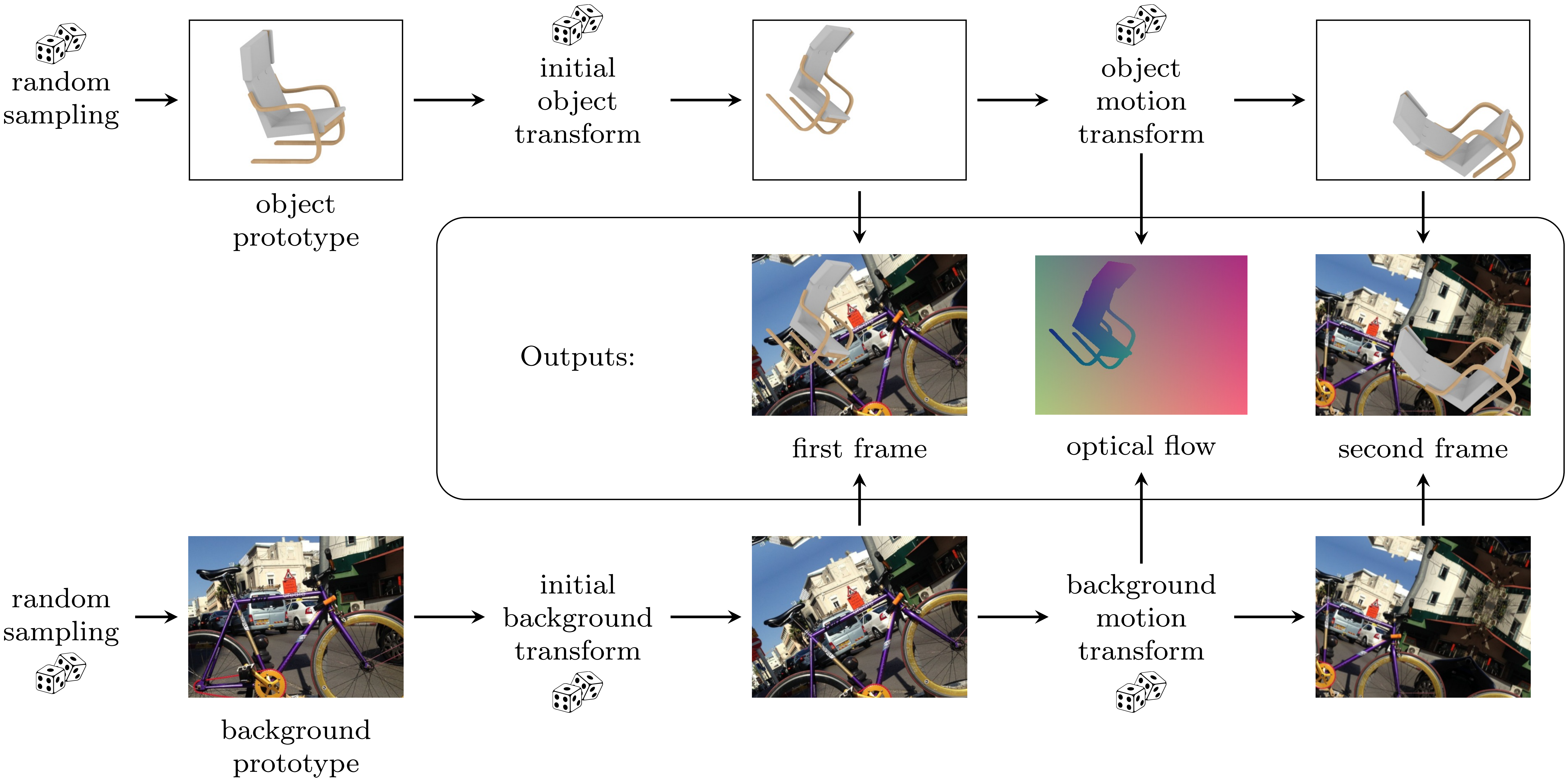
Models Matter, So Does Training: An Empirical Study of CNNs for Optical Flow Estimation. Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. arXiv 2018.

PWC Net

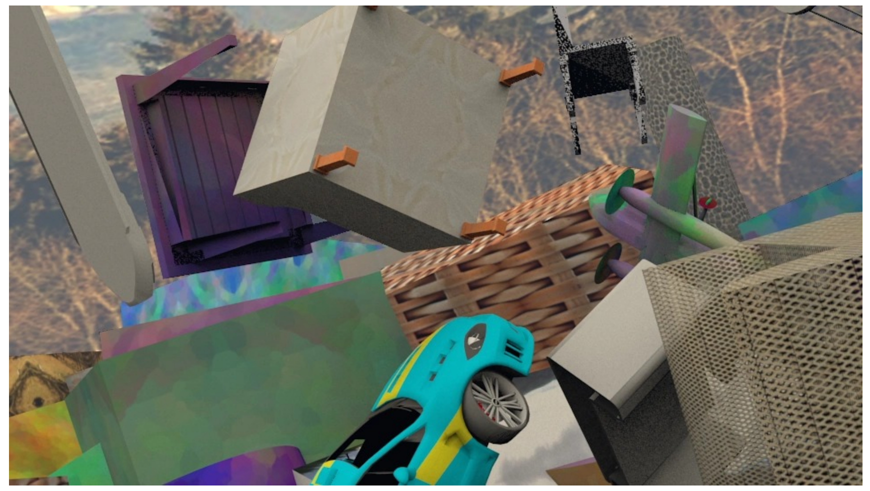
Max. Disp.	Chairs	Sintel Clean	Sintel Final	KITTI 2012 AEPE	KITTI 2012 Fl-all	KITTI 2015 AEPE	KITTI 2015 Fl-all
0	2.13	3.66	5.09	5.25	29.82%	13.85	43.52%
2	2.09	**3.30**	**4.50**	5.26	**25.99**%	13.67	**38.99**%
Full model (4)	2.00	3.33	4.59	5.14	28.67%	13.20	41.79%
6	**1.97**	3.31	4.60	**4.96**	27.05%	**12.97**	40.94%

(b) **Cost volume.** Removing the cost volume (0) results in moderate performance loss. PWC-Net can handle large motion using a small search range to compute the cost volume.

Flying Chairs Dataset



"FlyingChairs" (synth.)
Dosovitskiy et al (2015)



"FlyingThings3D" (synth.)
Mayer et al (2016)

"Monkaa" (synth.)
Mayer et al (2016)

"Virtual KITTI" (synth.)
Gaidon et al (2016)

	Test data		
Training data	Sintel	KITTI2015	FlyingChairs
Sintel	6.42	18.13	5.49
FlyingChairs	**5.73**	16.23	**3.32**
FlyingThings3D	6.64	18.31	5.21
Monkaa	8.47	16.17	7.08
Driving	10.95	**11.09**	9.88

Tracking

- Problem Statements
- Tracking by Detection
- General Object Tracking

Problem Statements

- Single Object Tracking (eg: https://nanonets.com/blog/content/images/2019/07/messi_football_track.gif)
- Multi-object Tracking (eg: https://motchallenge.net/vis/MOT20-02/gt/)
- Multi-object Tracking and Segmentation (eg: https://www.youtube.com/watch?v=K38_pZw_P9s)

Tracking by Detection

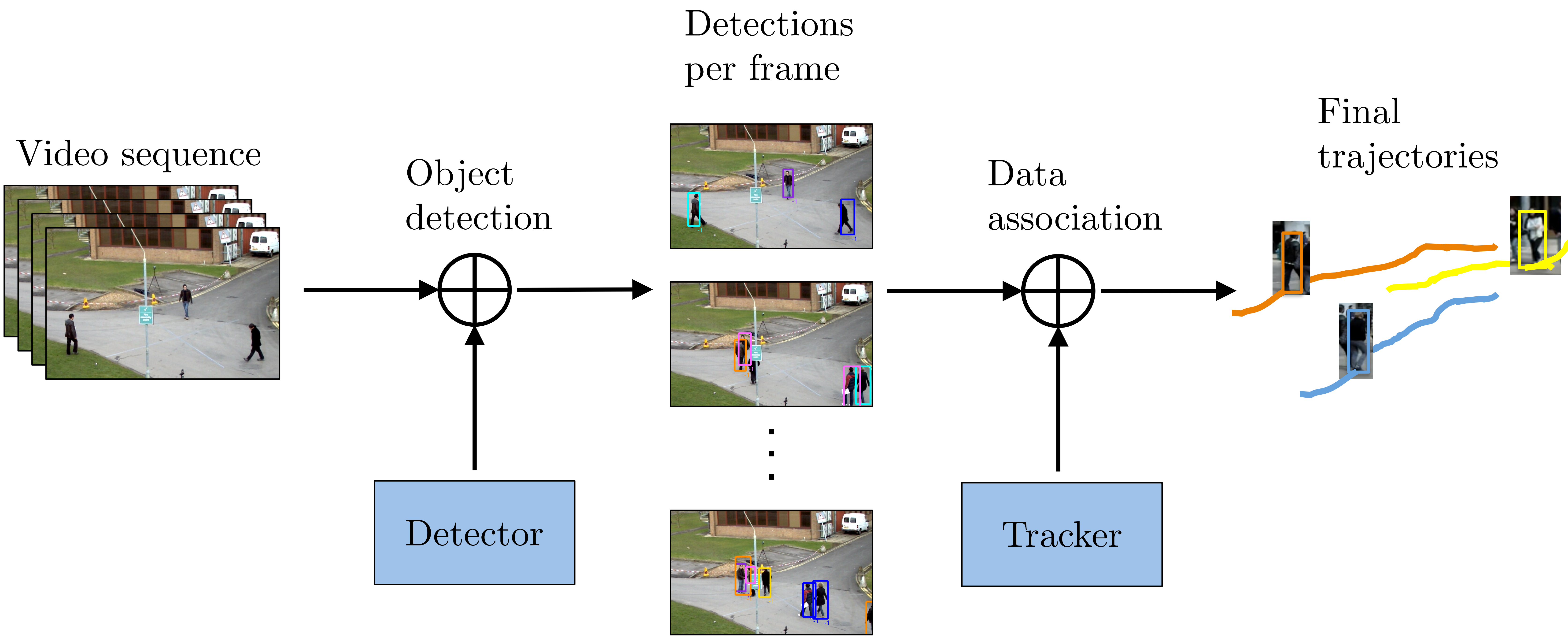
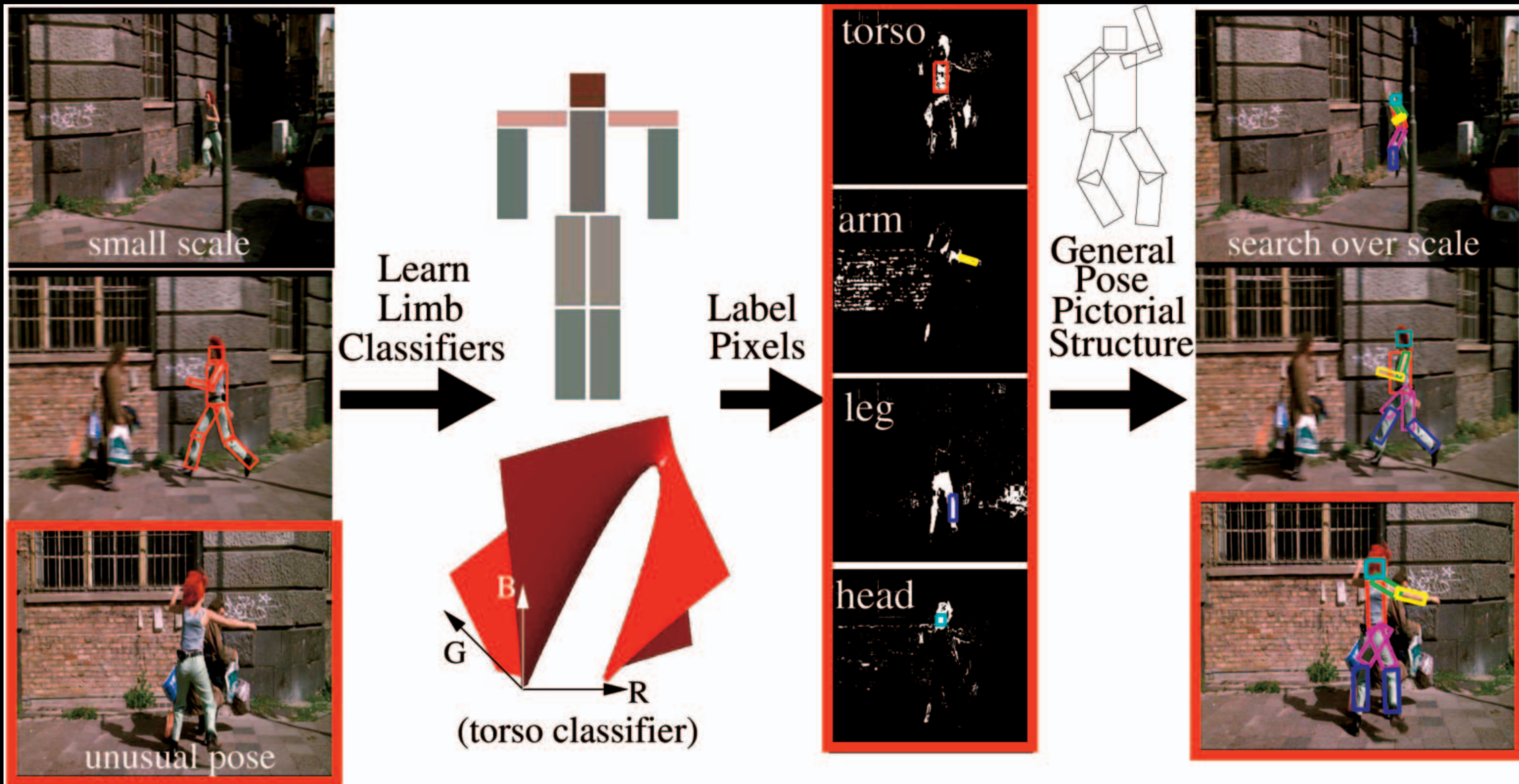


FIGURE 2.2: Tracking-by-detection paradigm. Firstly, an independent detector is applied to all image frames to obtain likely pedestrian detections. Secondly, a tracker is run on the set of detections to perform data association, *i.e.*, link the detections to obtain full trajectories.

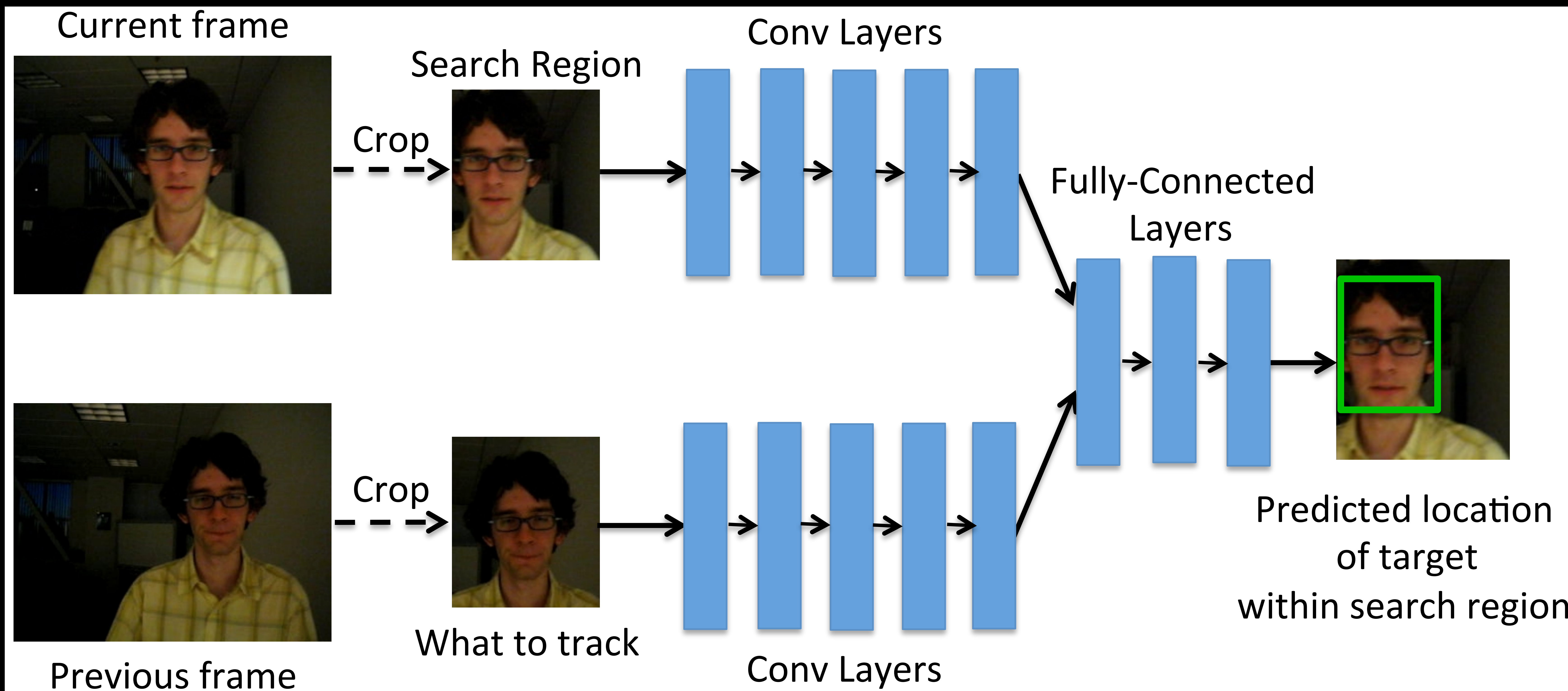
Source: Laura Leal-Taixé

Tracking by Detection



Strike a Pose! Tracking People by Learning Their Appearance. D. Ramanan et al. , PAMI 2007

General Object Tracking



Learning to Track at 100 FPS with Deep Regression Networks. D. Held et al., ECCV16.

Correspondence in Time

Tracking
(Box-level, long-range)

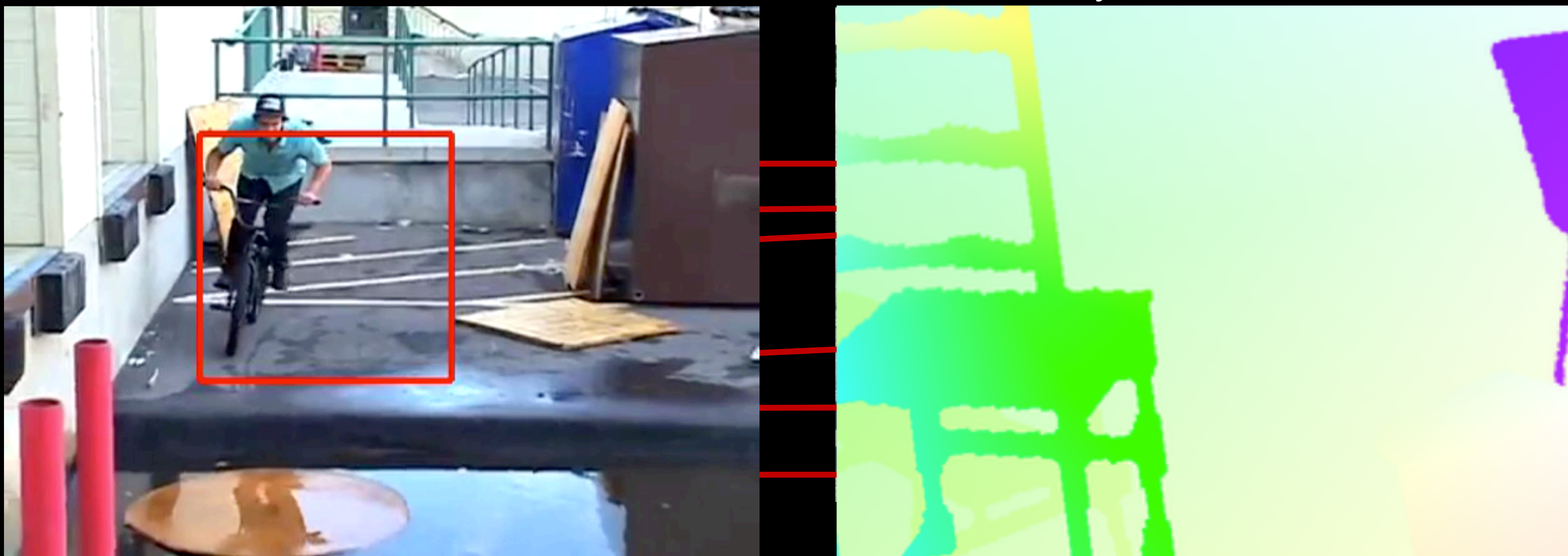
Middle Ground
(Mid-level, long-range)

Optical Flow
(Pixel-level, short-range)

Human Annotations

Self-Supervised / Unsupervised Learning

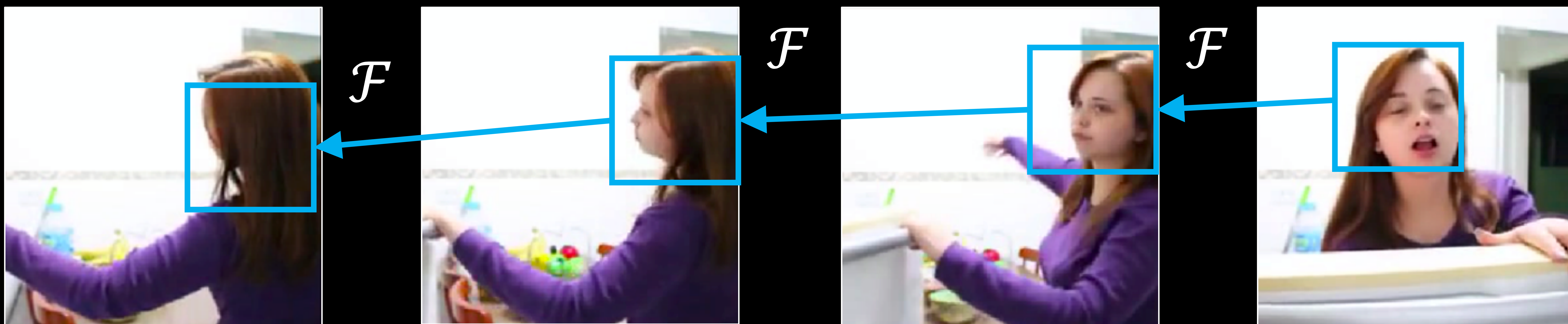
Synthetic Data



Source: Xiaolong Wang

Learning to Track

$\mathcal{F}$: a deep tracker

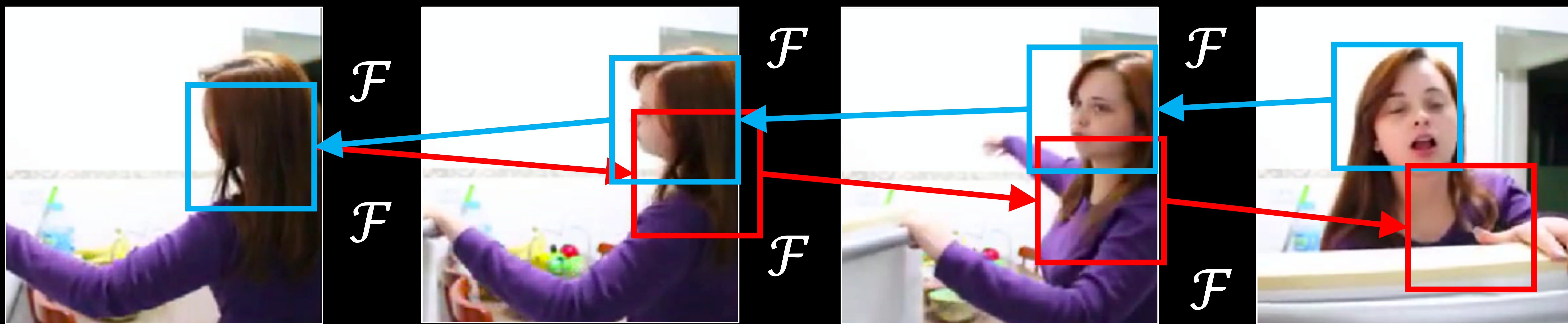


How to obtain supervision?

Source: Xiaolong Wang

Supervision: Cycle-Consistency in Time

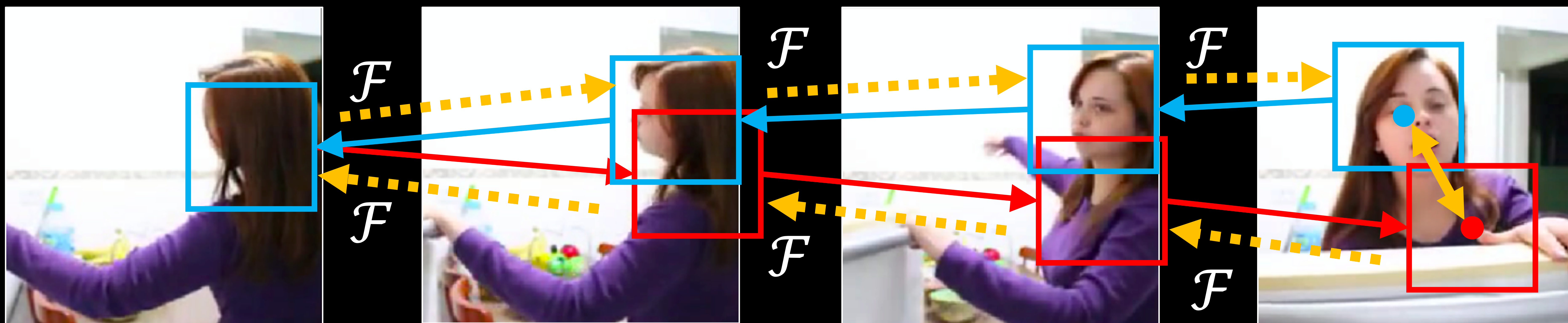
Track backwards



Track forwards, back to the future

Source: Xiaolong Wang

Supervision: Cycle-Consistency in Time



Backpropagation through time, along the cycle

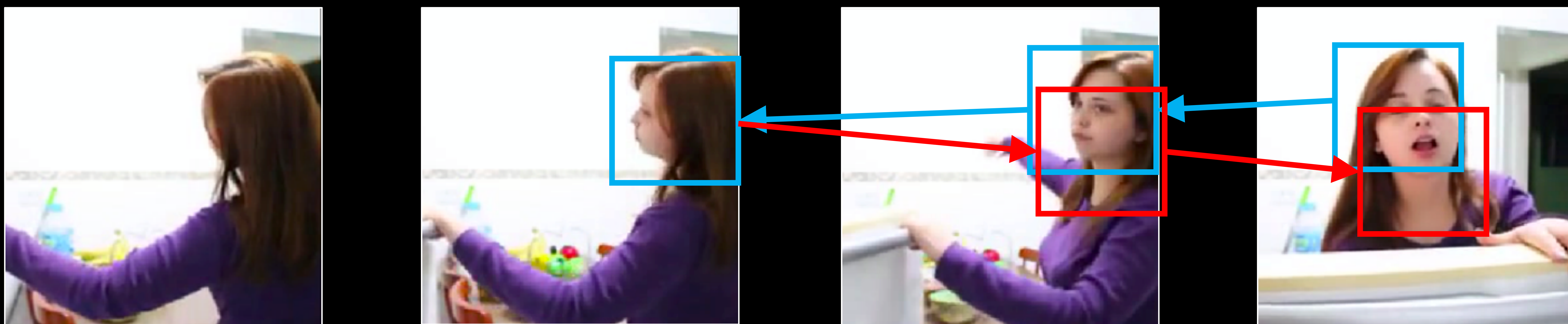
Source: Xiaolong Wang

Multiple Cycles

Sub-cycles: a natural curriculum

Source: Xiaolong Wang

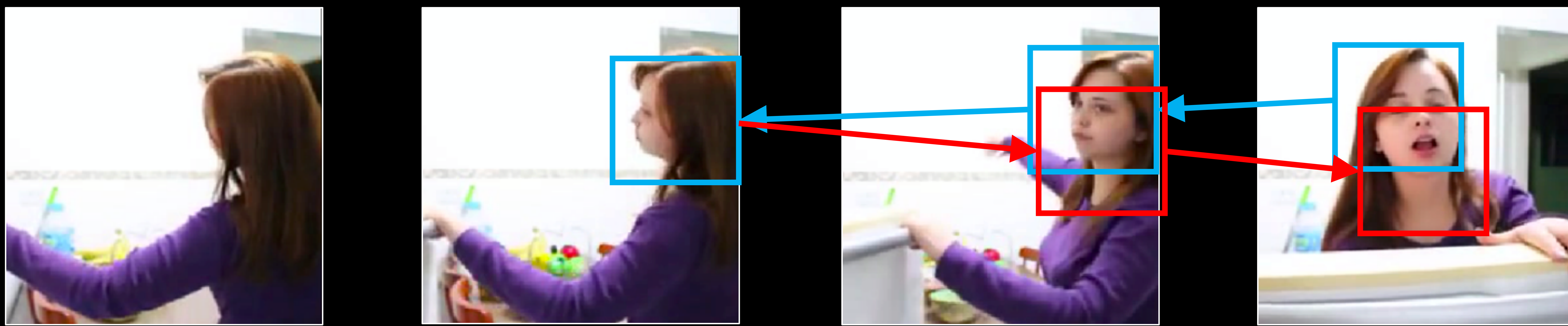
Multiple Cycles



Shorter cycles: a natural curriculum

Source: Xiaolong Wang

Multiple Cycles

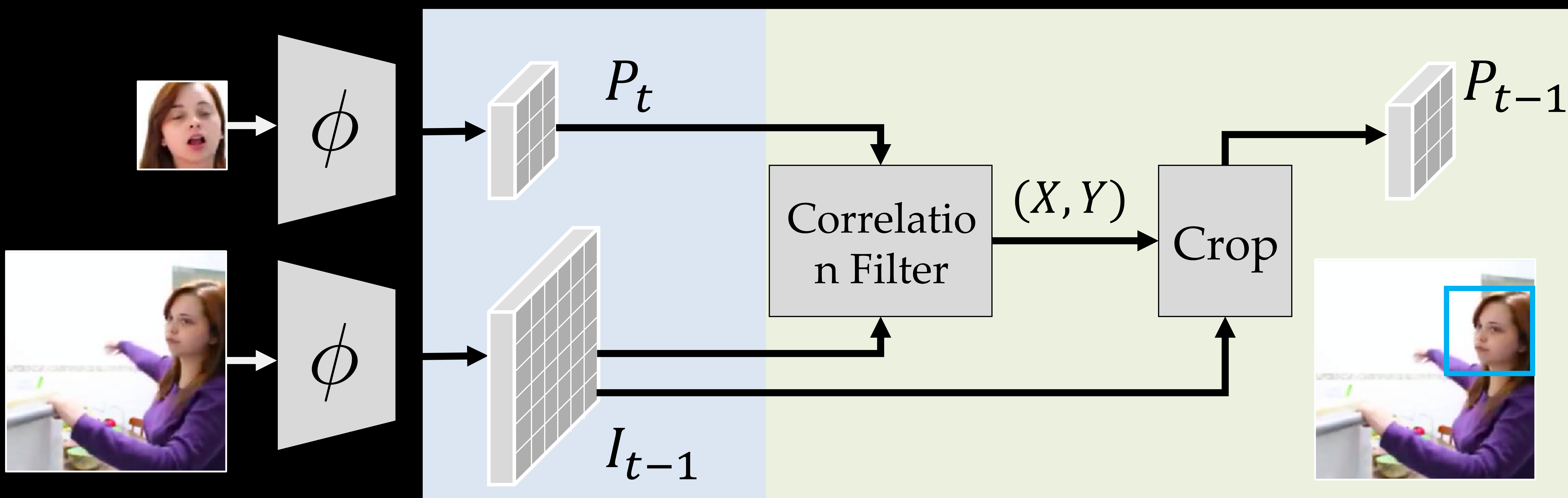


Shorter cycles: a natural curriculum

Source: Xiaolong Wang

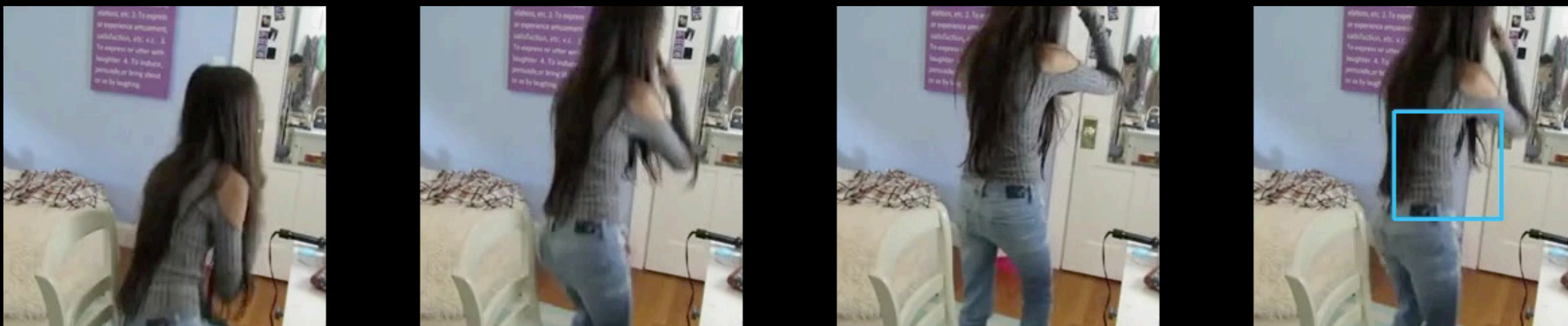
Tracker $\mathcal{F}$

Densely match features in learned feature space ϕ



Source: Xiaolong Wang

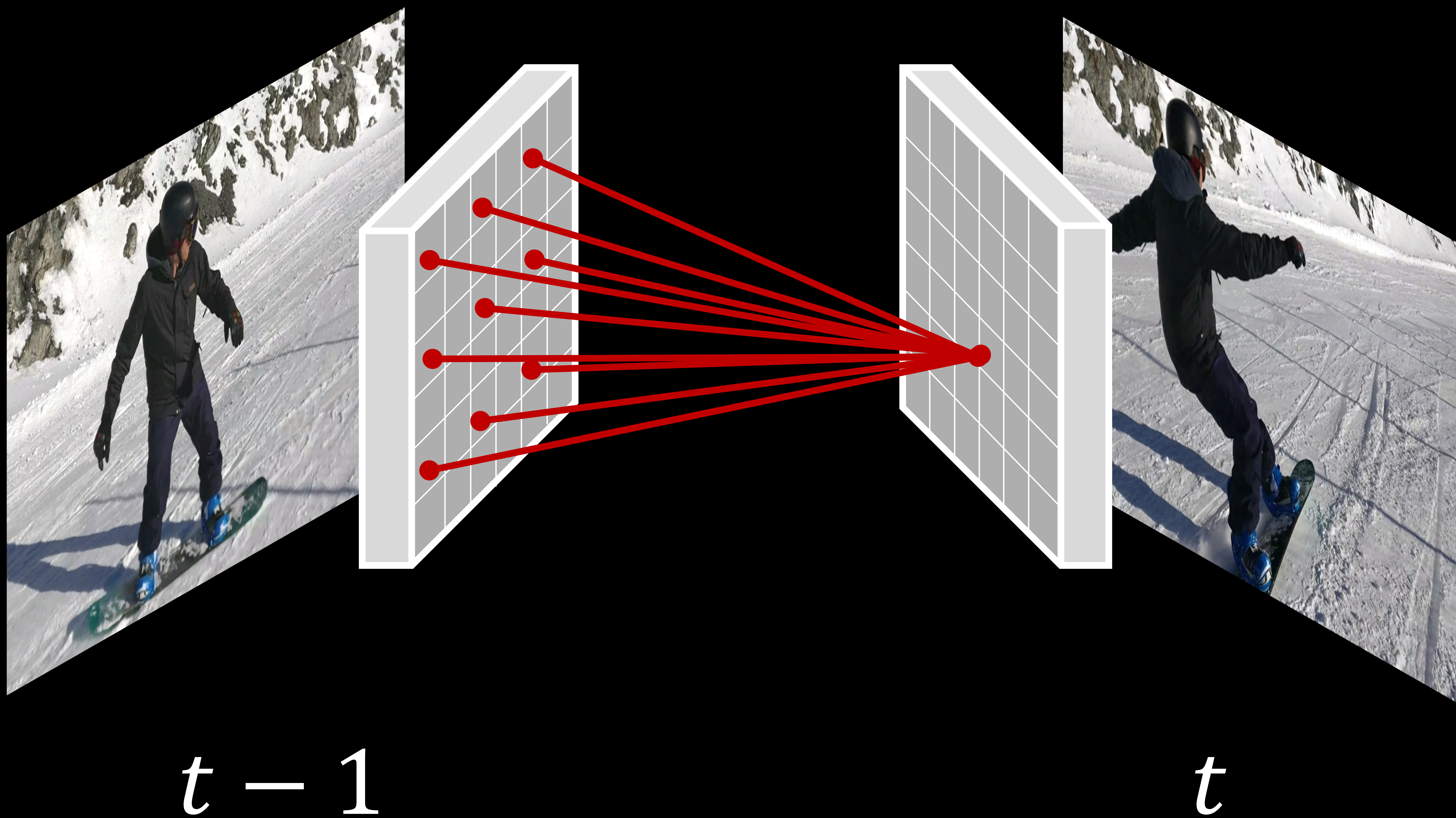
Visualization of Training



Iteration: 1200

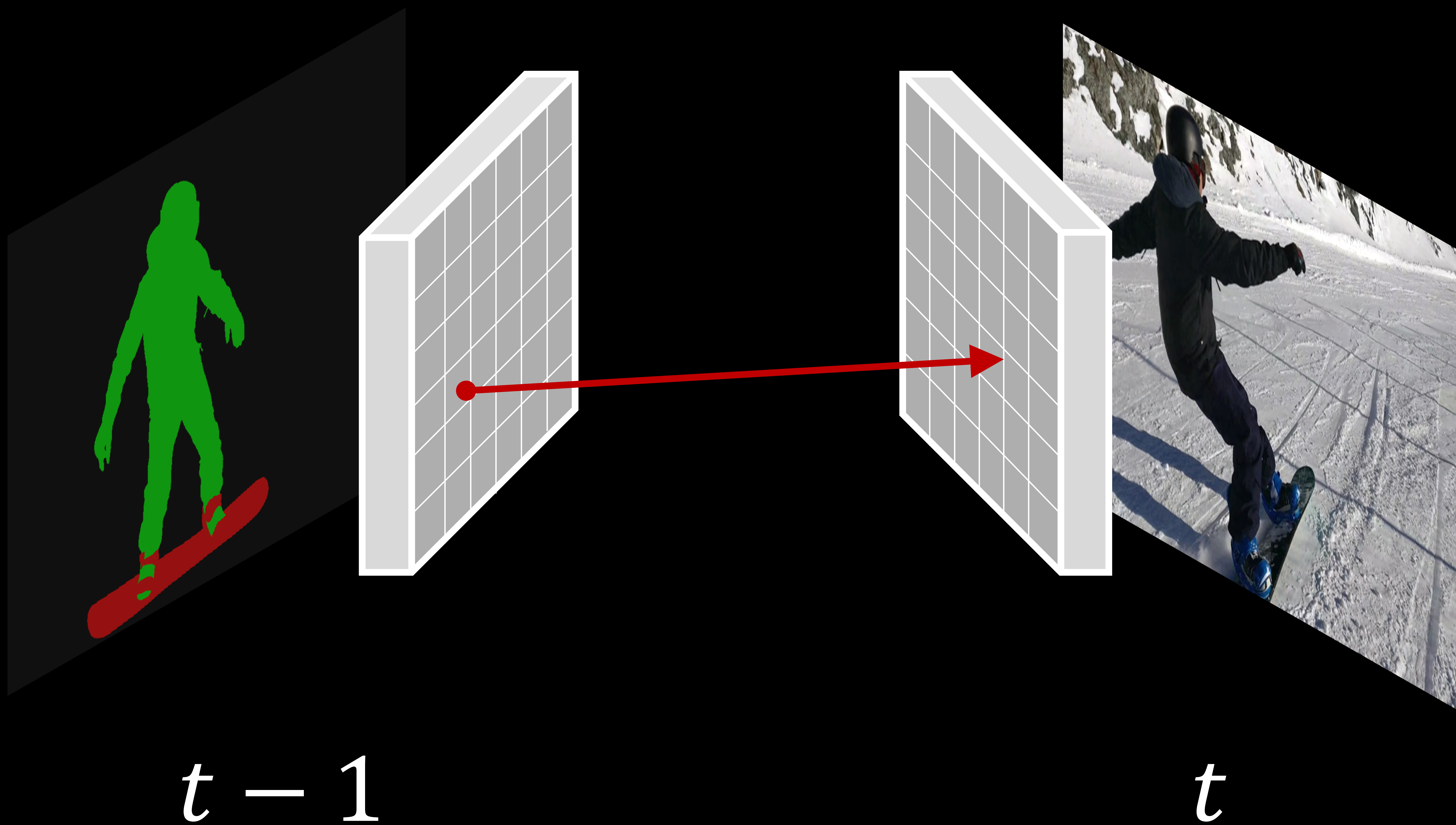
Source: Xiaolong Wang

Test Time: Nearest Neighbors in Feature Space ϕ



Source: Xiaolong Wang

Test Time: Nearest Neighbors in Feature Space ϕ

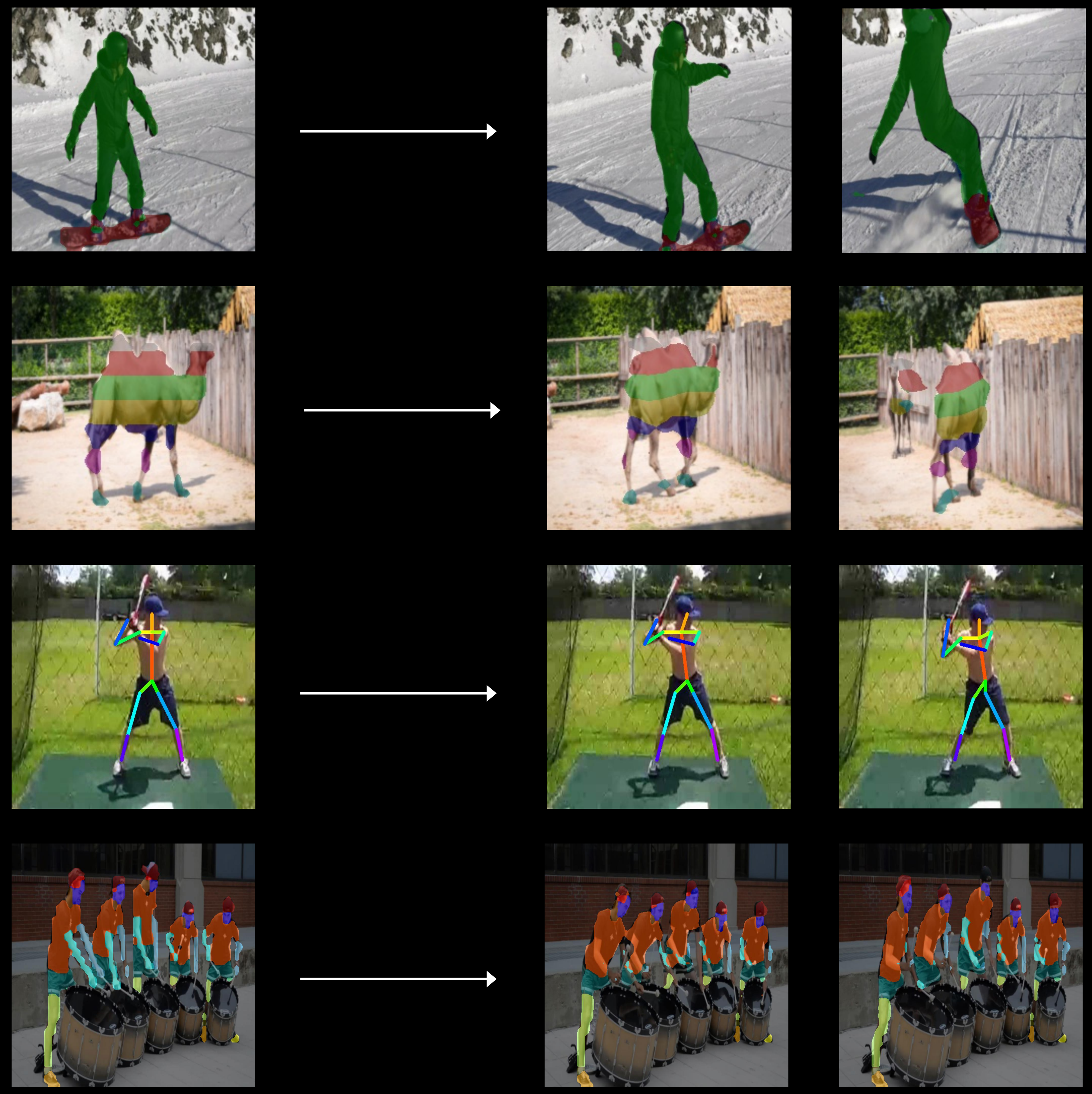


Source: Xiaolong Wang

Evaluation: Label Propagation

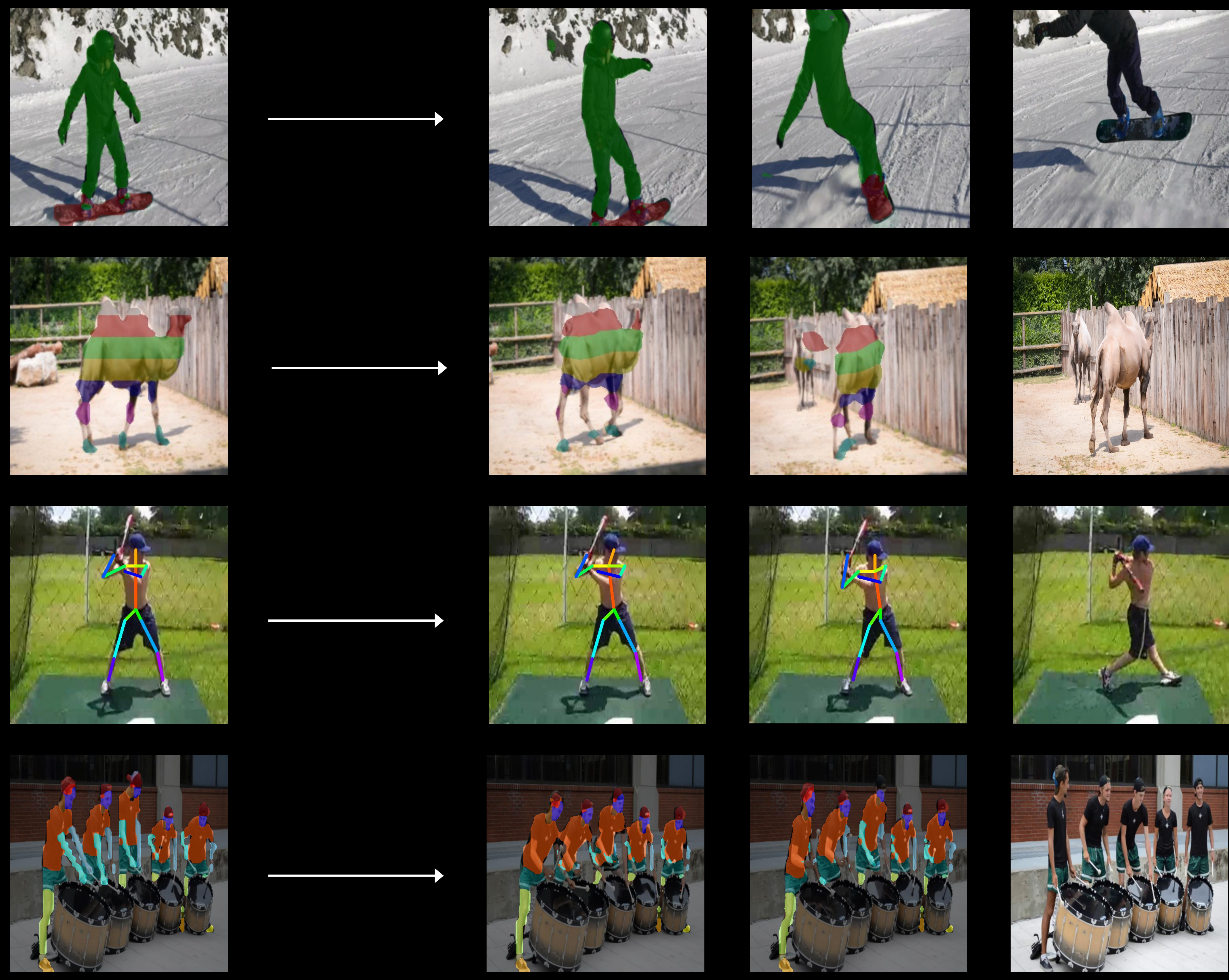
Source: Xiaolong Wang

Evaluation: Label Propagation



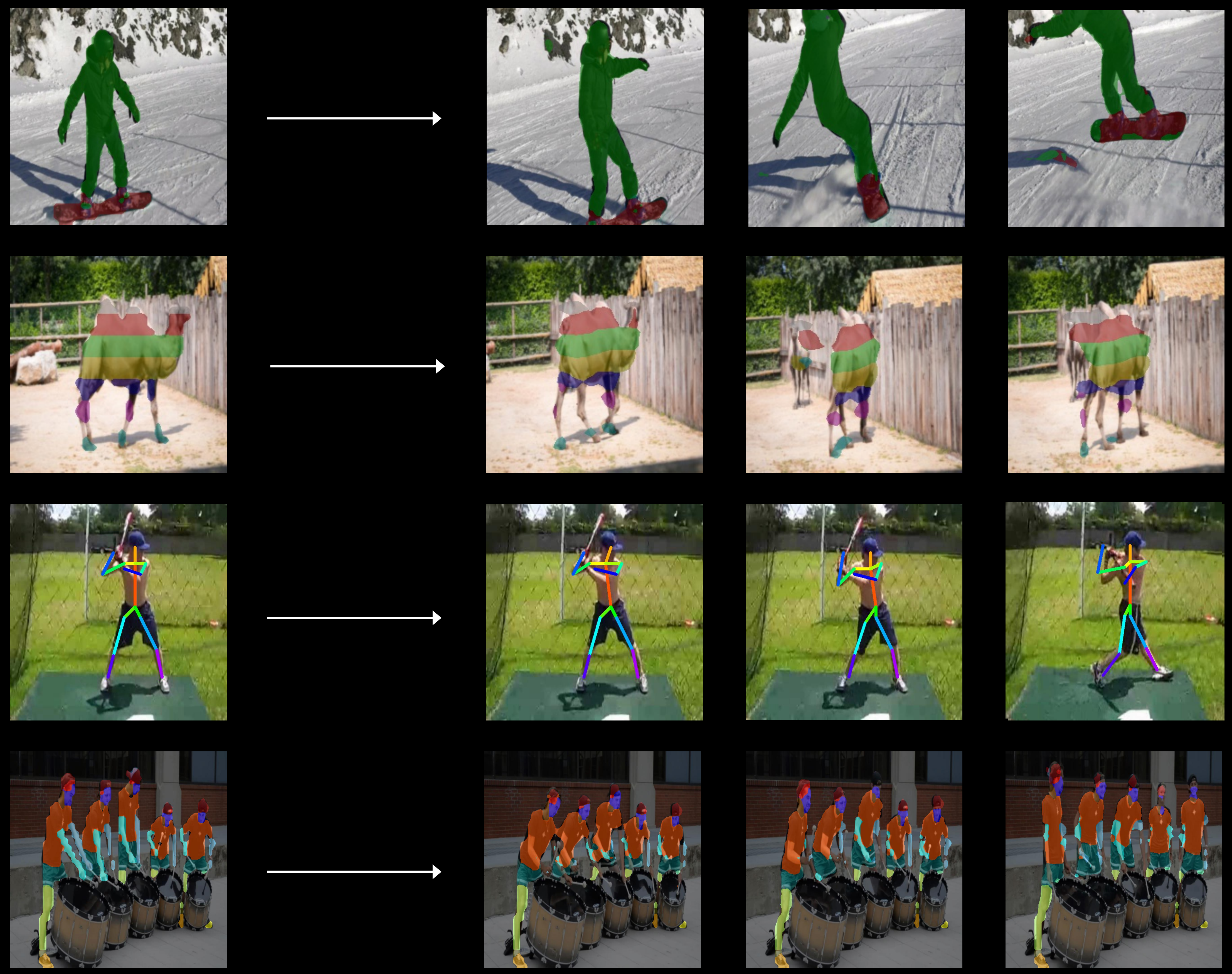
Source: Xiaolong Wang

Evaluation: Label Propagation



Source: Xiaolong Wang

Evaluation: Label Propagation



Source: Xiaolong Wang

Source: Xiaolong Wang

Instance Mask Tracking

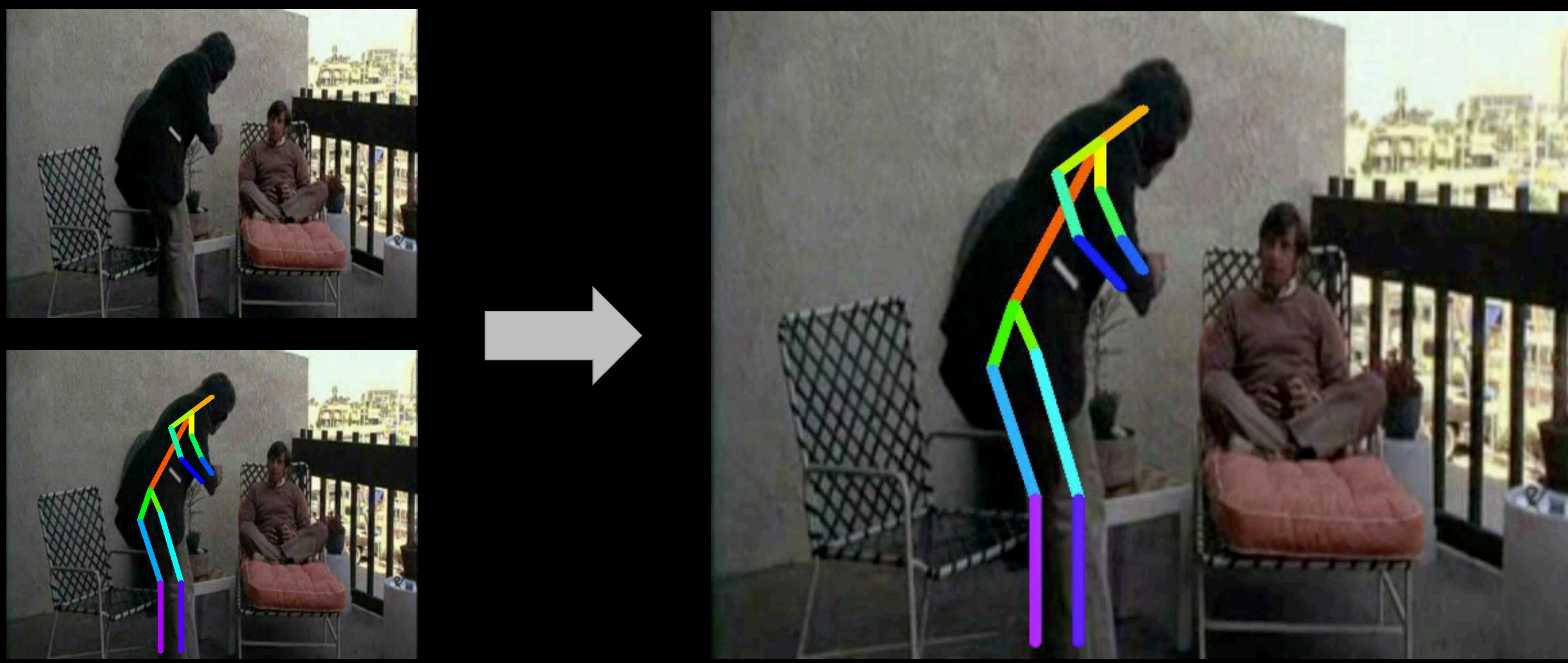
DAVIS Dataset

Source: Xiaolong Wang

DAVIS Dataset: Pont-Tuset et al. *The 2017 DAVIS Challenge on Video Object Segmentation*. 2017.

Pose Keypoint Tracking

JHMDB Dataset



Source: Xiaolong Wang

Comparison

Our Correspondence

Optical Flow

Source: Xiaolong Wang

Texture Tracking

DAVIS Dataset

Source: Xiaolong Wang

DAVIS Dataset: Pont-Tuset et al. *The 2017 DAVIS Challenge on Video Object Segmentation*. 2017.

Semantic Masks Tracking

Video Instance Parsing Dataset

Source: Xiaolong Wang

Zhou et al. *Adaptive Temporal Encoding Network for Video Instance-level Human Parsing*. ACM MM 2018.

Outline

- Optical Flow
- Tracking
- Correspondence
- Recognition in Videos
  - Tasks
  - Datasets
  - Models
- Applications

Recognition in Videos

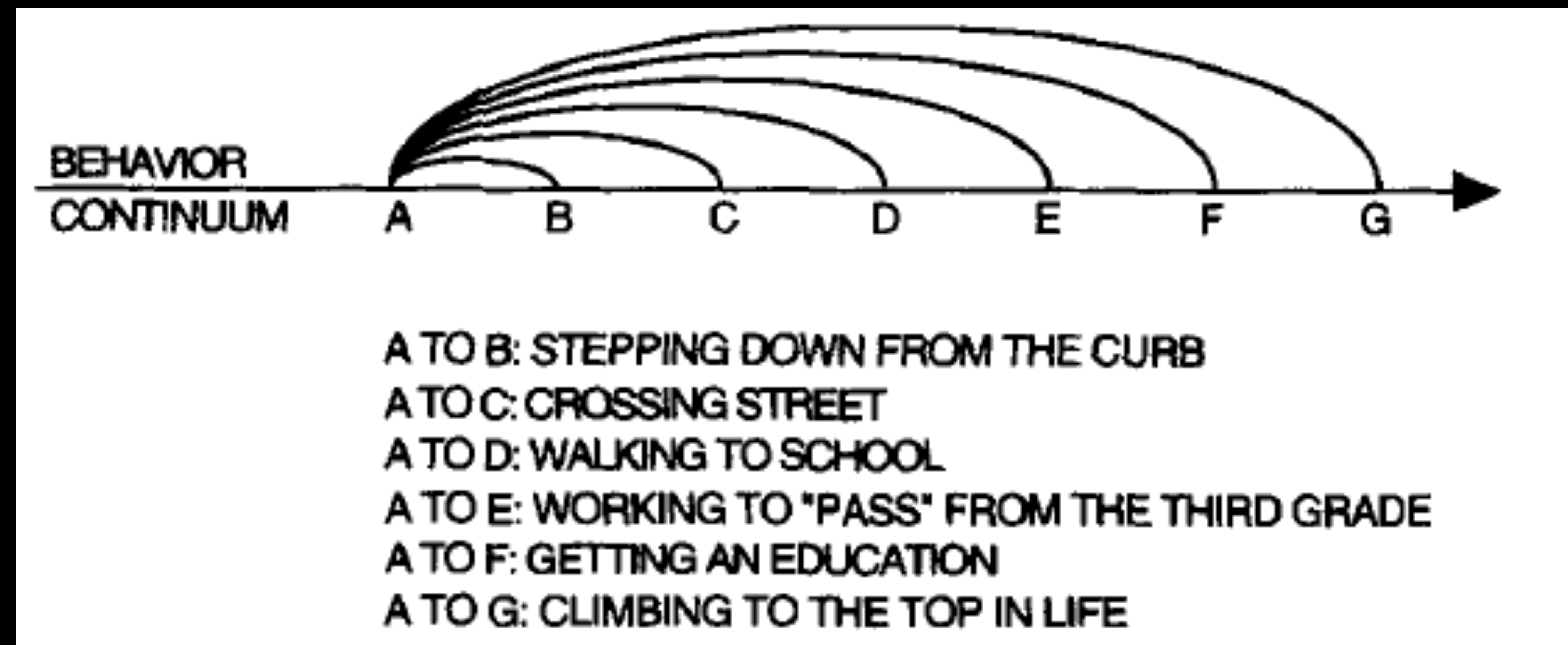
- Tasks / Datasets
- Models

Tasks and Datasets

- Action Classification
  - Kinetics Dataset: https://arxiv.org/pdf/1705.06950.pdf
  - ActivityNet, Sports-8M, …
- Action "Detection"
  - In space, in time. Eg: JHMDB, AV

Tasks and Datasets

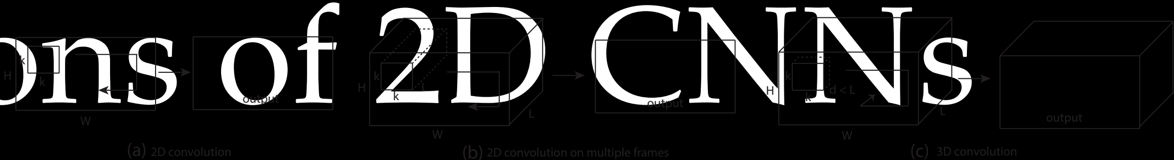
- Time scale
  - Atomic Visual Actions (AVA) Dataset: https://research.google.com/ava/explore.html
- Bias
  - Something Something Dataset: https://20bn.com/datasets/something-something



We don't quite know how do define good meaningful tasks for videos. More on this later.

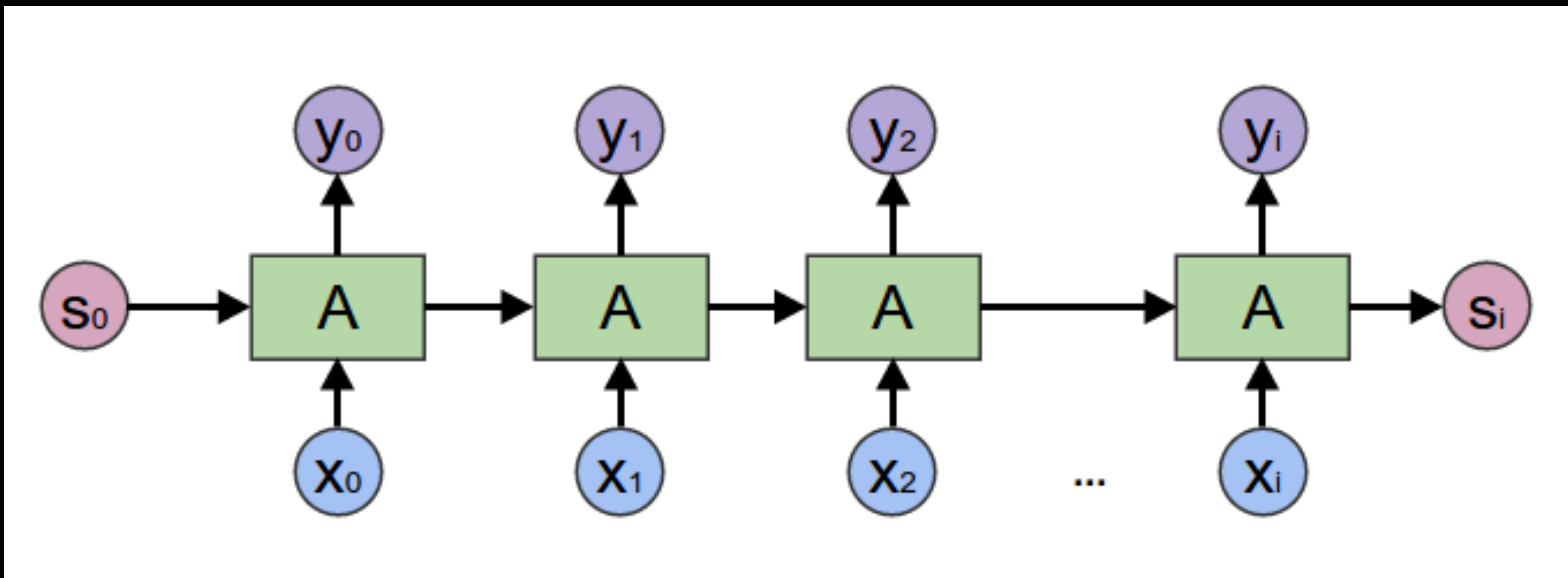
Models

- Recurrent Neural Nets (See: https://colah.github.io/posts/2015-08-Understanding-LSTMs/)
- Simple Extensions of 2D CNNs



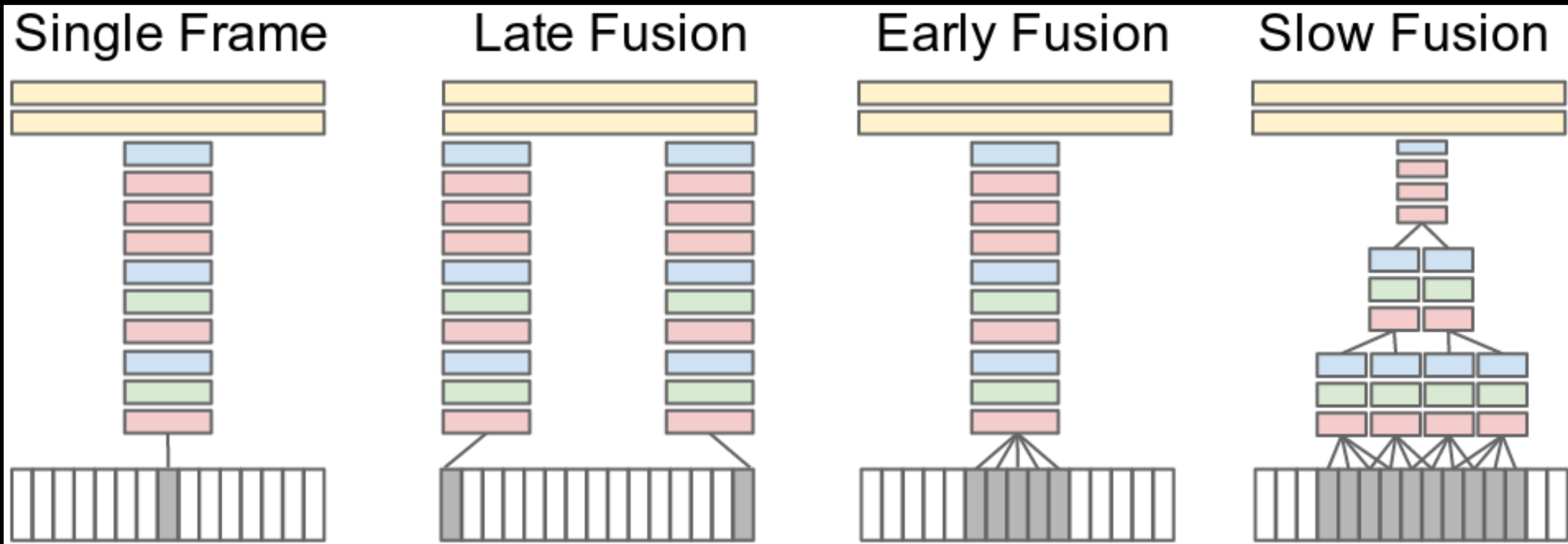
- 3D Convolution Networks
- Two-Stream Networks
- Inflated 3D Conv Nets
- Slow Fast Networks
- Non-local Networks

Recurrent Neural Networks



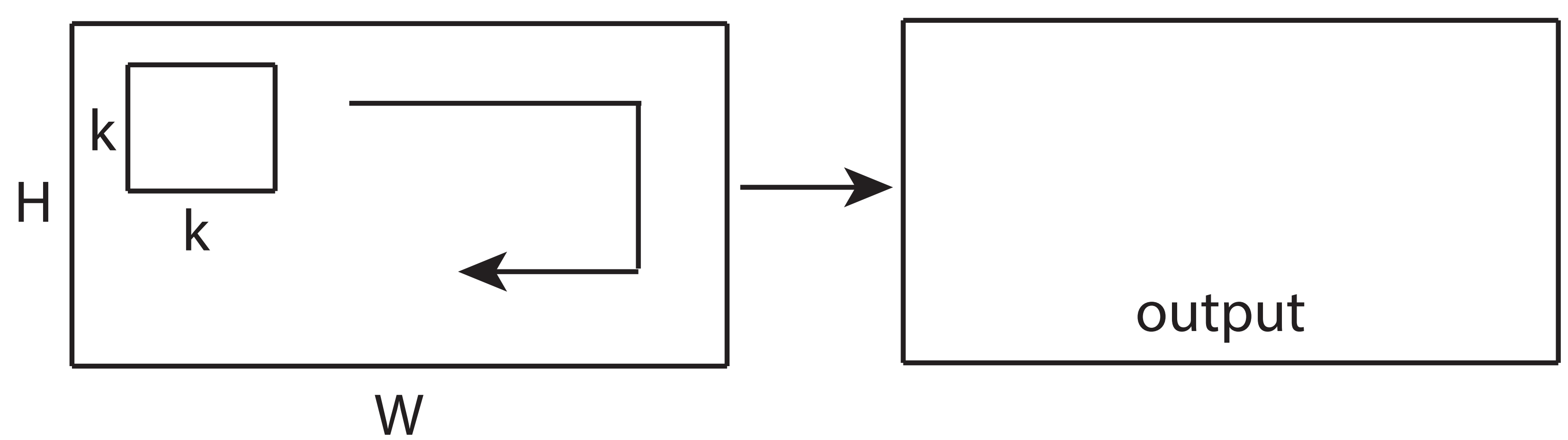
Source: https://colah.github.io/posts/2015-09-NN-Types-FP/

3D Convolutions

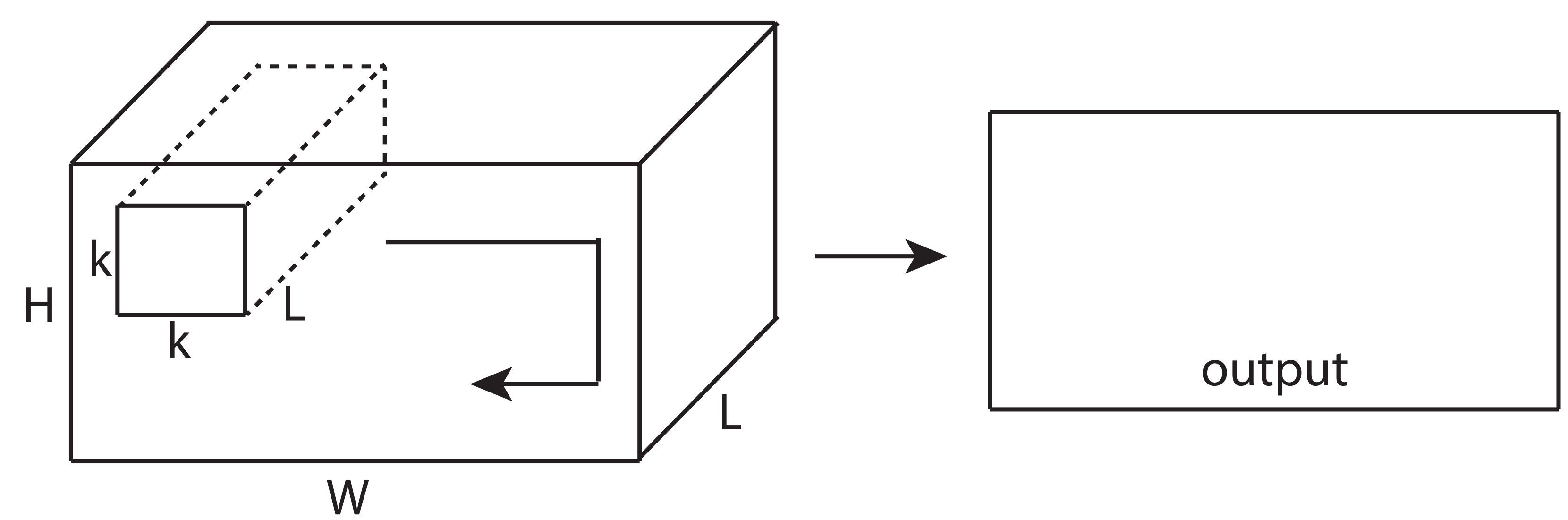


Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks, CVPR 2014

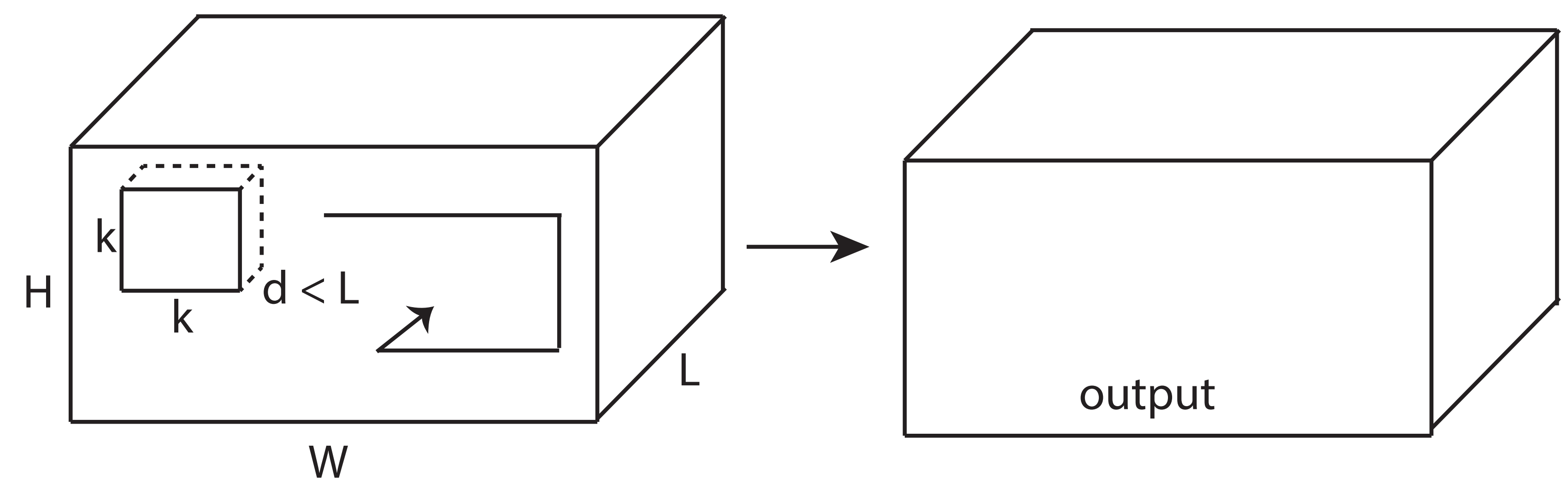
3D Convolutions



(a) 2D convolution

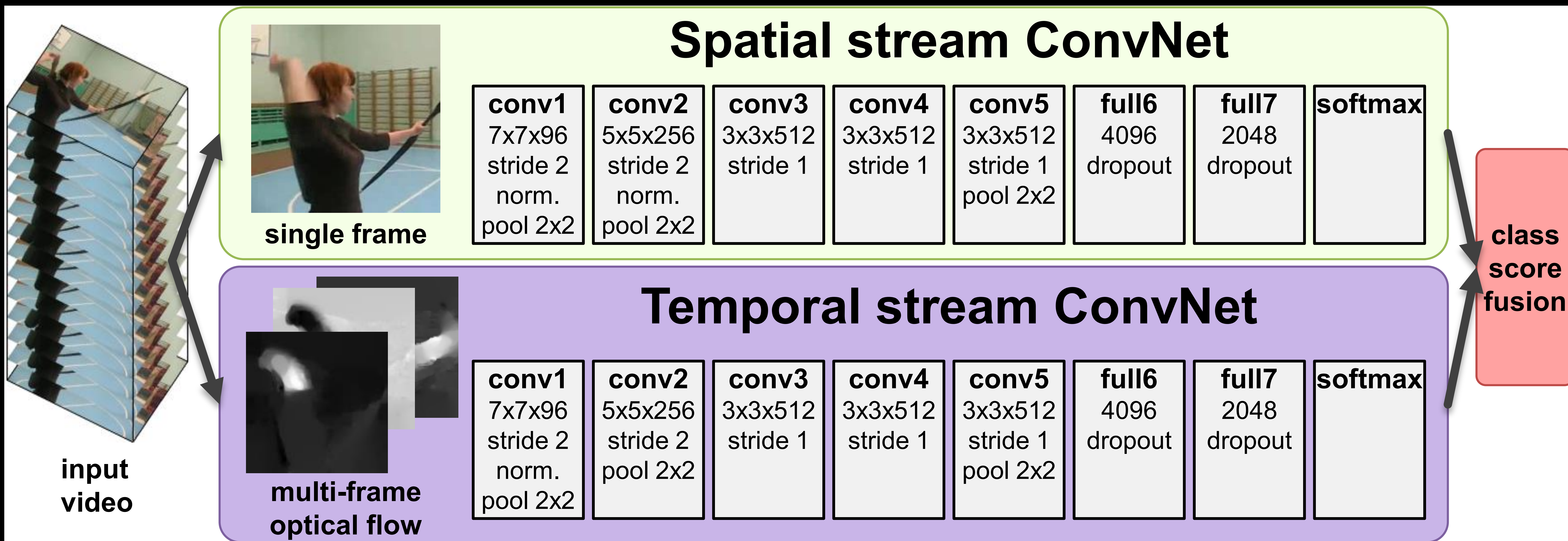


(b) 2D convolution on multiple frames



(c) 3D convolution

Two Stream Networks



Simonyan and Zisserman, Two-Stream Convolutional Networks for Action Recognition in Videos, NIPS 2014

Two Stream Networks

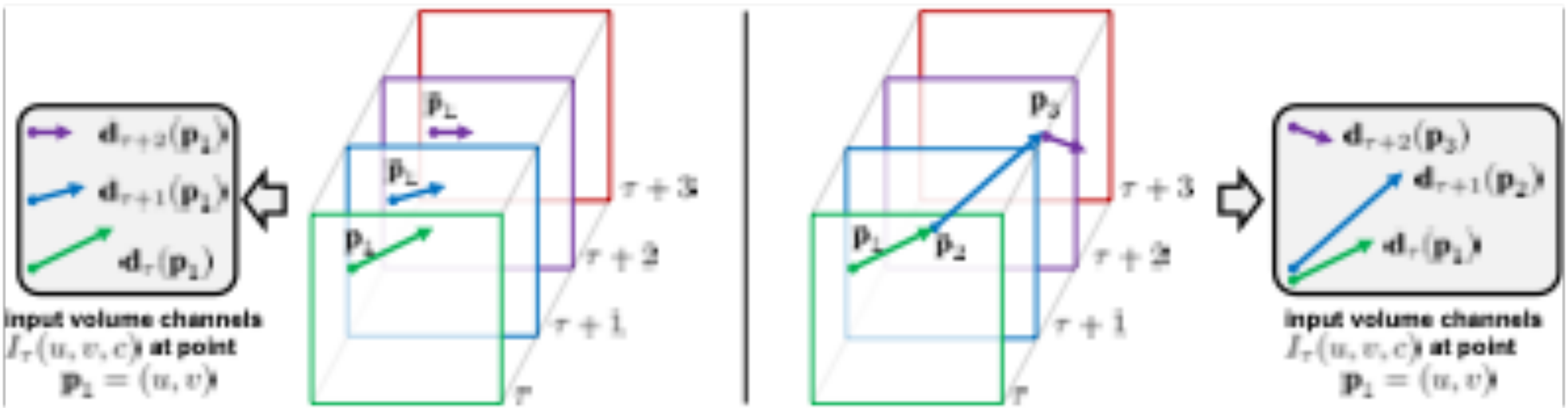


Figure 3: **ConvNet input derivation from the multi-frame optical flow.** *Left:* optical flow stacking (1) samples the displacement vectors **d** at the same location in multiple frames. *Right:* trajectory stacking (2) samples the vectors along the trajectory. The frames and the corresponding displacement vectors are shown with the same colour.

Simonyan and Zisserman, Two-Stream Convolutional Networks for Action Recognition in Videos, NIPS 2014

Two Stream Networks

Table 1: **Individual ConvNets accuracy on UCF-101 (split 1).**

(a) **Spatial ConvNet.**

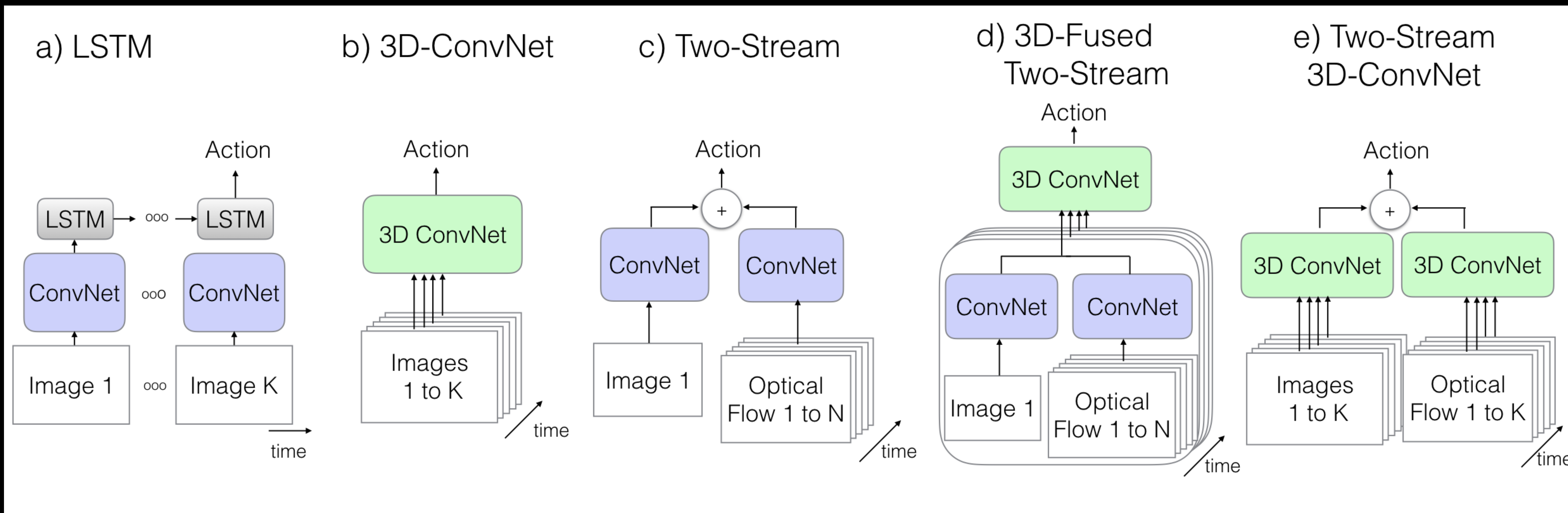
Training setting	Dropout ratio 0.5	Dropout ratio 0.9
From scratch	42.5%	52.3%
Pre-trained + fine-tuning	70.8%	**72.8%**
Pre-trained + last layer	**72.7%**	59.9%

(b) **Temporal ConvNet.**

Input configuration	Mean subtraction off	Mean subtraction on
Single-frame optical flow ($L = 1$)	-	73.9%
Optical flow stacking (1) ($L = 5$)	-	80.4%
Optical flow stacking (1) ($L = 10$)	79.9%	**81.0%**
Trajectory stacking (2)($L = 10$)	79.6%	80.2%
Optical flow stacking (1)($L = 10$), bi-dir.	-	**81.2%**

Simonyan and Zisserman, Two-Stream Convolutional Networks for Action Recognition in Videos, NIPS 2014

Inflated 3D Convolutions



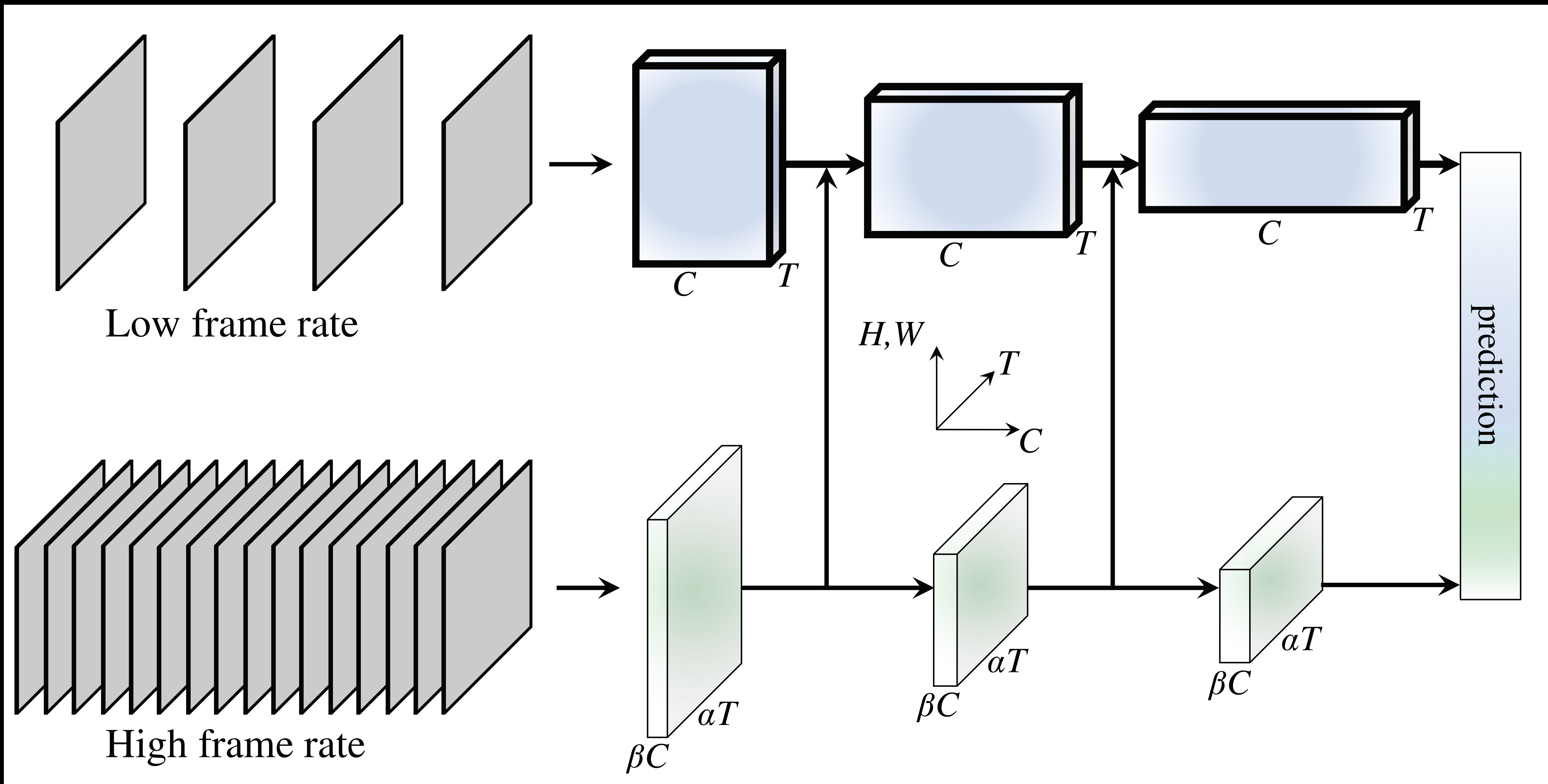
Joao Carreira, Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset, CVPR 2017

Inflated 3D Convolutions

Architecture	UCF-101			HMDB-51			Kinetics		
	RGB	Flow	RGB + Flow	RGB	Flow	RGB + Flow	RGB	Flow	RGB + Flow
(a) LSTM	81.0	–	–	36.0	–	–	63.3	–	–
(b) 3D-ConvNet	51.6	–	–	24.3	–	–	56.1	–	–
(c) Two-Stream	83.6	85.6	91.2	43.2	56.3	58.3	62.2	52.4	65.6
(d) 3D-Fused	83.2	85.8	89.3	49.2	55.5	56.8	–	–	67.2
(e) Two-Stream I3D	**84.5**	**90.6**	**93.4**	**49.8**	**61.9**	**66.4**	**71.1**	**63.4**	**74.2**

Joao Carreira, Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset, CVPR 2017

SlowFast Networks



Christoph Feichtenhofer et al., Quo Vadis, SlowFast Networks for Video Recognition, CVPR 2019

SlowFast Networks

stage	*Slow* pathway	*Fast* pathway	output sizes $T\times S^2$
raw clip	-	-	64×224^2
data layer	stride 16, 1^2	stride **2**, 1^2	*Slow* : 4×224^2 *Fast* : 32×224^2
conv_1	1×7^2, 64 stride 1, 2^2	5×7^2, 8 stride 1, 2^2	*Slow* : 4×112^2 *Fast* : 32×112^2
pool_1	1×3^2 max stride 1, 2^2	1×3^2 max stride 1, 2^2	*Slow* : 4×56^2 *Fast* : 32×56^2
res_2	$\begin{bmatrix}1\times1^2, 64\\1\times3^2, 64\\1\times1^2, 256\end{bmatrix}\times3$	$\begin{bmatrix}3\times1^2, 8\\1\times3^2, 8\\1\times1^2, 32\end{bmatrix}\times3$	*Slow* : 4×56^2 *Fast* : 32×56^2
res_3	$\begin{bmatrix}1\times1^2, 128\\1\times3^2, 128\\1\times1^2, 512\end{bmatrix}\times4$	$\begin{bmatrix}3\times1^2, 16\\1\times3^2, 16\\1\times1^2, 64\end{bmatrix}\times4$	*Slow* : 4×28^2 *Fast* : 32×28^2
res_4	$\begin{bmatrix}3\times1^2, 256\\1\times3^2, 256\\1\times1^2, 1024\end{bmatrix}\times6$	$\begin{bmatrix}3\times1^2, 32\\1\times3^2, 32\\1\times1^2, 128\end{bmatrix}\times6$	*Slow* : 4×14^2 *Fast* : 32×14^2
res_5	$\begin{bmatrix}3\times1^2, 512\\1\times3^2, 512\\1\times1^2, 2048\end{bmatrix}\times3$	$\begin{bmatrix}3\times1^2, 64\\1\times3^2, 64\\1\times1^2, 256\end{bmatrix}\times3$	*Slow* : 4×7^2 *Fast* : 32×7^2
global average pool, concate, fc			# classes

Table 1. **An example instantiation of the SlowFast network**. The dimensions of kernels are denoted by $\{T\times S^2, C\}$ for temporal, spatial, and channel sizes. Strides are denoted as {temporal stride, spatial stride^2}. Here the speed ratio is $\alpha = 8$ and the channel ratio is $\beta = 1/8$. τ is 16. The green colors mark *higher* temporal resolution, and orange colors mark *fewer* channels, for the Fast pathway. Non-degenerate temporal filters are underlined. Residual blocks are shown by brackets. The backbone is ResNet-50.

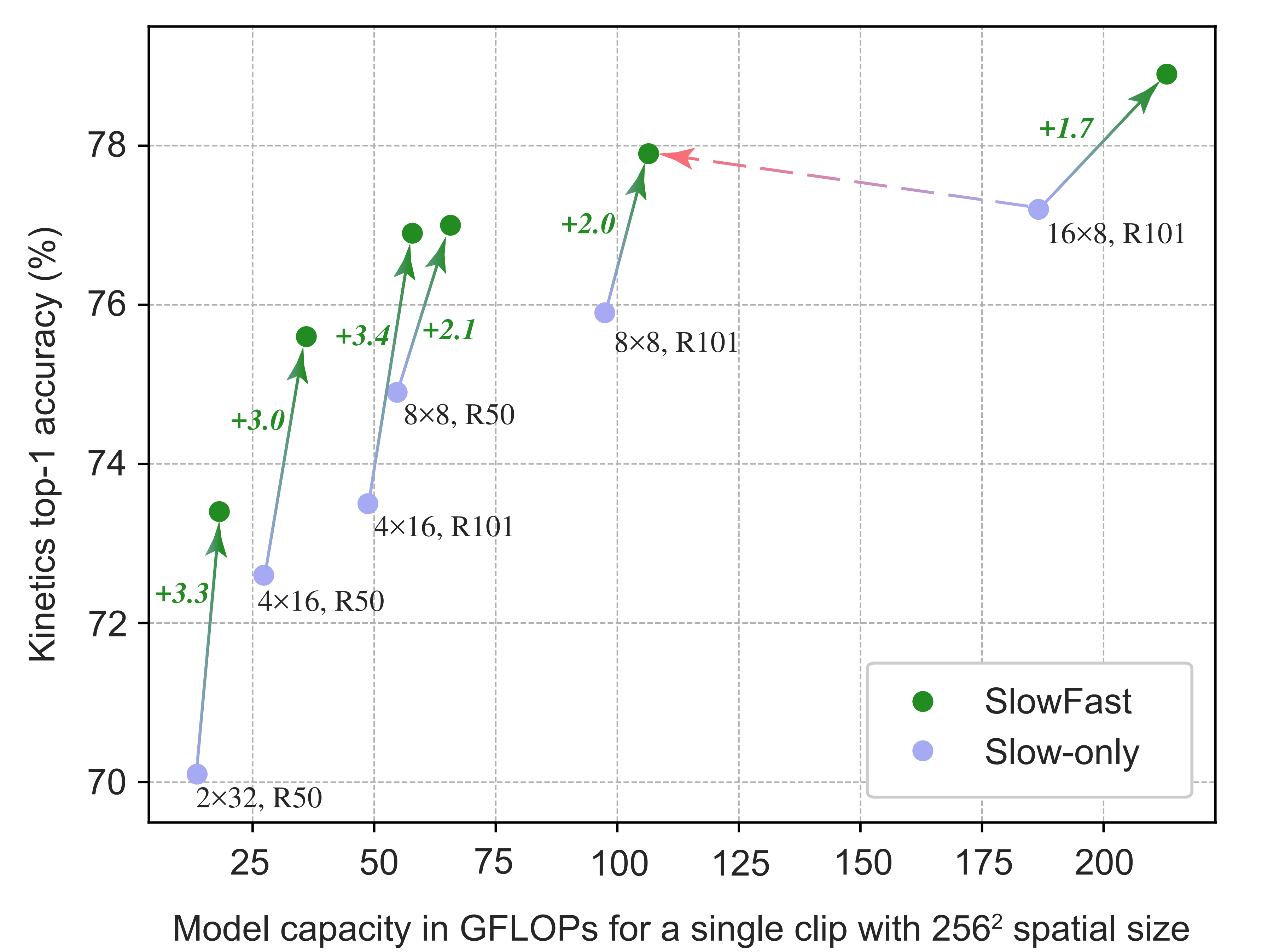
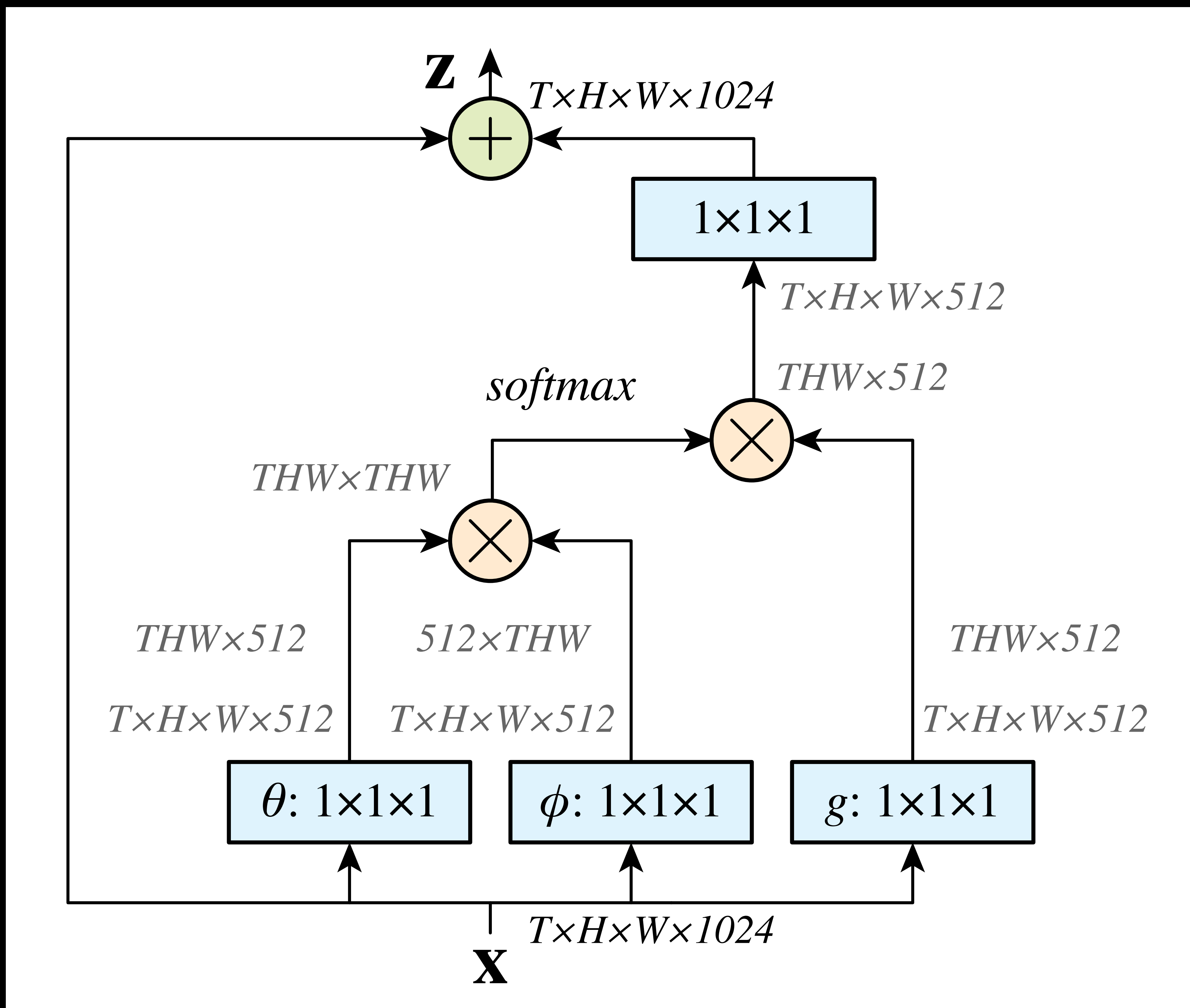


Figure 2. **Accuracy/complexity tradeoff** on Kinetics-400 for the SlowFast (green) *vs*. Slow-only (blue) architectures. SlowFast is consistently better than its Slow-only counterpart in all cases (green arrows). SlowFast provides higher accuracy *and* lower cost than temporally heavy Slow-only (*e.g*. red arrow). The complexity is for a single 256^2 view, and accuracy are obtained by 30-view testing.

Christoph Feichtenhofer et al., Quo Vadis, SlowFast Networks for Video Recognition, CVPR 2019

Non-local Networks



$$\mathbf{y}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{\forall j} f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j).$$

Xiaolong Wang et al., Non-local Neural Networks, CVPR 2018

Non-local Networks

Xiaolong Wang et al., [Non-local Neural Networks](), CVPR 2018