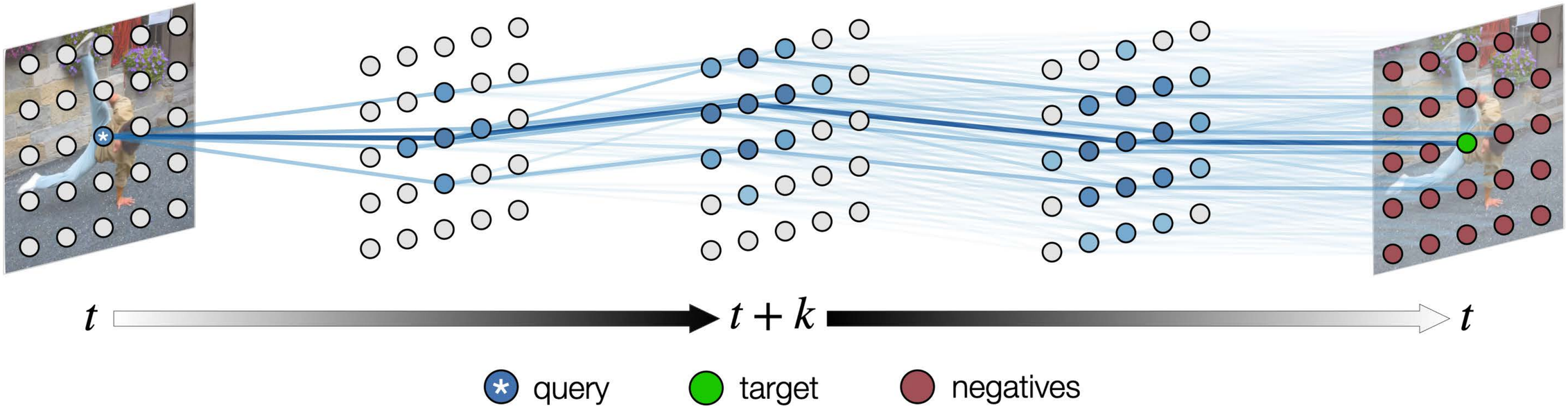
Using Video to Learn about Visual Correspondence

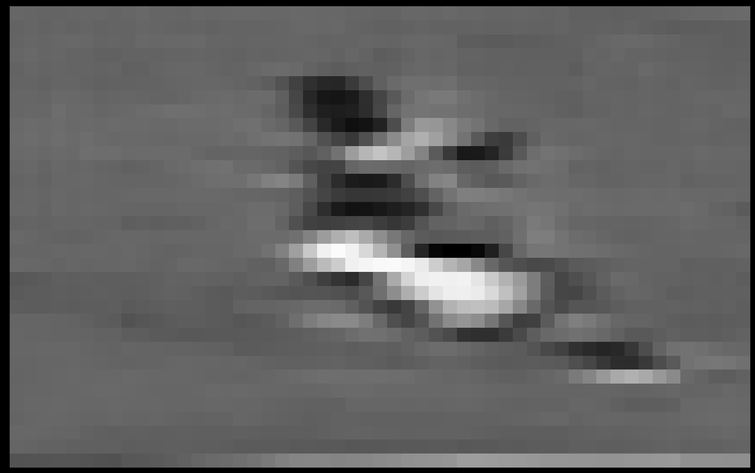


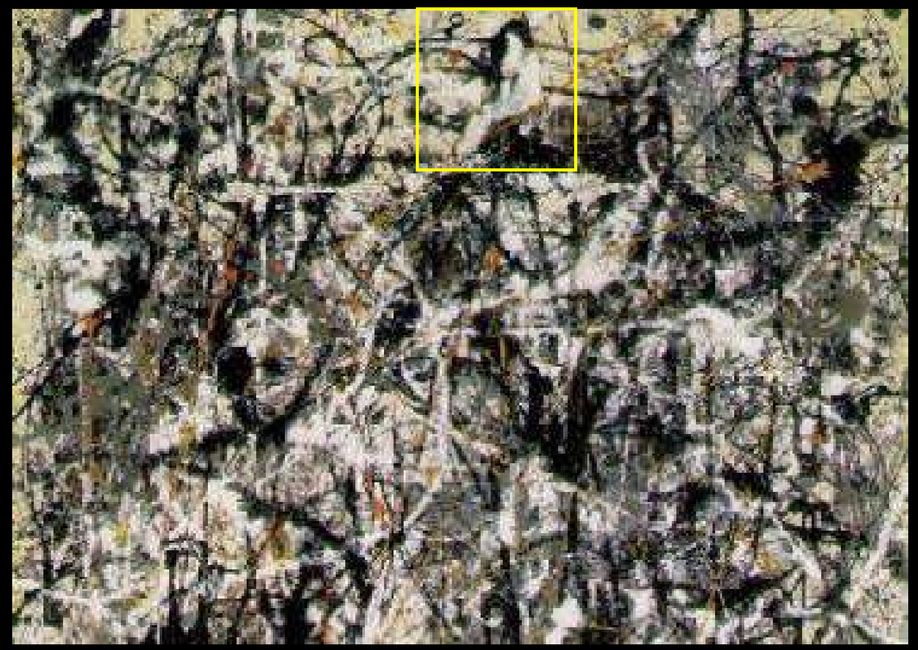
Alexei (Alyosha) Efros
UC Berkeley

BAIR
BERKELEY ARTIFICIAL INTELLIGENCE RESEARCH

Why video?

1. Richer signal





Jackson Pollock
Number 21 (detail)

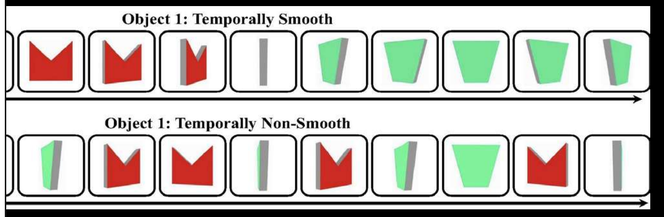
2. Correspondences

"It irritated him that the "dog" of 3:14 in the afternoon, seen in profile, should be indicated by the same noun as the dog of 3:15, seen frontally…"

-- from *Funes the Memorious*

Jorge Luis Borges

Continuity crucial for visual development

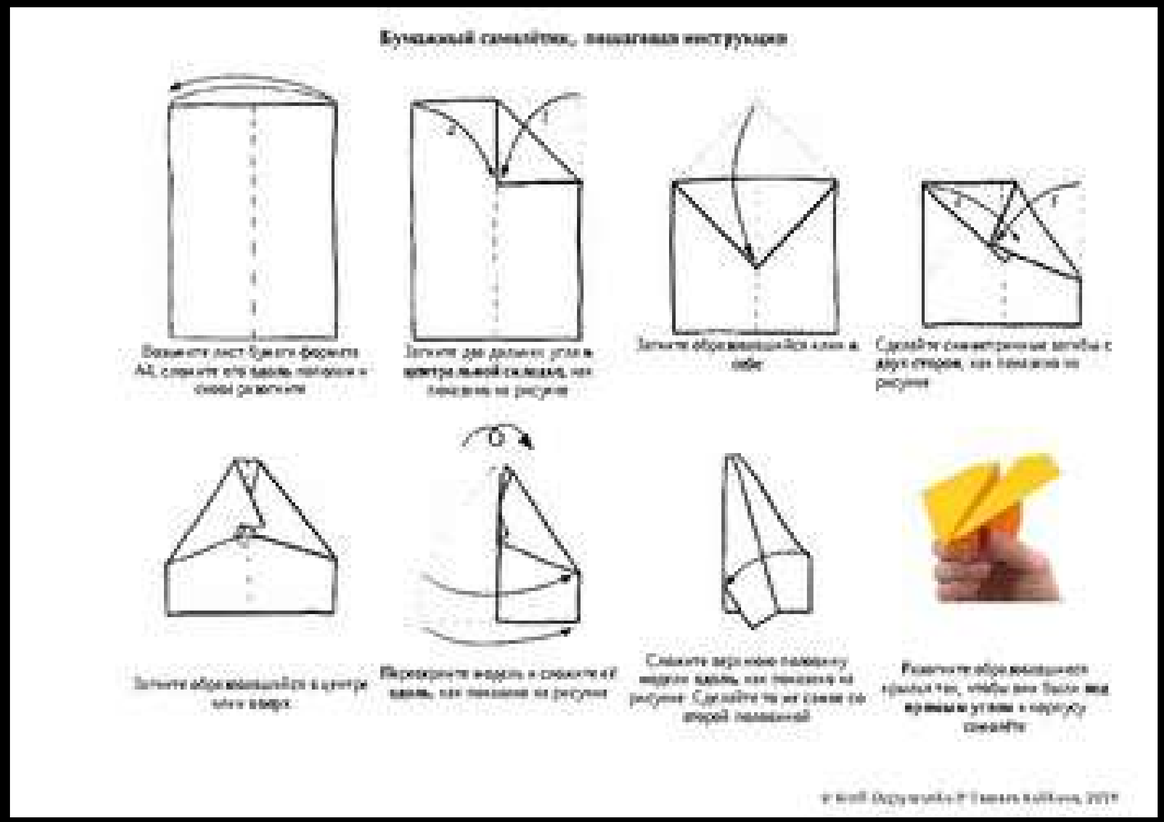


Wood 2013, 2016, 2018

3. Ordering?

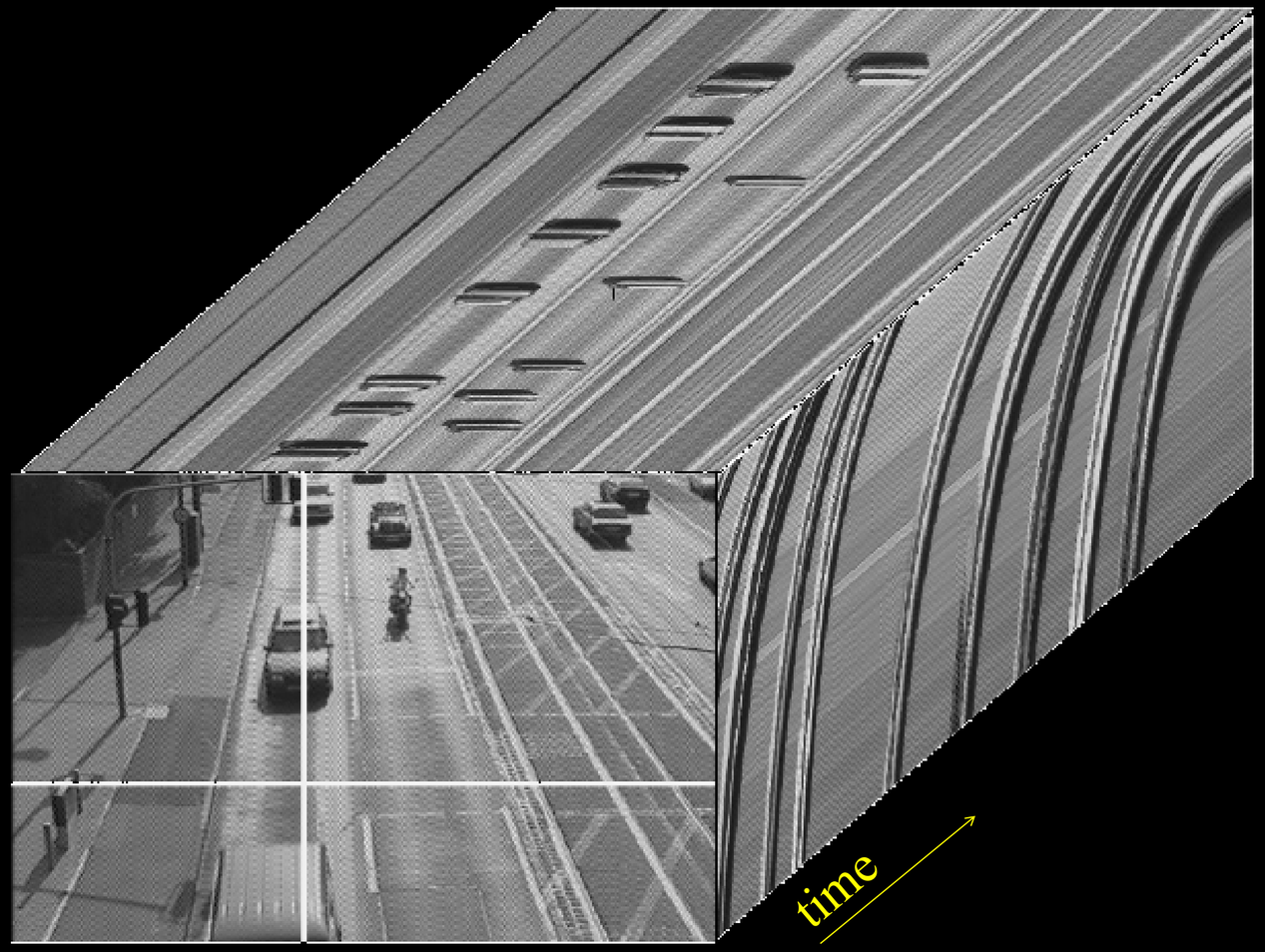
“Time is what keeps everything from happening at once.”

-- Ray Cummings (1919)



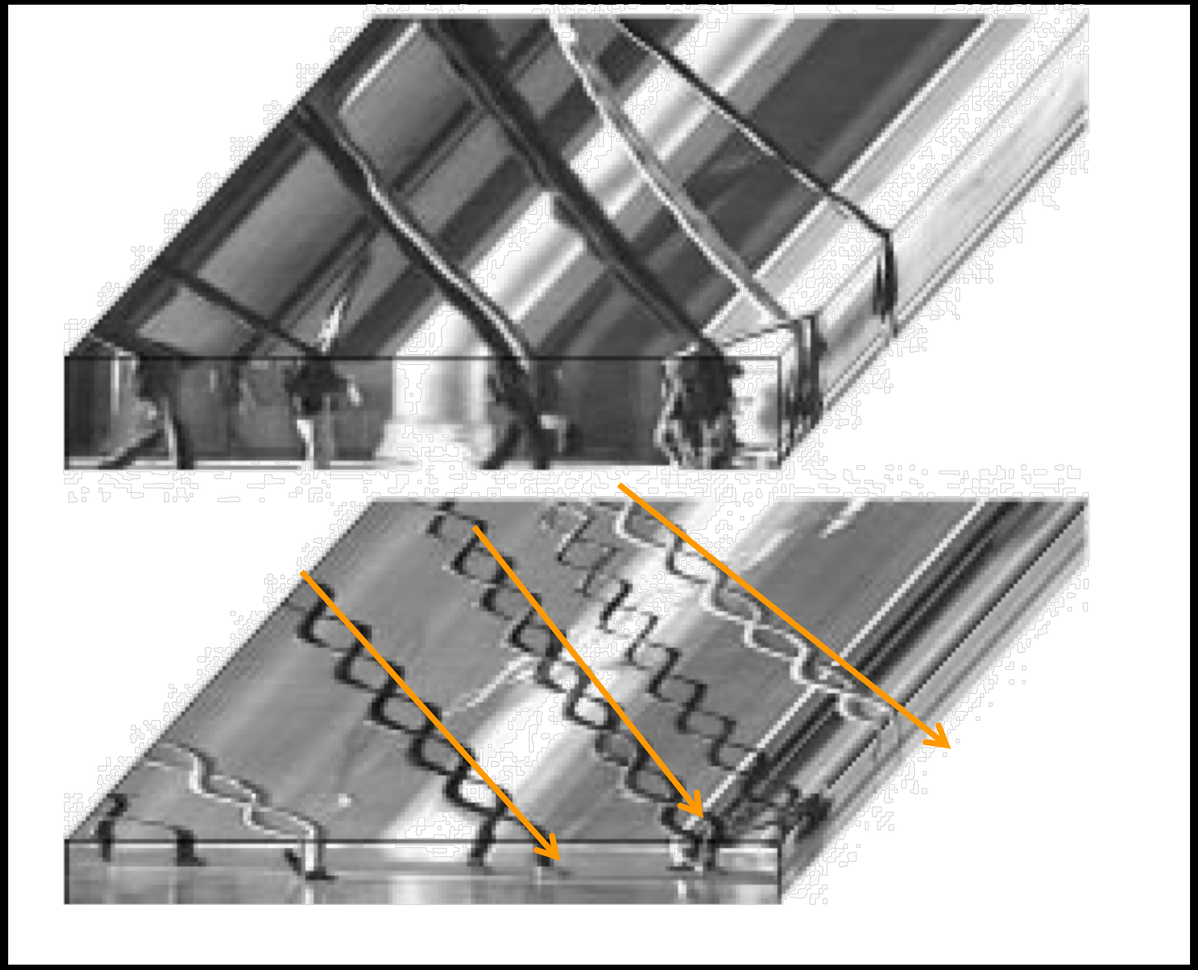
Video in the “old days”

Space-time *XYT* volumes

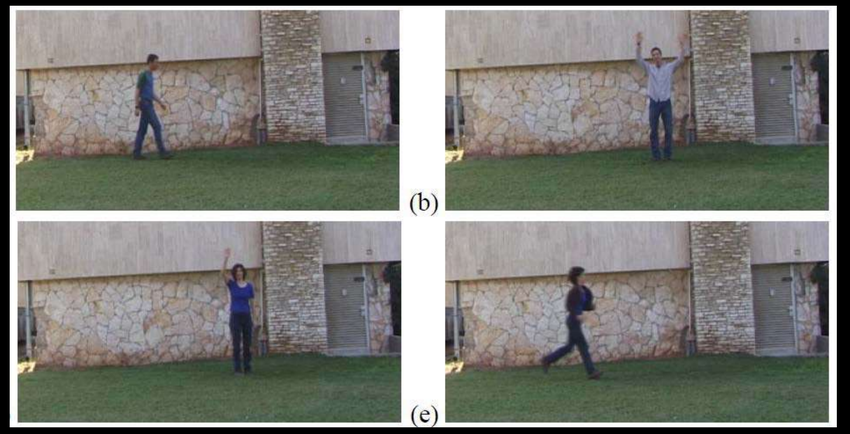


Clean and beautiful story

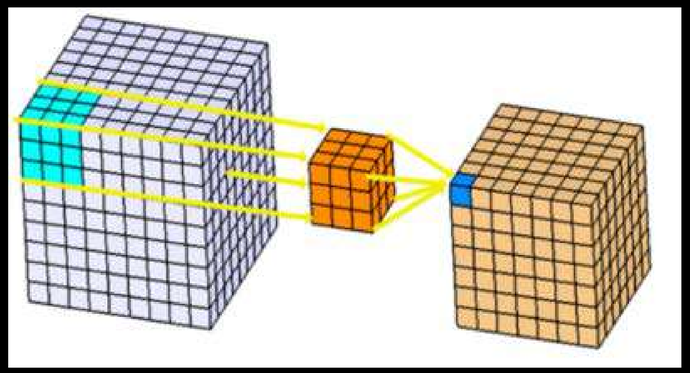
Recognition with *XYT* volumes



“braided patterns” of cyclic motion
[Niyogi & Adelson, 1994]

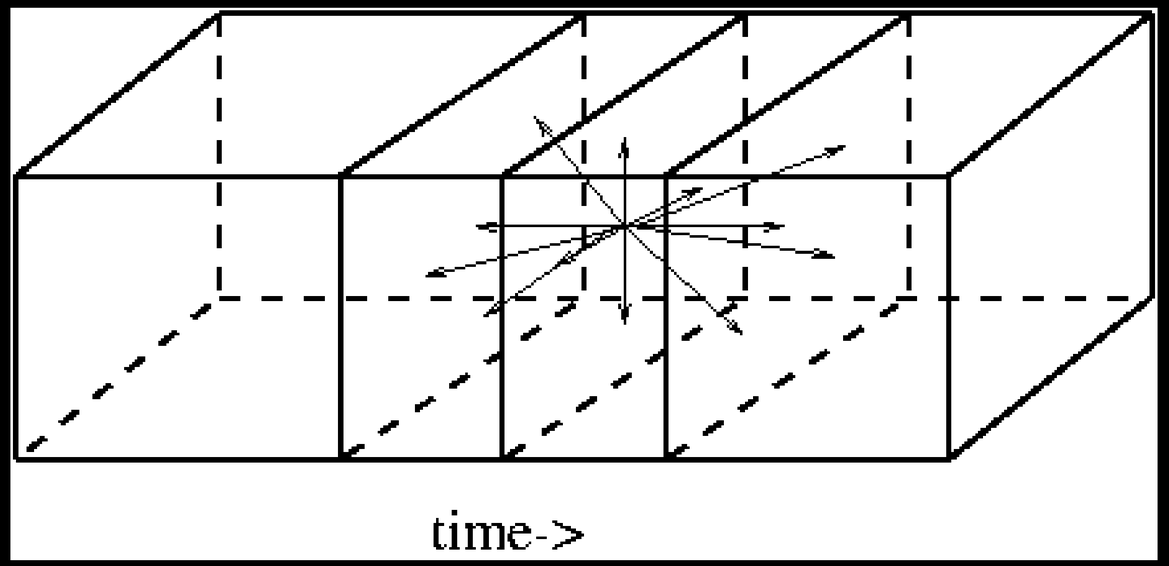


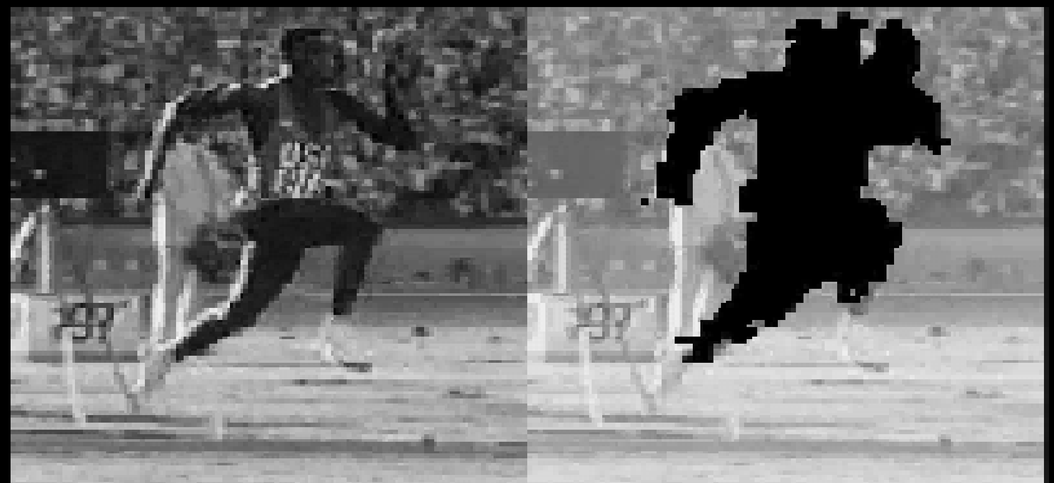
histograms of spatio-temporal gradients
[Zelnik-Manor & Irani, 2001]



3D Convolutions
[Tran et al, 2015]

XYT segmentation [Shi & Malik, 98]





Video Textures

Arno Schödl
Richard Szeliski
David Salesin
Irfan Essa

SIGGRAPH **2000**

Problem statement

video clip

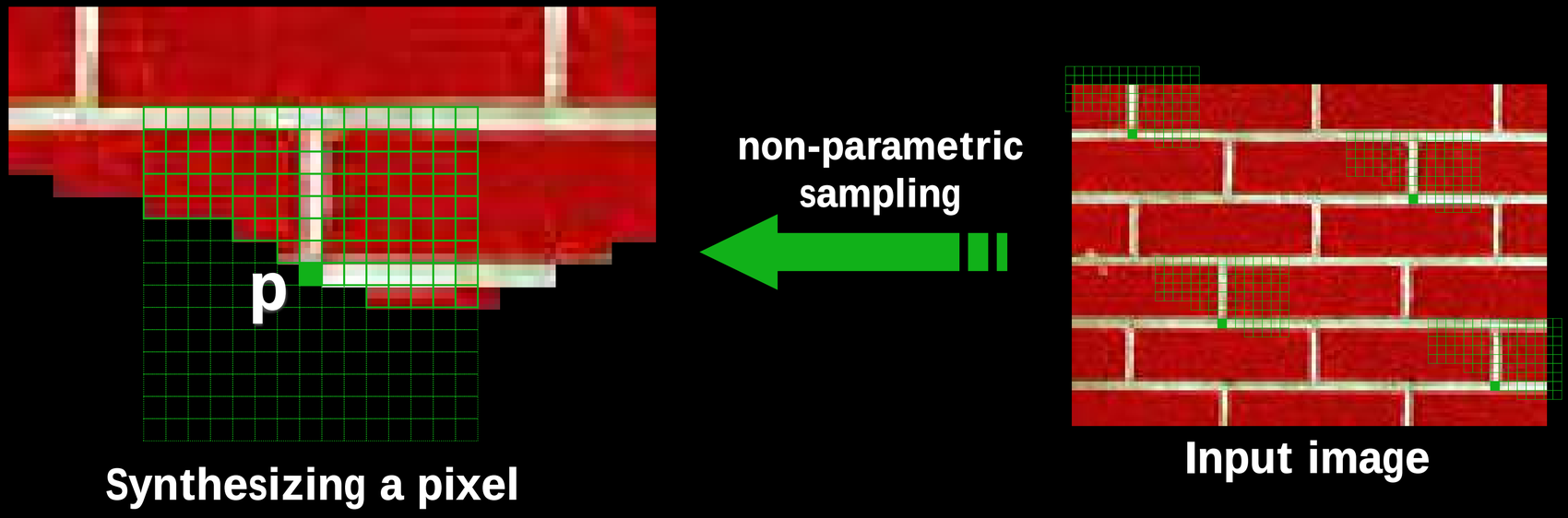
video texture

Text Synthesis

- [Shannon,’48] proposed a way to generate English-looking text using N-grams:
  - Assume a generalized Markov model
  - Use a large text to compute prob. distributions of each letter given N-1 previous letters
  - Starting from a seed repeatedly sample this Markov chain to generate new letters
  - Also works for whole words

WE NEED TO EAT CAKE

Texture Synthesis: Markov chain on pixels



Efros & Leung, ICCV 1999

Video Textures: Markov chain on frames

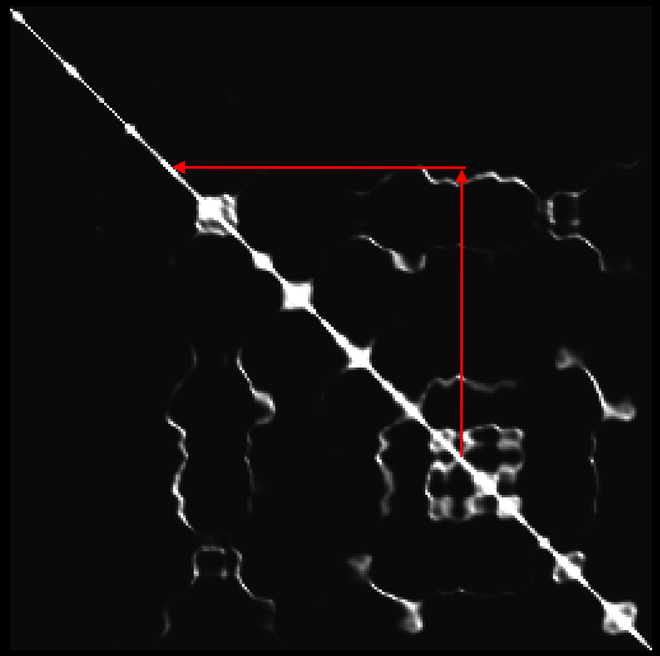
- How do we find good transitions?

Finding good transitions

- Compute L_2 distance $D_{i,j}$ between all frames

vs. frame *i*

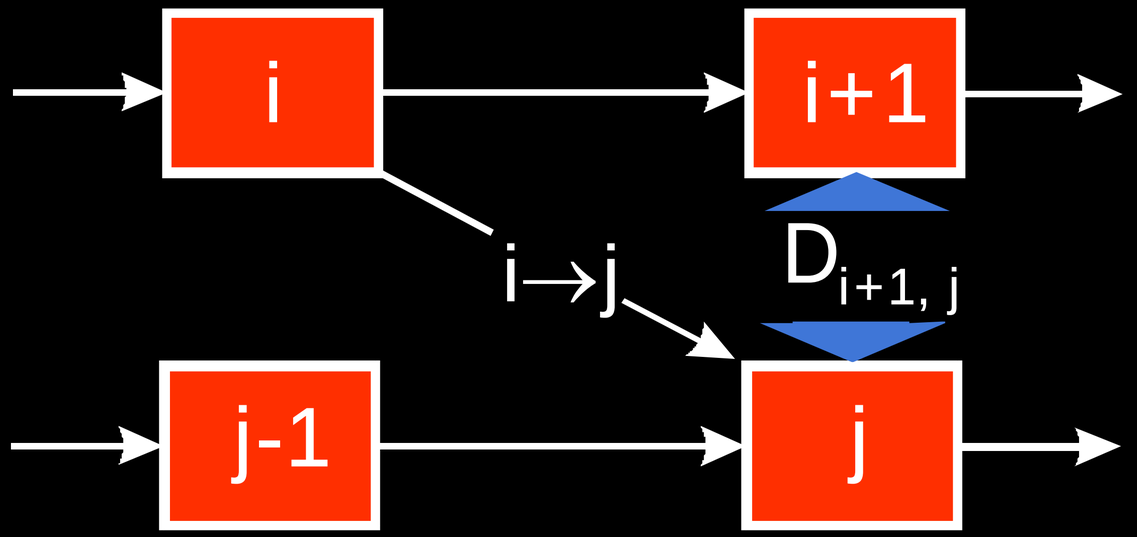
frame *j*



Similar frames make good transitions

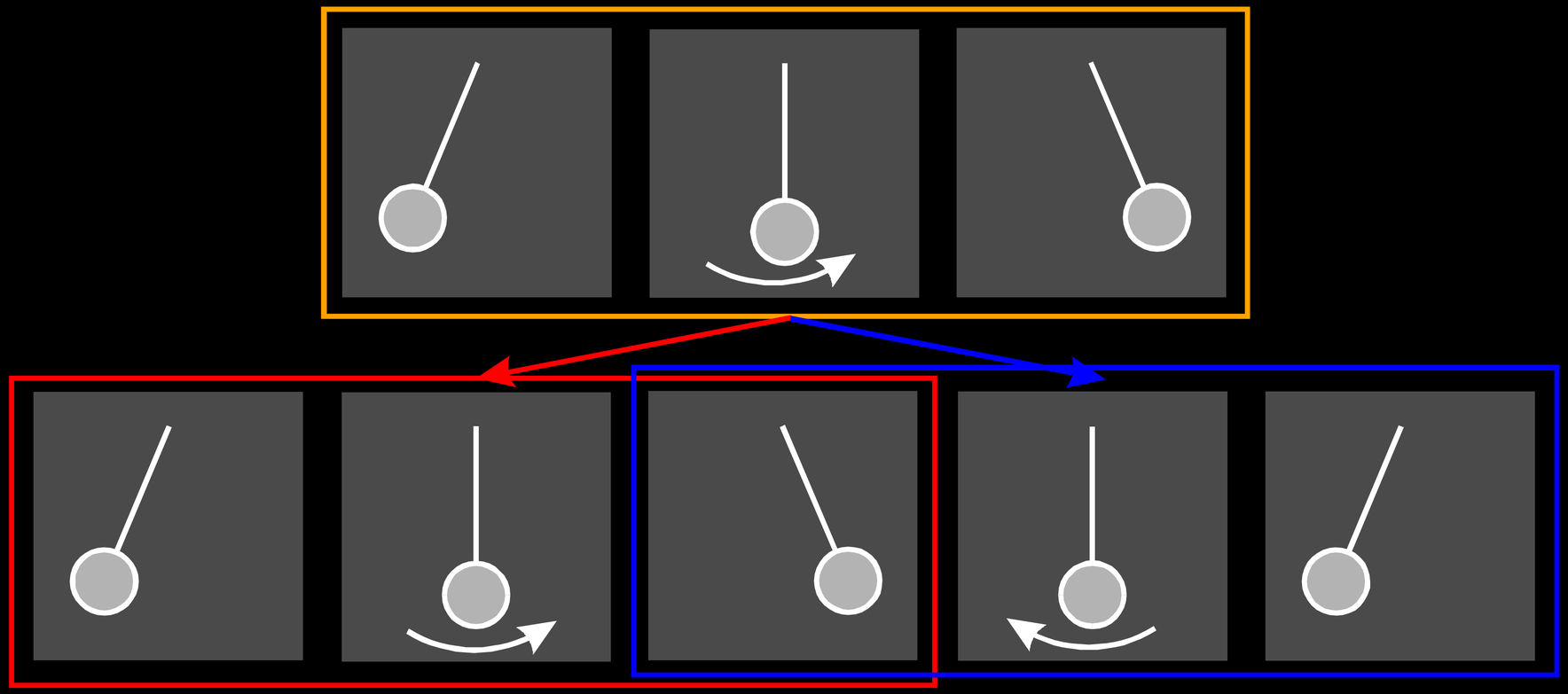
Transition costs

- Transition from i to j if successor of i is similar to j
  - Cost function: $C_{i \to j} = D_{i+1, j}$
-



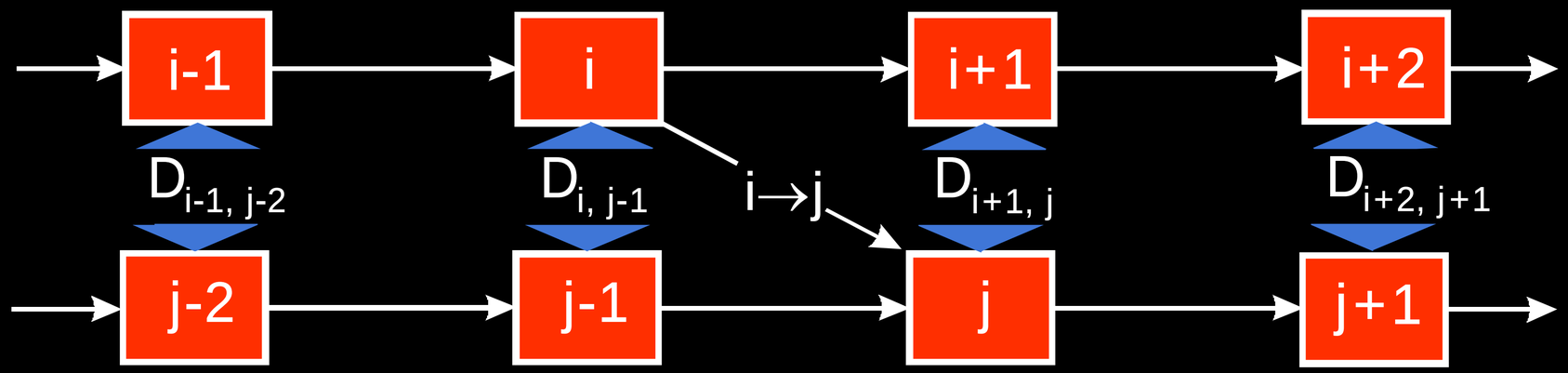
Preserving dynamics

Preserving dynamics



Preserving dynamics

- Cost for transition $i \rightarrow j$
  - $C_{i \rightarrow j} = \sum_{k=-N}^{N-1} w_k \, D_{i+k+1,\, j+k}$



Preserving dynamics – effect

- Cost for transition $i \rightarrow j$
  - $C_{i \rightarrow j} = \sum_{k=-N}^{N-1} w_k \, D_{i+k+1,\, j+k}$

User-controlled video textures

slow

variable

fast

User selects target frame range

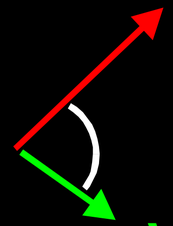
Video sprite extraction

blue screen matting
and velocity estimation

Video sprite control

- Augmented transition cost:

$$C^{\text{Animation}}_{i \to j} = \alpha \underbrace{C_{i \to j}}_{\text{Similarity term}} + \beta \underbrace{\text{angle}}_{\text{Control term}}$$



vector to
mouse pointer

velocity vector

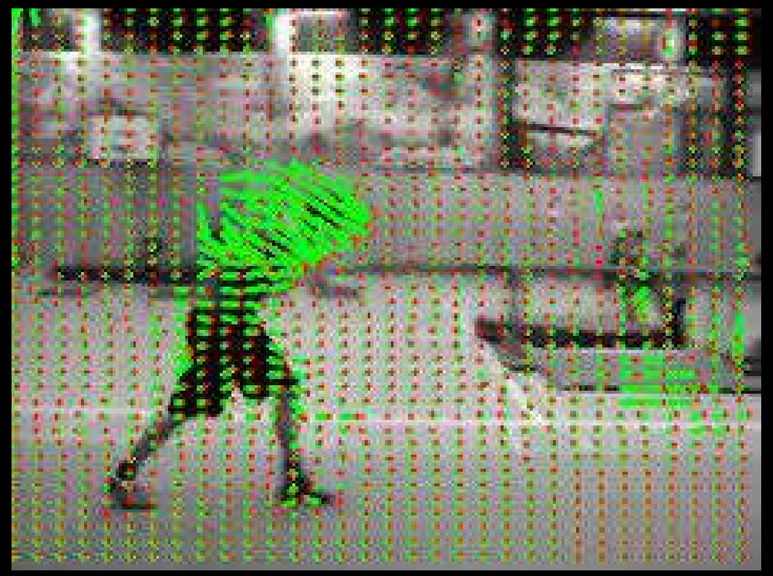
Interactive fish

Trouble with *XYT* volumes

- Time is not just “another dimension”
  - Very sparse sampling in *t*
- Implicit correspondence assumed
  - *e.g. (x,y,t) → (x,y,t+1)* (fixed camera)
  - or other simple models

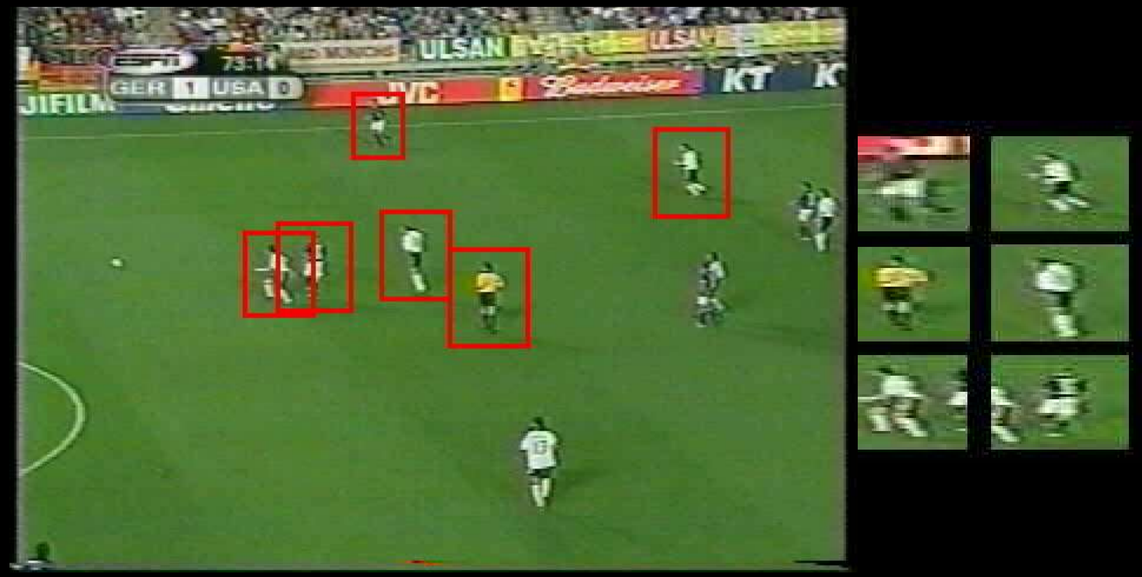
Explicit Correspondences

Optical Flow



+ dense correspondences
- short-range (2 frames)
- local

Object Tacking



+ longer-range & mid-level
- one object at a time
- not stable
  - “time to failure” metric

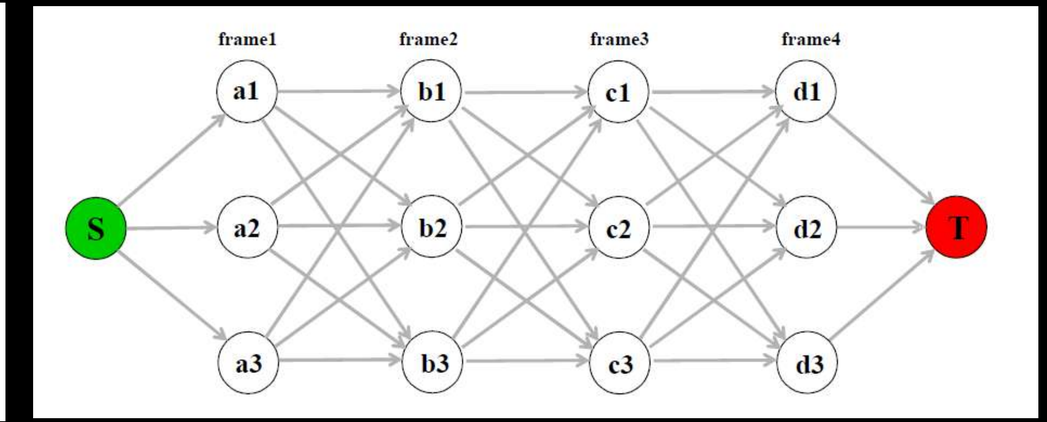
Tracking-by-Detection

“Strike a Pose” [Ramanan, Forsyth, Zisserman, 2005]

- Enter: **learning**
- Train simple person detector on one frame, then detect in all other frames
- Now fancy off-the-shelf detectors
- But no explicit correspondence

Data association

- Temporal correspondence across detections
  - Step 1: detect objects (requires supervision)
  - Step 2: find correspondences across detections (but divorced from pixels)



Using Time as Supervision

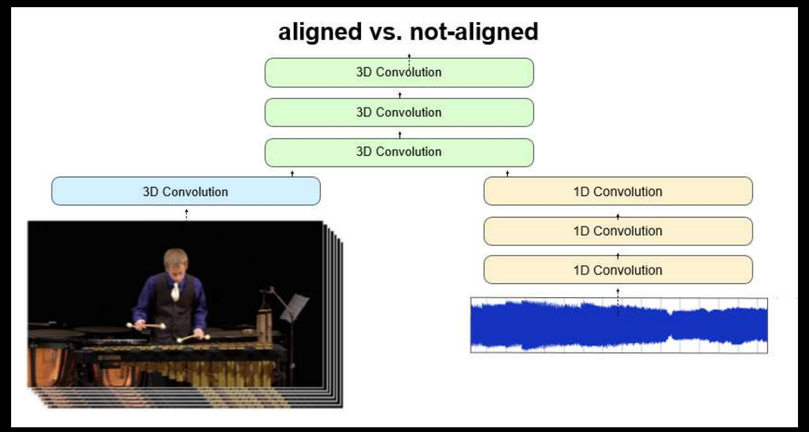
Temporal Self-supervised Feature Learning tasks

Inputs

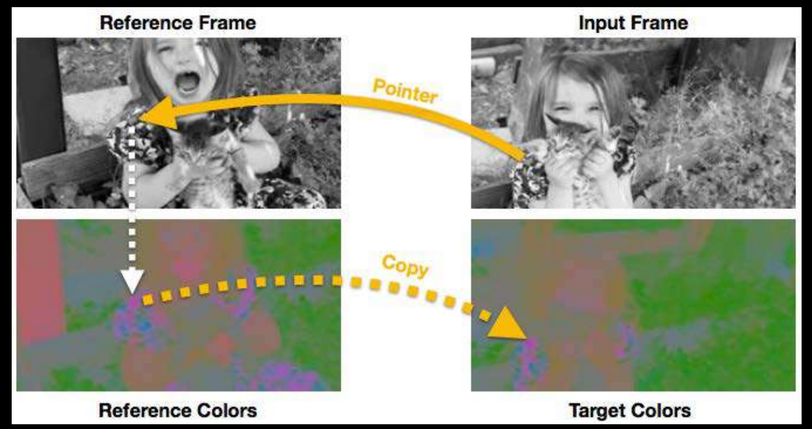
Outputs

Predict Pixels in Time
[Mathieu et al, 2015]

Predict Arrow of Time
[Misra et al, 2016; Wei et al, 2018]

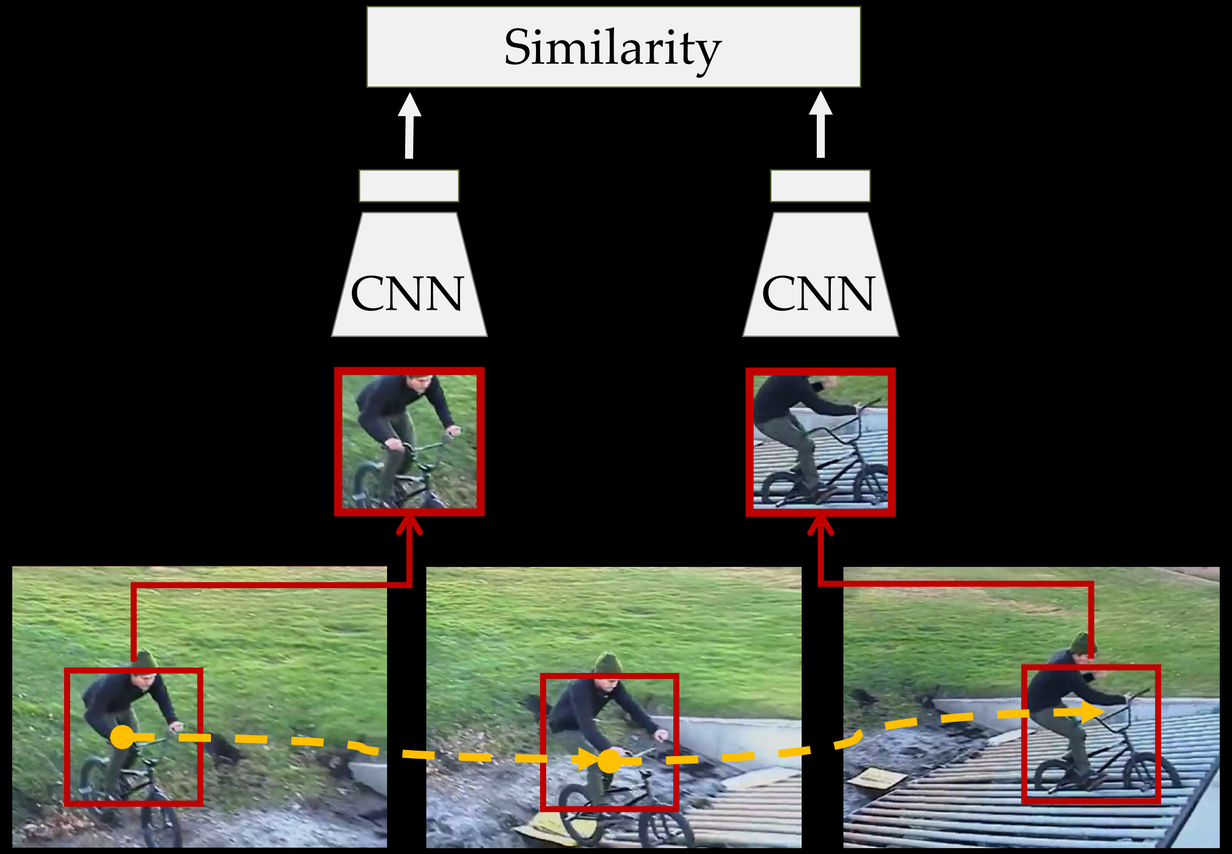


Predict Audio-visual Shifts
[Owens & Efros, 2018]



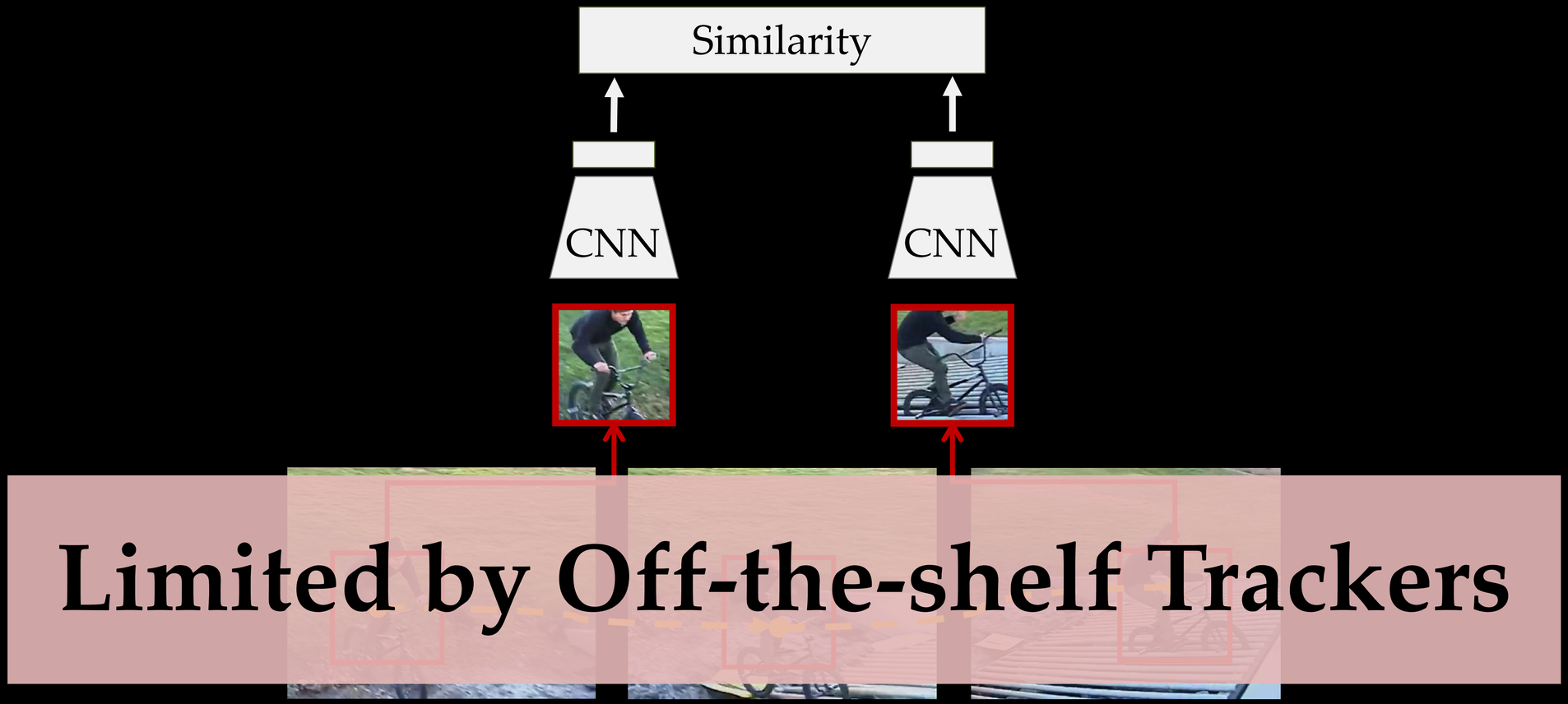
Predict Color in Time
[Vondrick et al, 2018]

Using Tracking to Learn Features



Tracking → Similarity
[Wang et al, 2015; Pathak et al, 2017]

Using Tracking to Learn Features



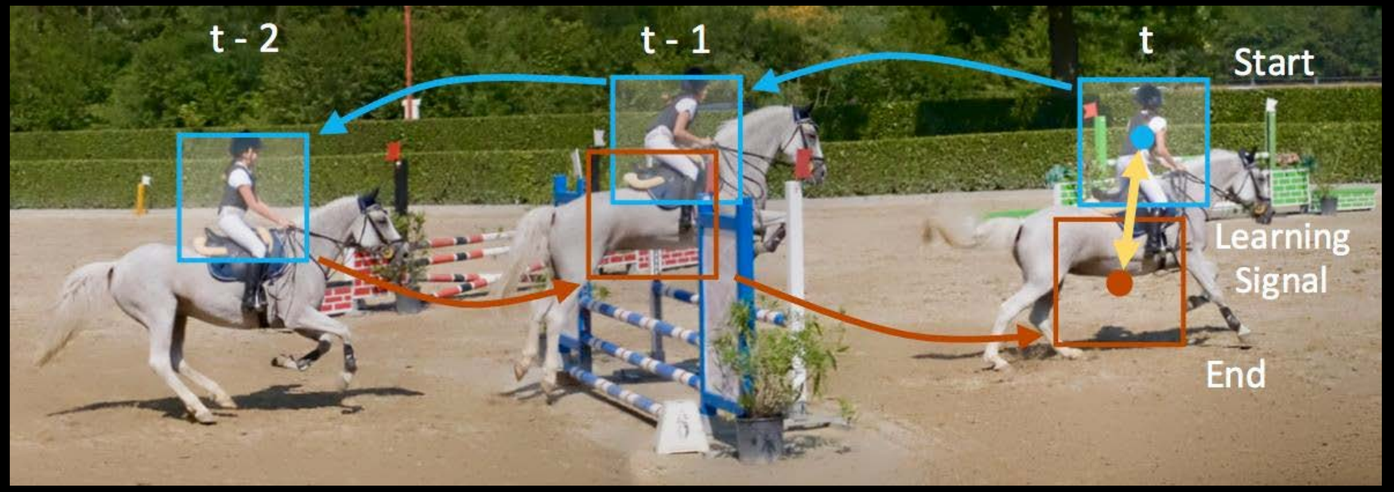
Limited by Off-the-shelf Trackers

Tracking → Similarity
[Wang et al, 2015; Pathak et al, 2017]

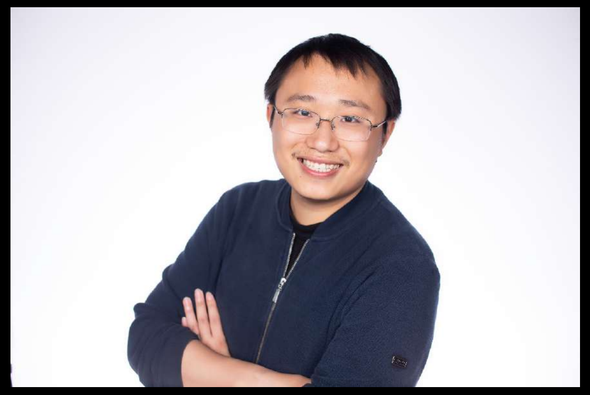
Similarity requires tracking

Tracking requires similarity

Let’s jointly learn both!



Learning Correspondence from the Cycle-consistency of Time

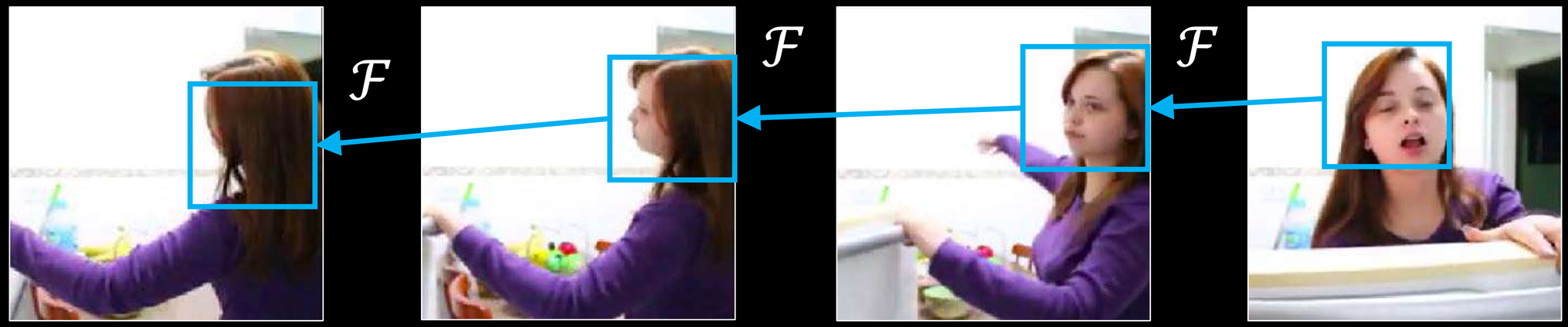


Xiaolong Wang Allan Jabri Alexei Efros

CVPR 2019

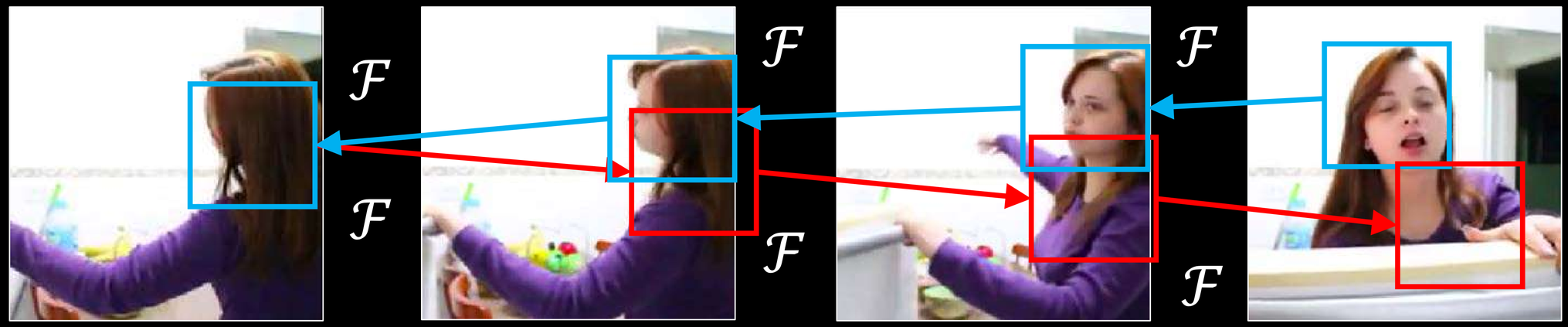
Learning to Track

$\mathcal{F}$: a deep tracker



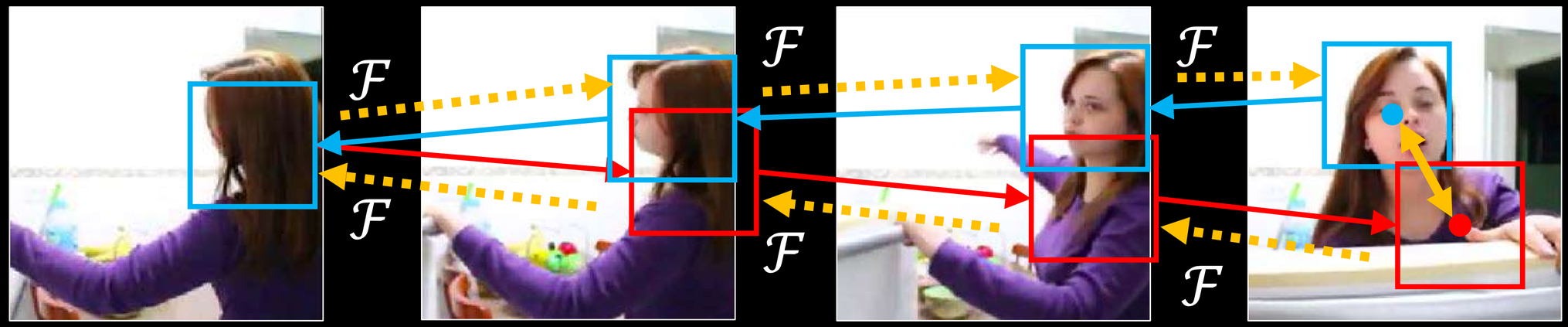
Supervision: Cycle-Consistency in Time

Track backwards



Track forwards, back to the future

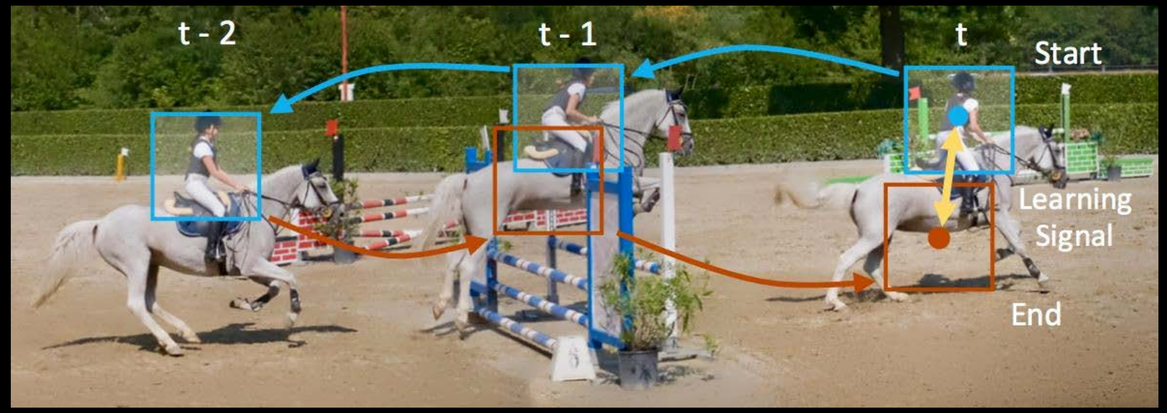
Supervision: Cycle-Consistency in Time



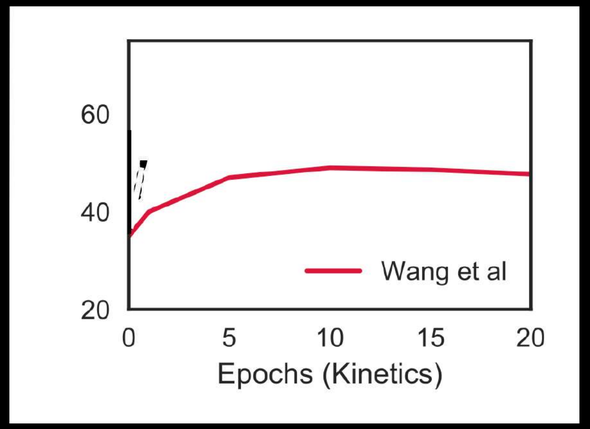
Backpropagation through time along the cycle

Visualization of Training

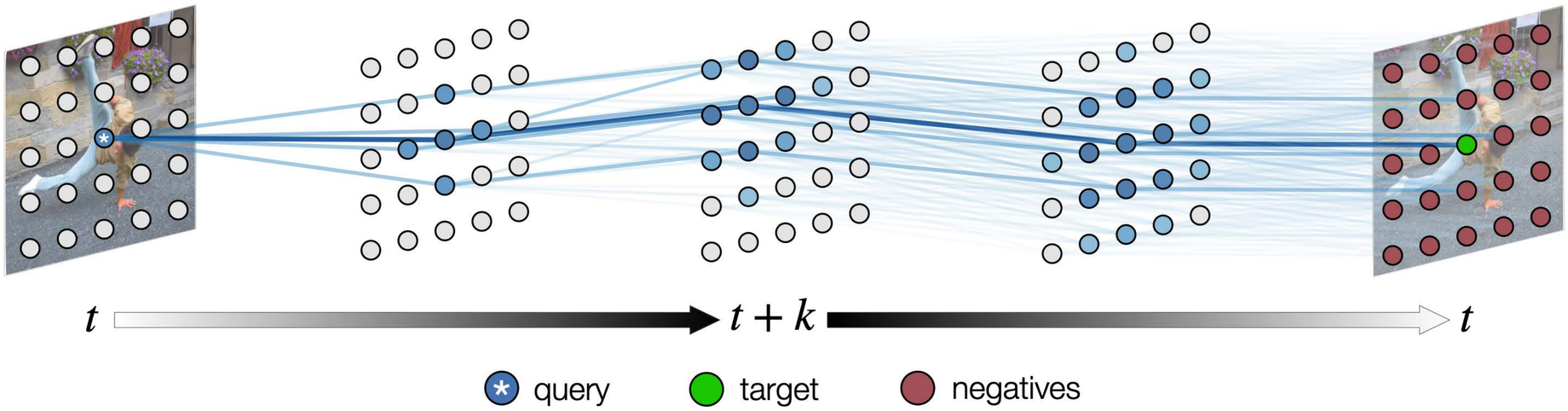
Limitations



- One patch at a time
- Winner-takes-all tracking
- Complex tracker (Spatial Transformer)
- Does not improve with longer cycles (6+ frames)
- Does not improve with more training data
- Fresh new work addresses these…



Space-Time Correspondence as a Contrastive Random Walk



Allan Jabri, Andrew Owens, Alexei A. Efros

(on ArXiv next week!)

Aim:

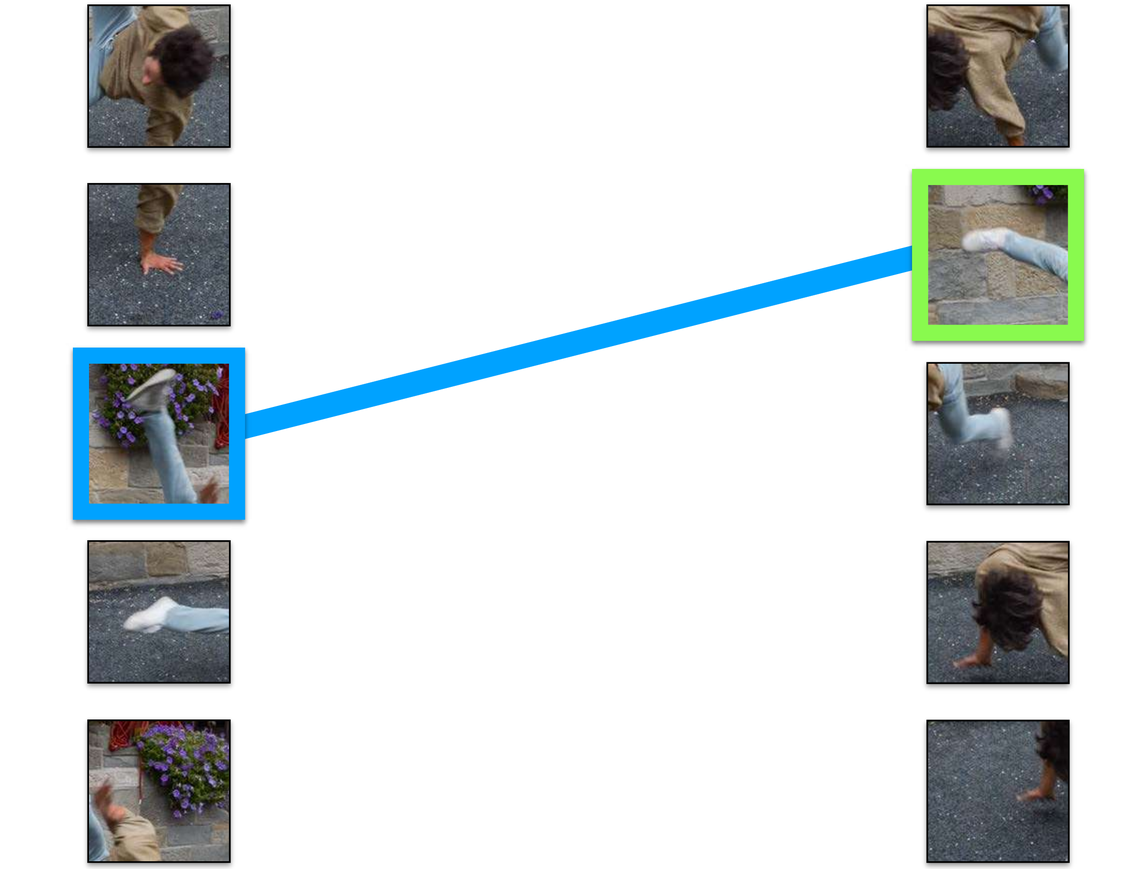
Learn a similarity representation for correspondence from unlabeled video

Focus:

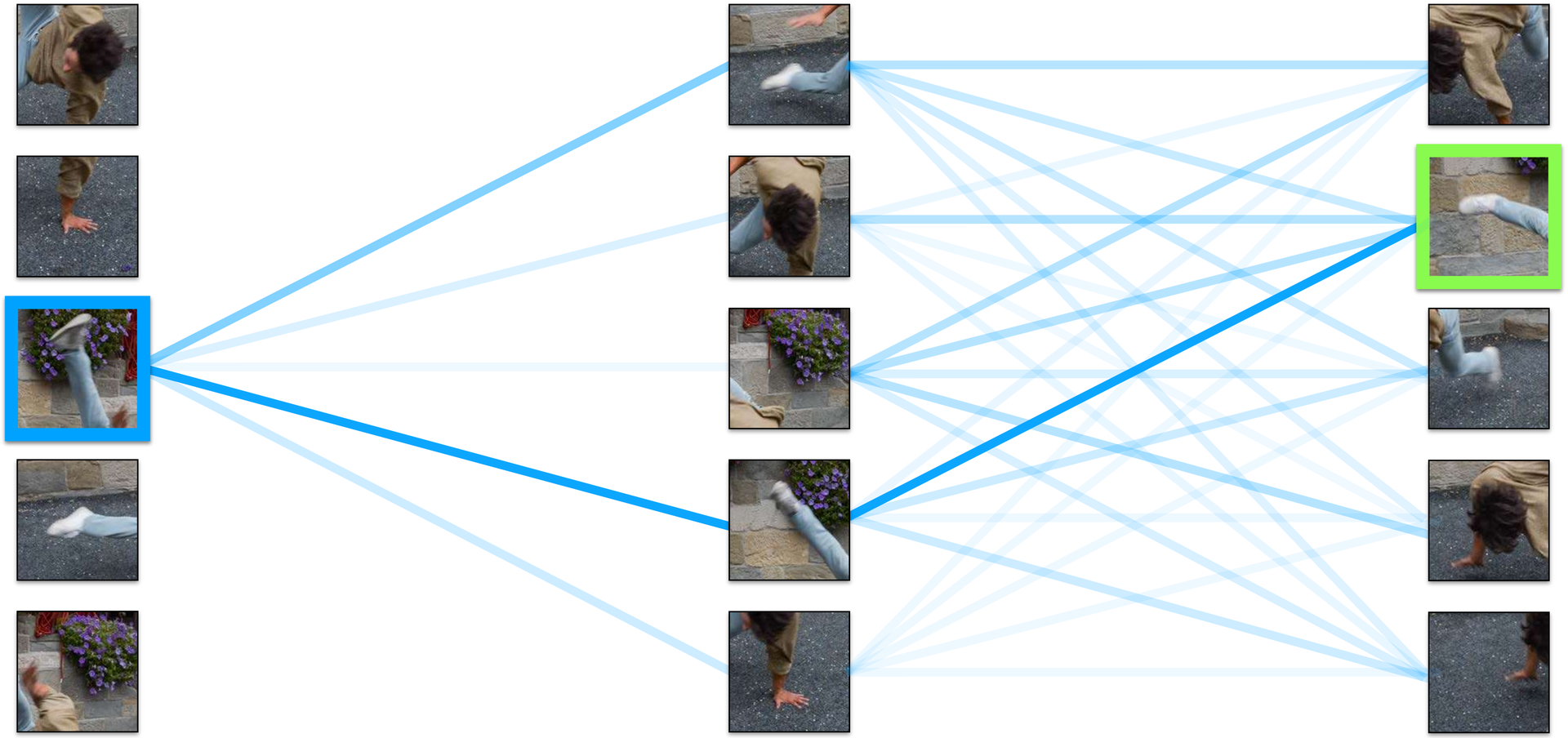
Simplicity and Scalability

Supervised correspondences

Supervised correspondences

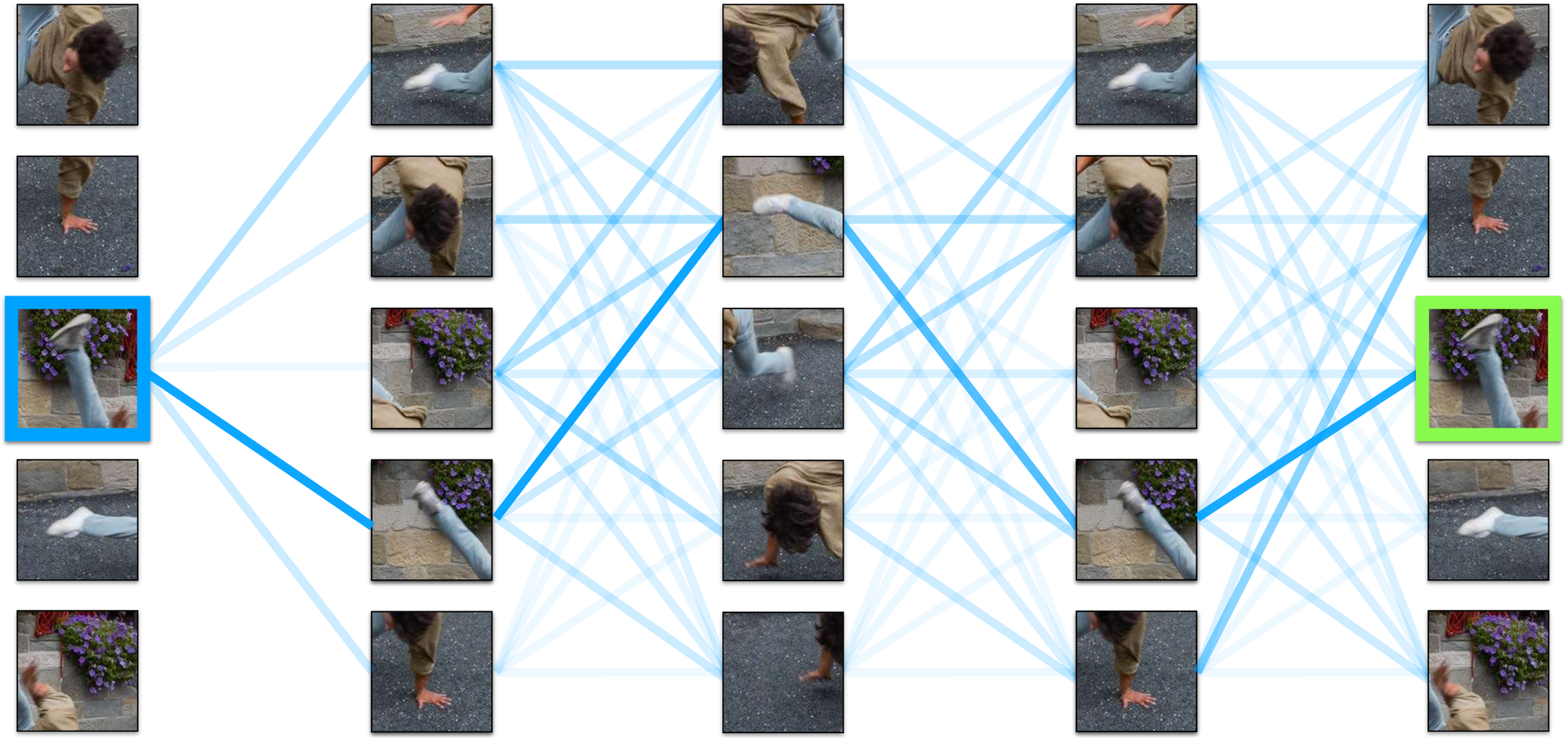


Latent correspondences

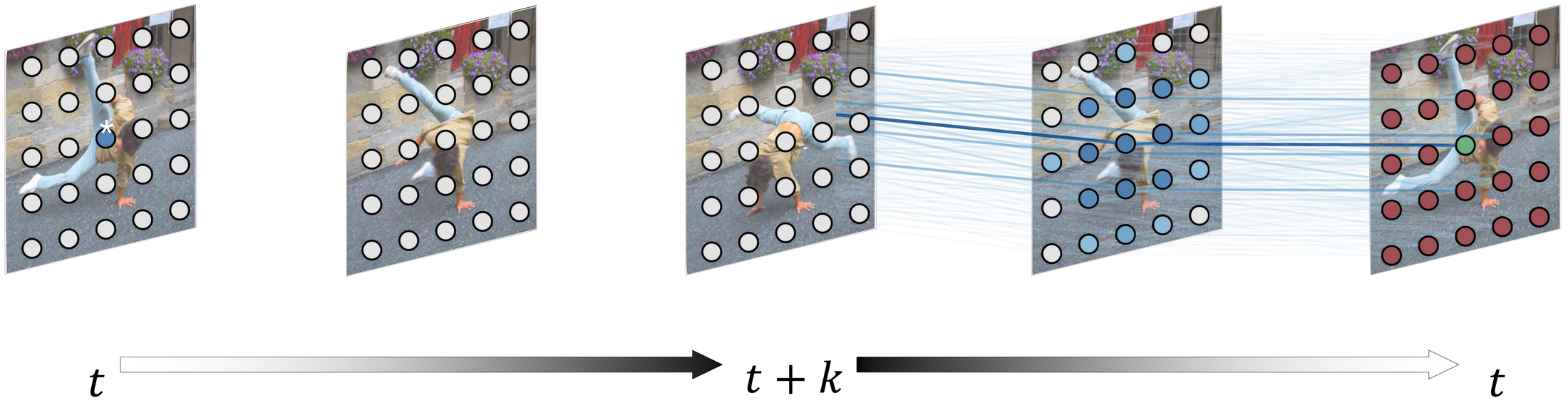


Palindrome sequence

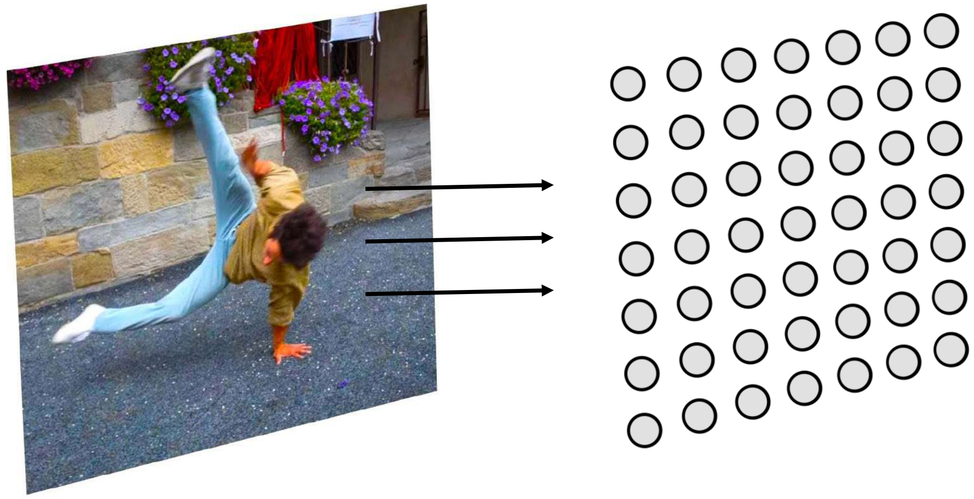
Self-supervised correspondences



Contrastive Random Walk



Video as a Graph



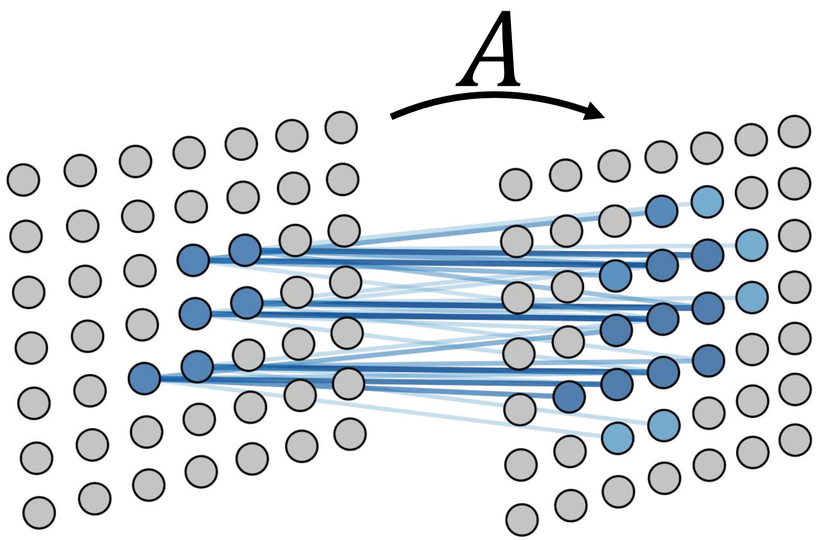
I_t

$\mathbf{q}_t$

Pixels

Nodes

Video as a Graph



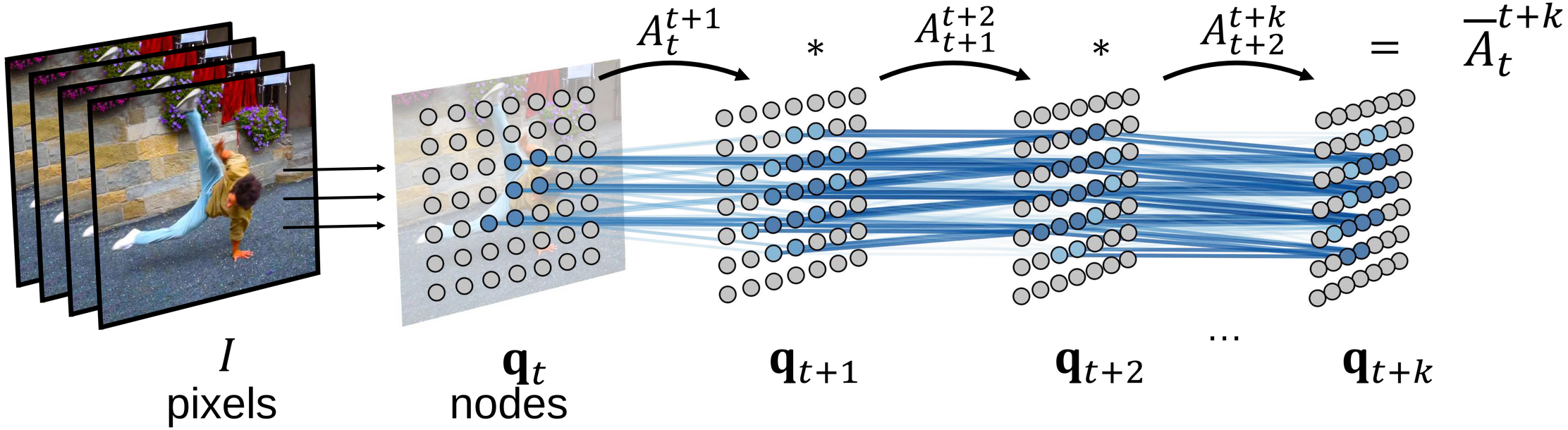
$$A_{ij} = \frac{e^{d_\phi(q_t^i,\, q_{t+1}^j)/\tau}}{\sum_l e^{d_\phi(q_t^i,\, q_{t+1}^l)/\tau}}$$

$$= P(X_{t+1} = j | X_t = i)$$

where $d_\phi(x, y) = \phi(x)^\top \phi(y)$

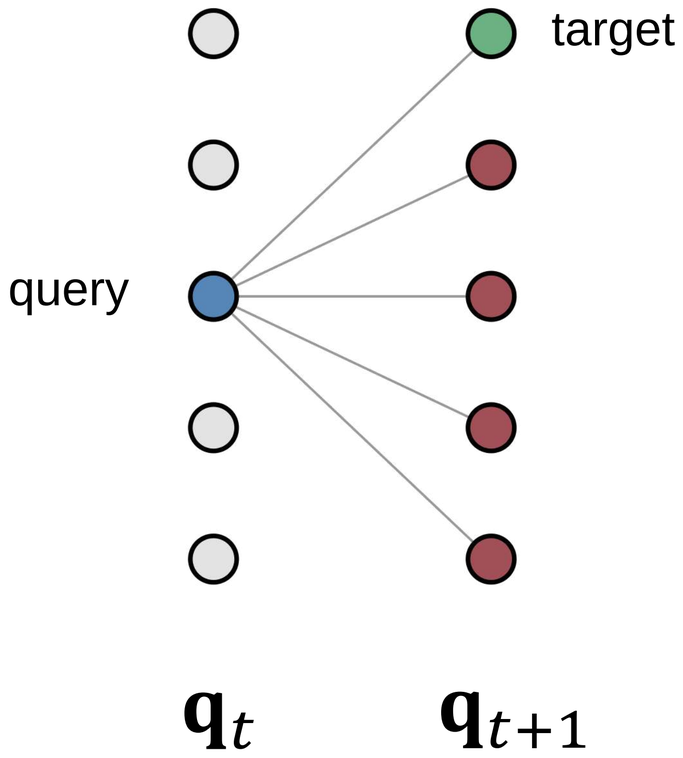
X_t is the position of walker at time t

Correspondence as a Random Walk



Learn embedding ϕ = Fitting transition probabilities

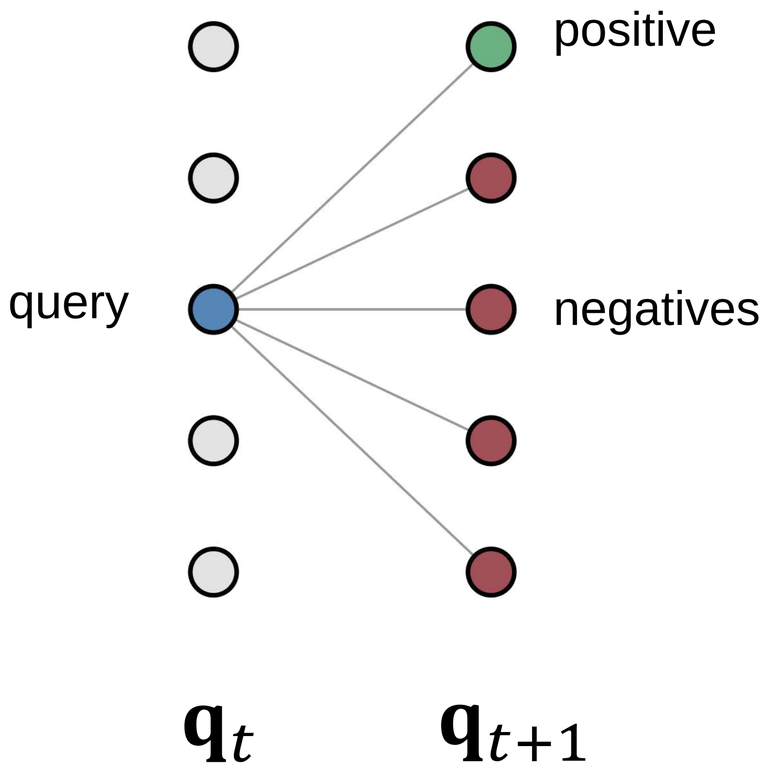
One Step Contrastive Learning



Maximize

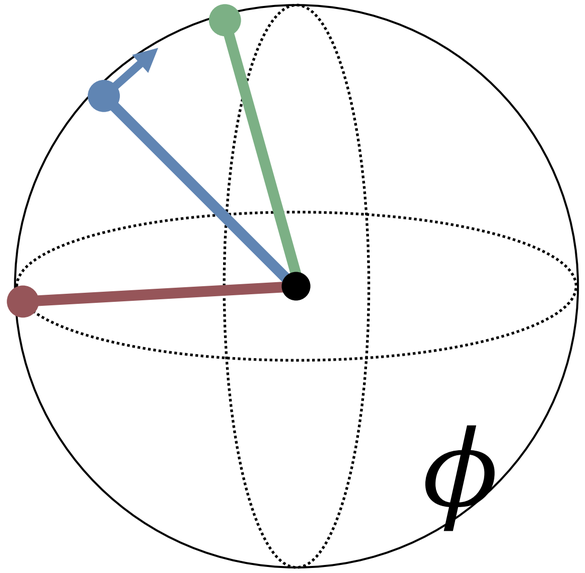
$$P(X_{t+1} = target | X_t = query)$$

One Step Contrastive Learning



Maximize

$$P(X_{t+1} = pos | X_t = query) = \frac{e^{\phi(query)^\top \phi(pos)}}{\sum_l e^{\phi(query)^\top \phi(neg_l)}}$$



l_2-normalized Embedding

Contrastive Learning

with cross entropy loss

Dosovitsky et al (2014)

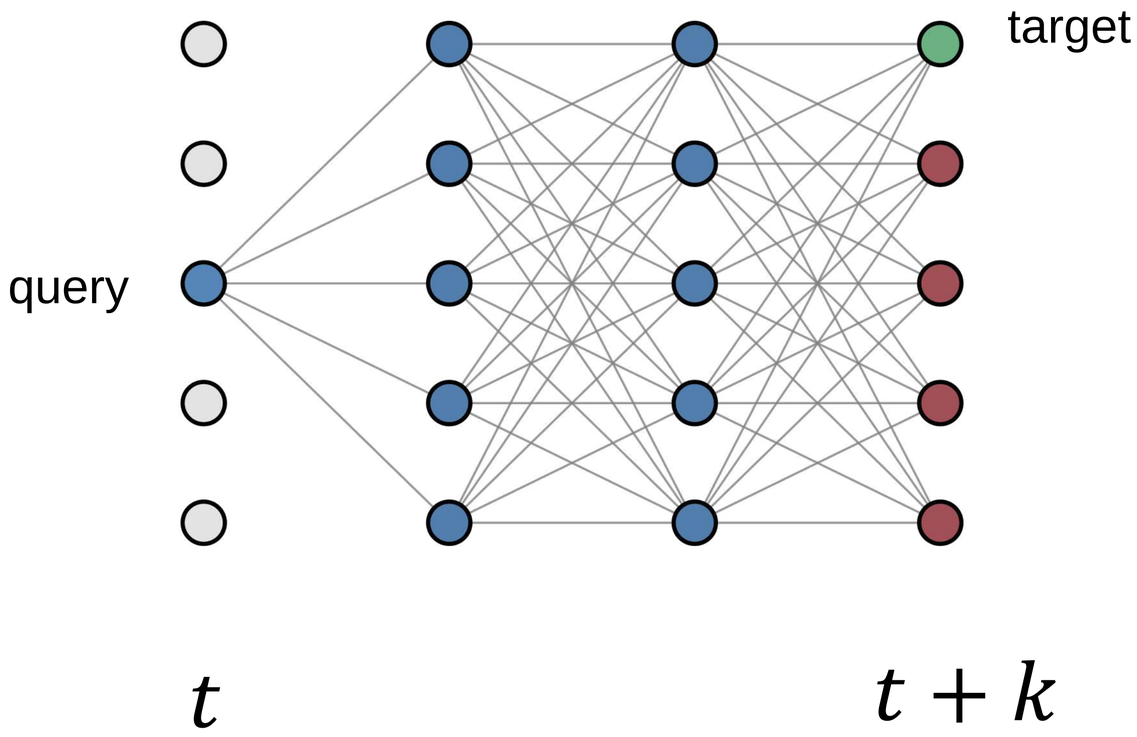
Isola et al (2015)

Tian et al (2019)

Wu et al (2018)

van den Oord et al (2019)

Chaining Correspondences



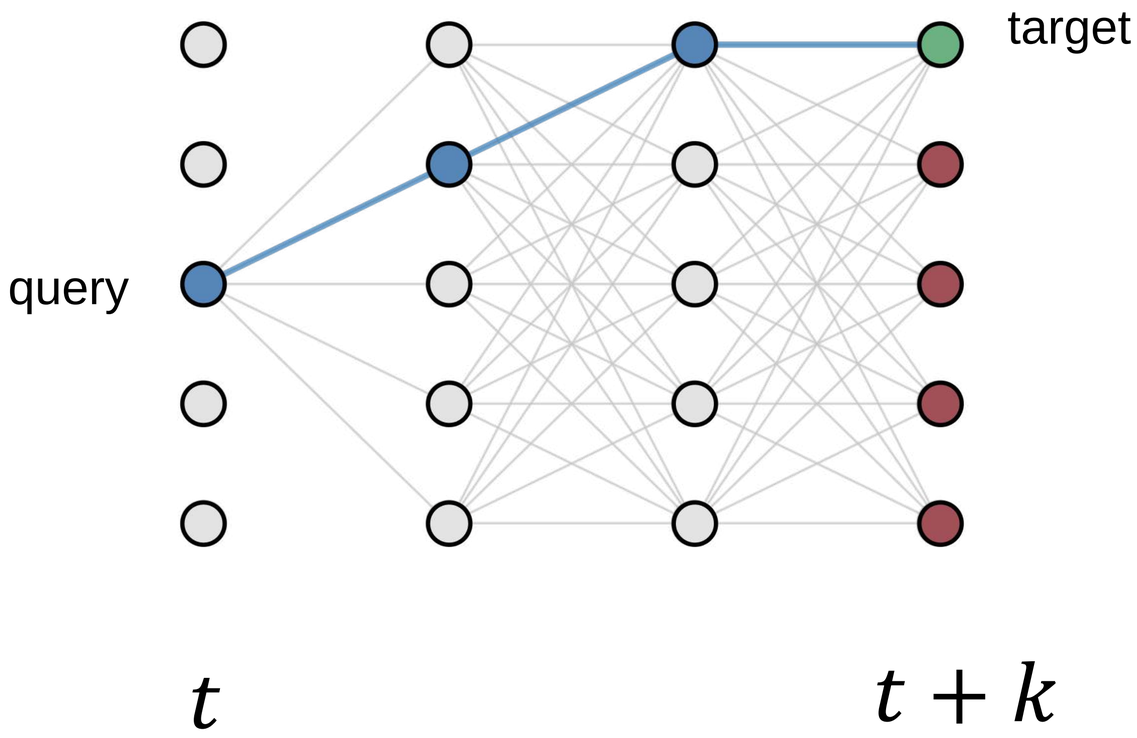
k-step Transition Matrix

$$\overline{A}_t^{t+k} = \prod_{i=0}^{k-1} A_{t+i}^{t+i+1}$$

← Sums over intermediate time steps

$$= P(X_{t+k} | X_t)$$

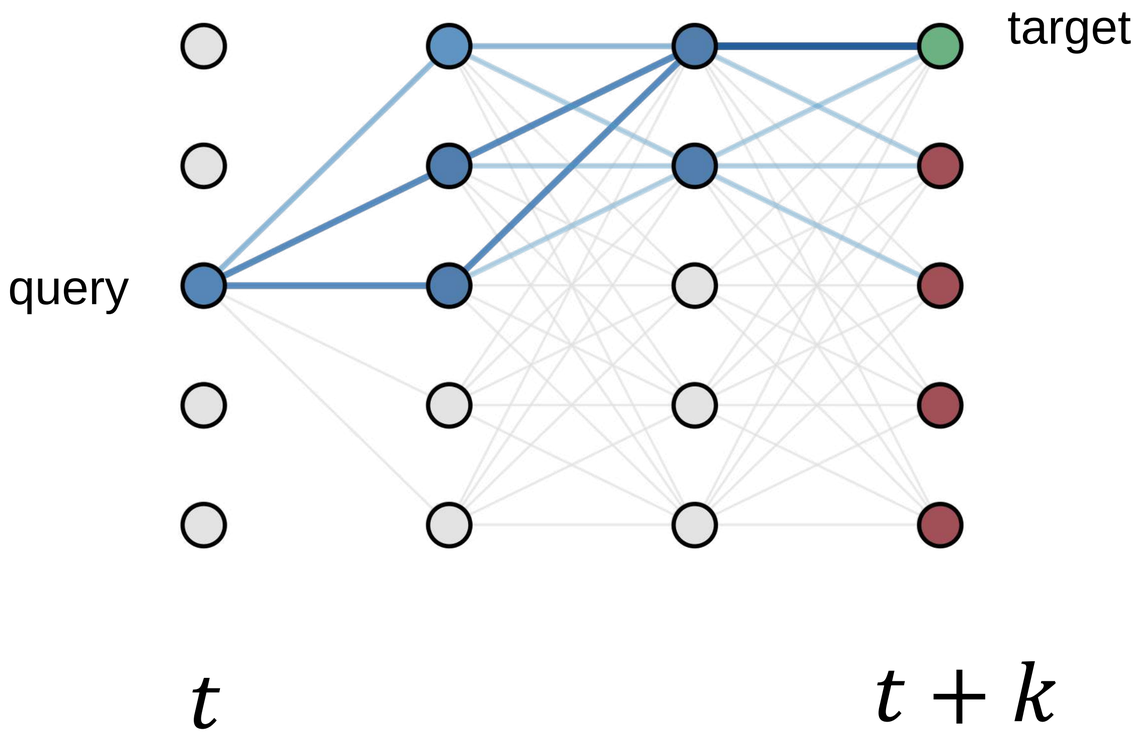
Chaining Correspondences



Easy case:

Correspondence is obvious

Chaining Correspondences

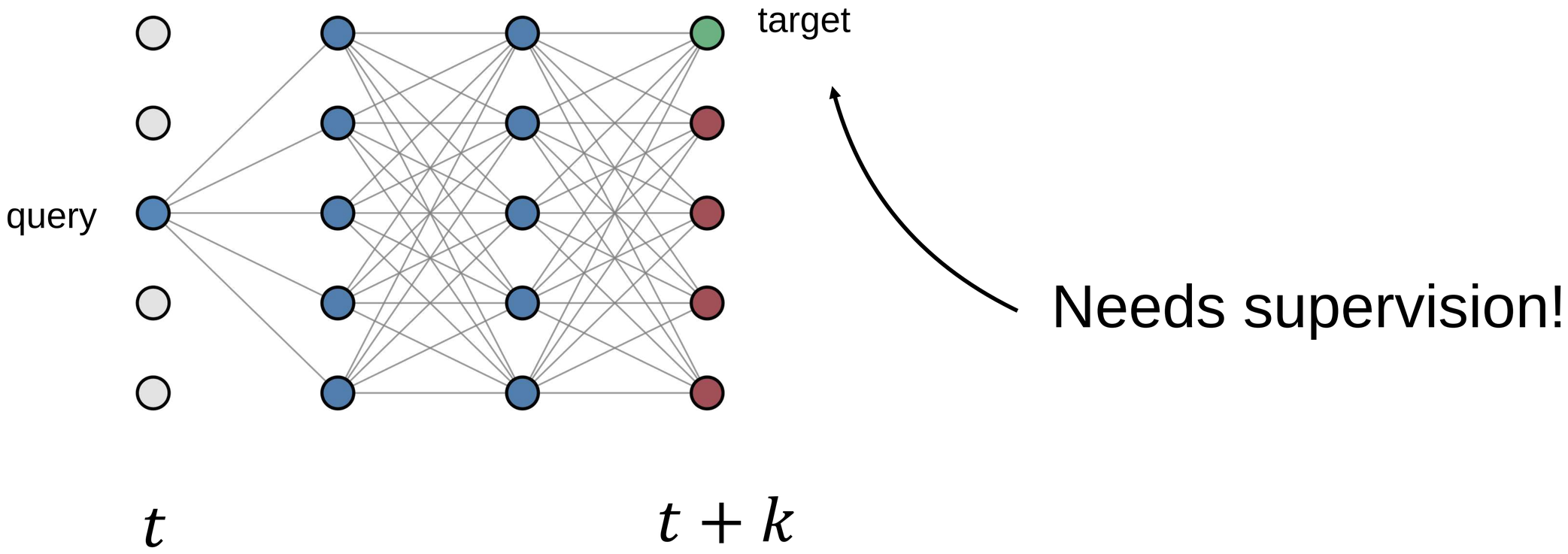


Harder case:

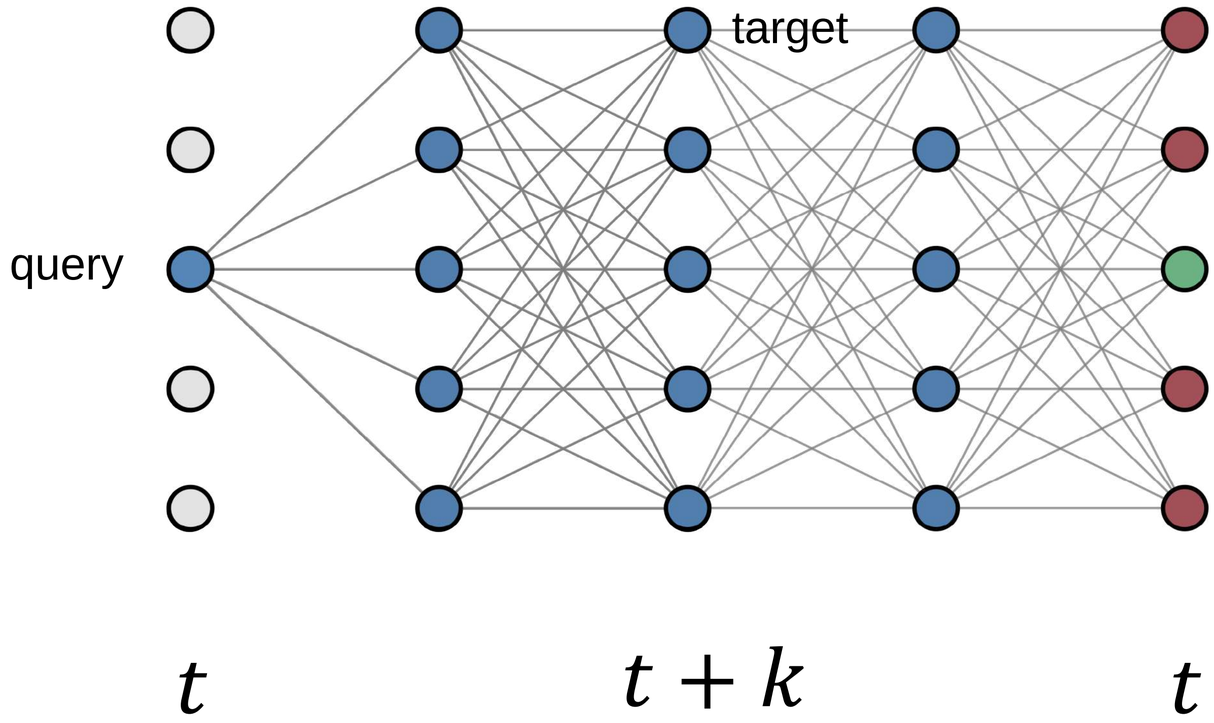
Ambiguity

Considers multiple paths

Chaining Correspondences



Supervised → Self-Supervised



Train on Palindromes

$$\mathcal{L}_{cyc}^{k} = \mathcal{L}_{CE}(\bar{A}_{t}^{t+k}\bar{A}_{t+k}^{t}, I)$$

Contrastive learning
with latent correspondences

Simple

Algorithm 1 Pseudocode in a PyTorch-like style.

Embed nodes

Construct graph

Walk on graph

```
# load a minibatch x with B sequences
for x in loader:
  # B x C x T x H x W -> B x C x T x P x h x w
  x = unfold(x, (patch_size, patch_size))
  x = spatial_jitter(x)
  v = l2_norm(resnet(x))  # Embed patches (B x C x T x P)

  # Transitions from t to t+1 (B x T x P x P)
  A = einsum("bcti,bctj->bcij", v[:,:,:-1], v[:,:,1:])
  A /= temperature

  # Transition similarities for palindrome graph
  AA = cat((A, A[:,::-1].transpose(-1,-2), 1)
  At = eye(P)

  # Perform random walk
  for t in range(2*T):
   At = bmm(softmax(dropedge(AA[:,t]),dim=-1), At)

  # Target is the original node
  loss = cross_ent_loss(At, labels=[range(P)]*B)
  loss.backward()
```

Evaluation

Use ϕ for
Label Propagation

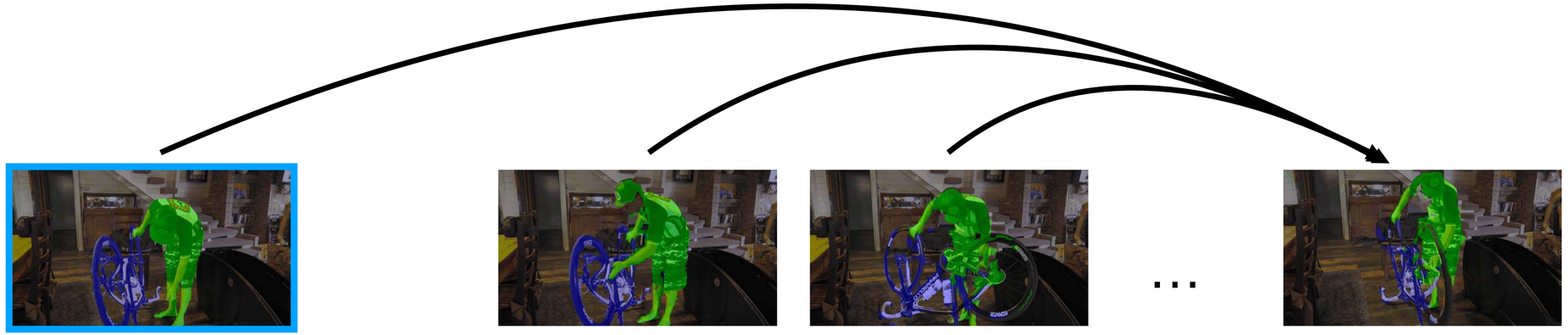
Semantic Part Propagation 20 Parts VIP Benchmark

Pose Propagation 15 Keypoints JHMDB Benchmark

Object Propagation 1-4 Objects DAVIS Benchmark

Label Propagation

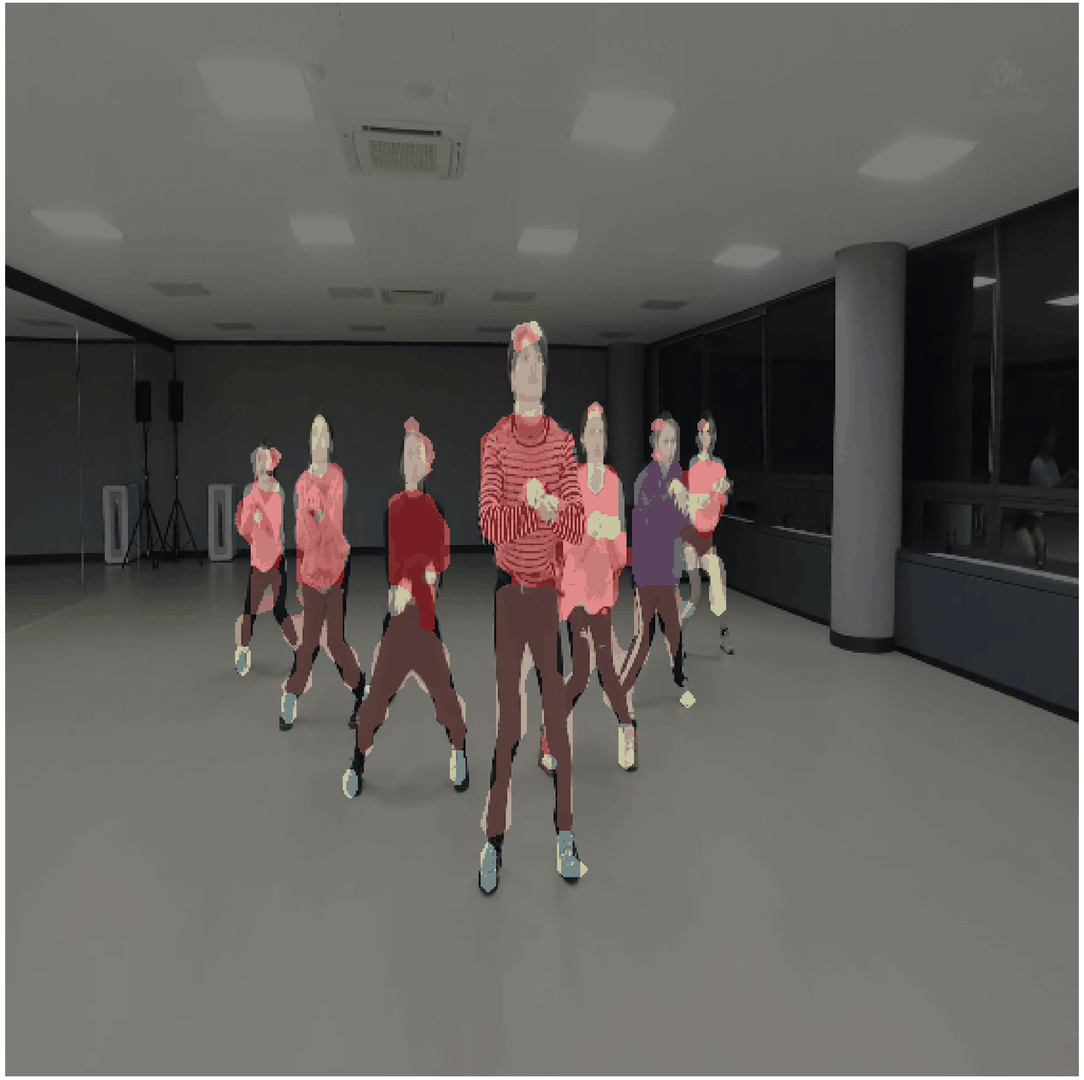
Wang et al. (2019)
Li et al. (2019)
Lai et al. (2019



Using more context, i.e. last m frames

Semantic Part Propagation

on Video Instance Parsing (VIP) dataset




BE
WATER
Combat Tai Chi DVD
Order Now @
www.JakeMace.com

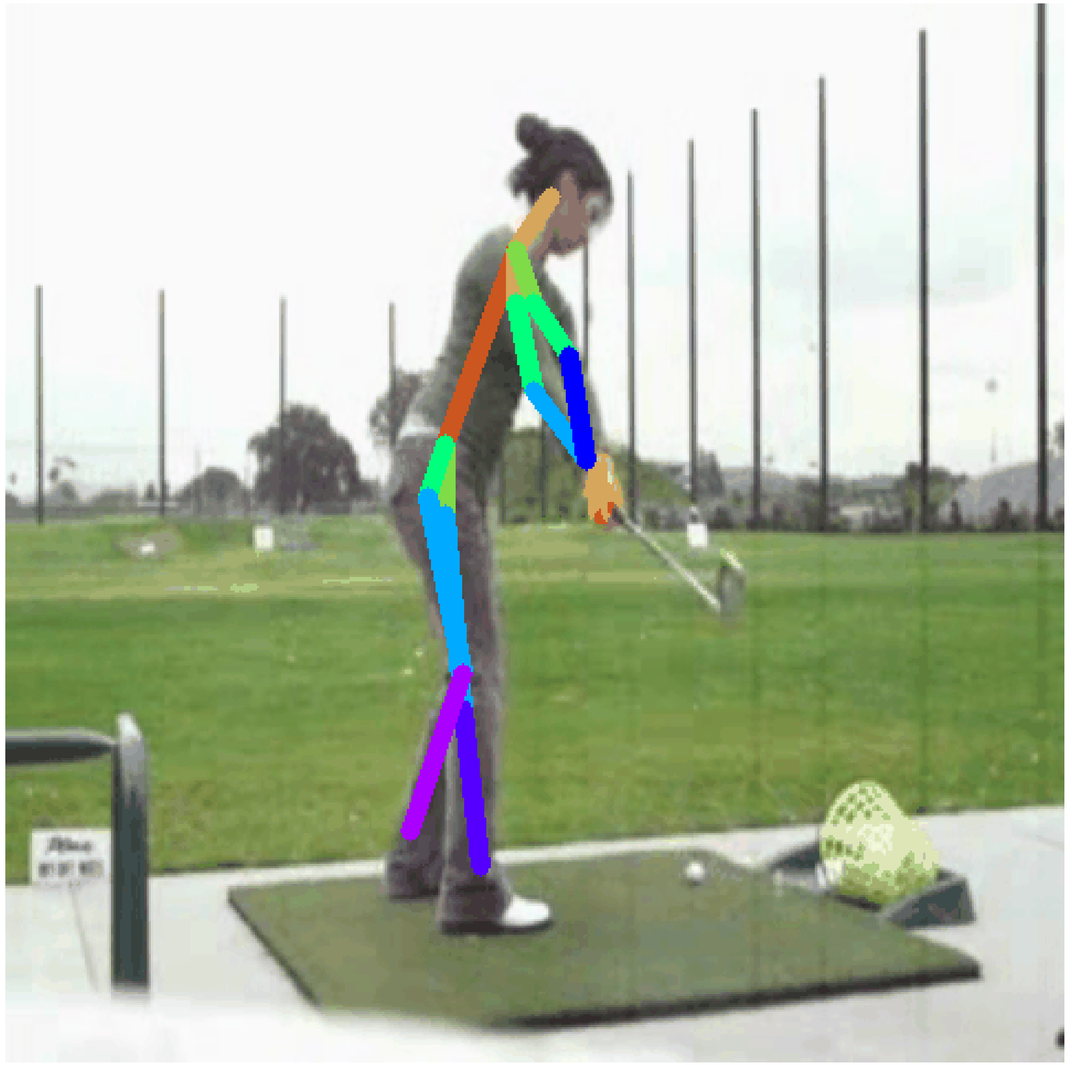

其實
abc

Pose Tracking

on JHMDB

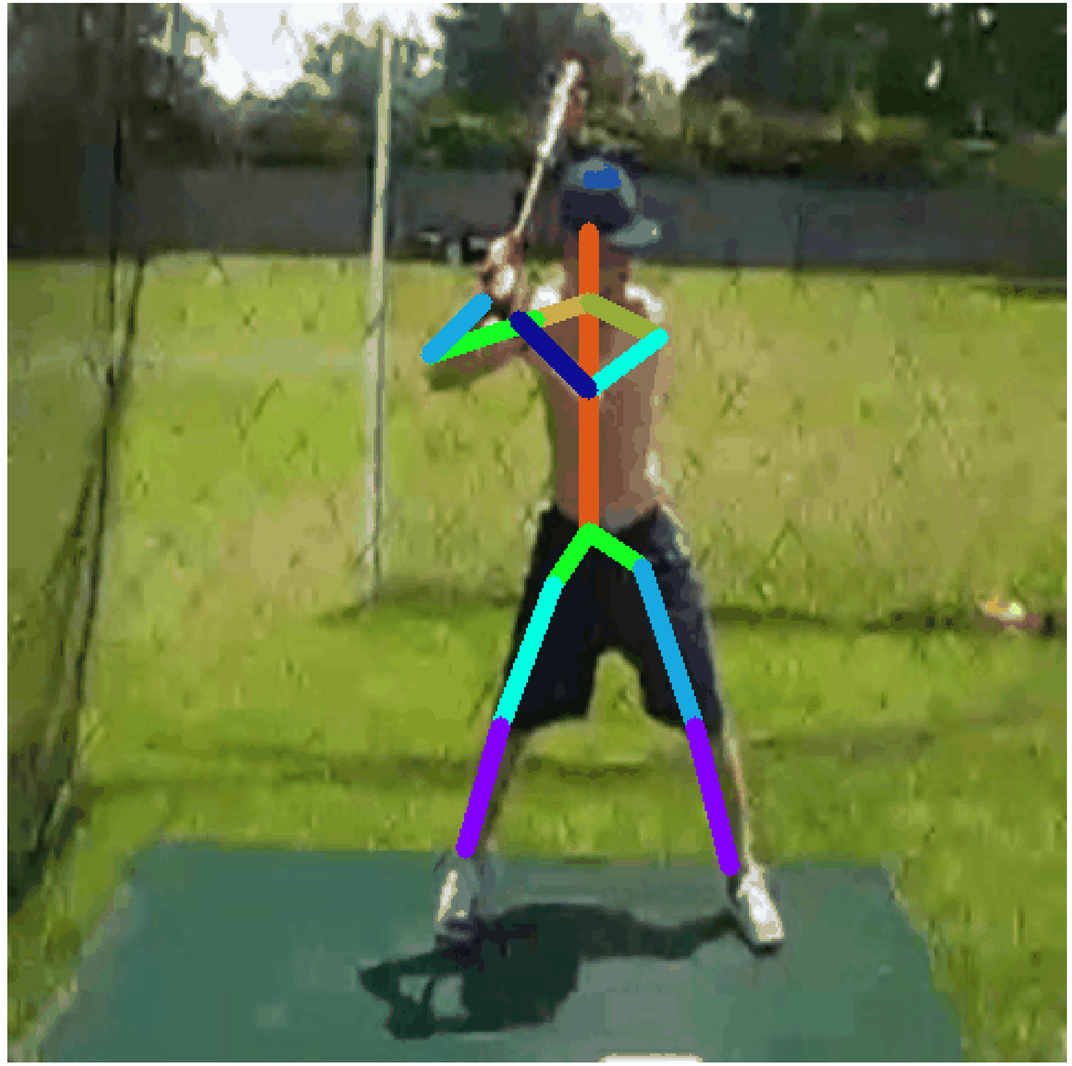
UVC [Li et al. 2019]

Ours



UVC [Li et al. 2019]

Ours

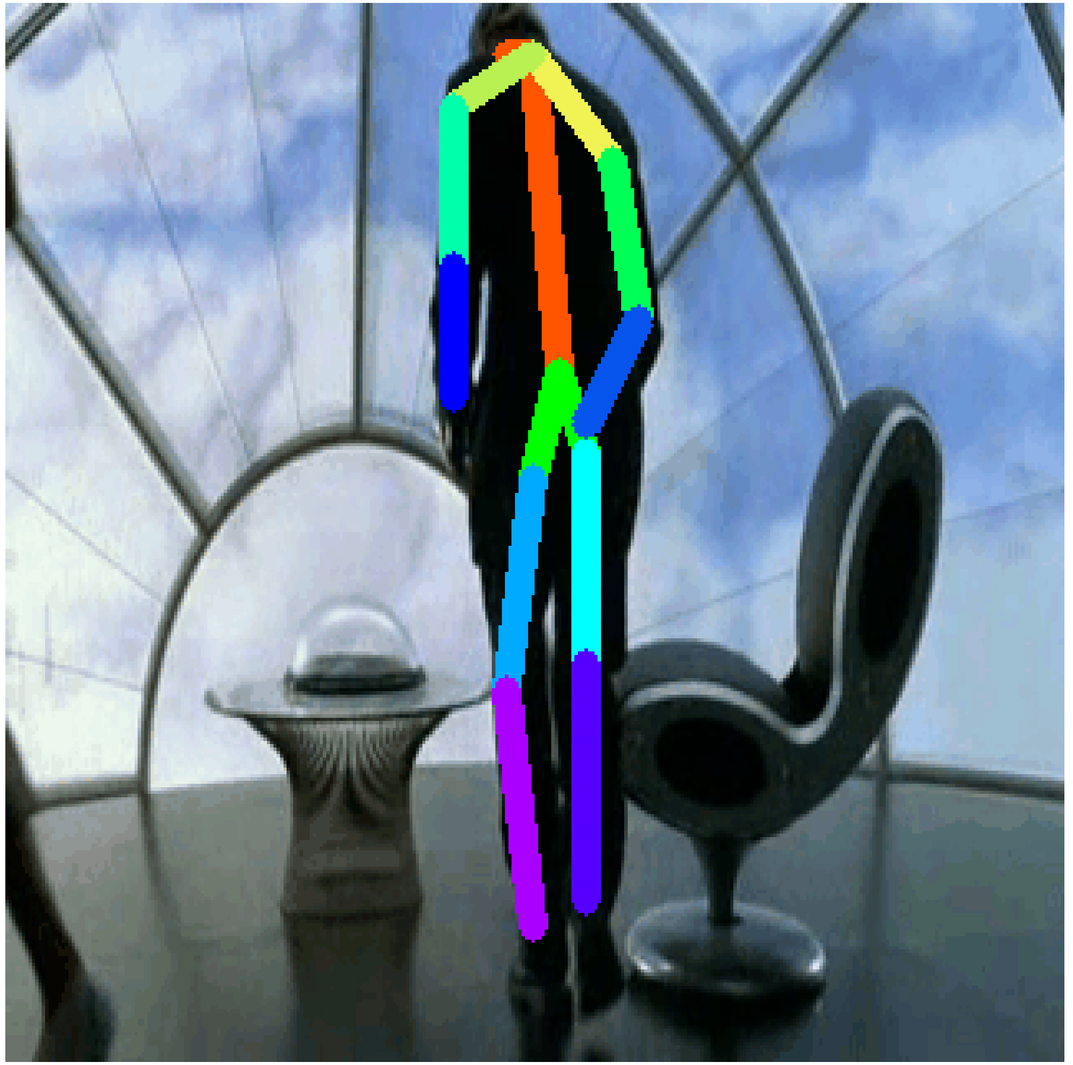


UVC [Li et al. 2019]

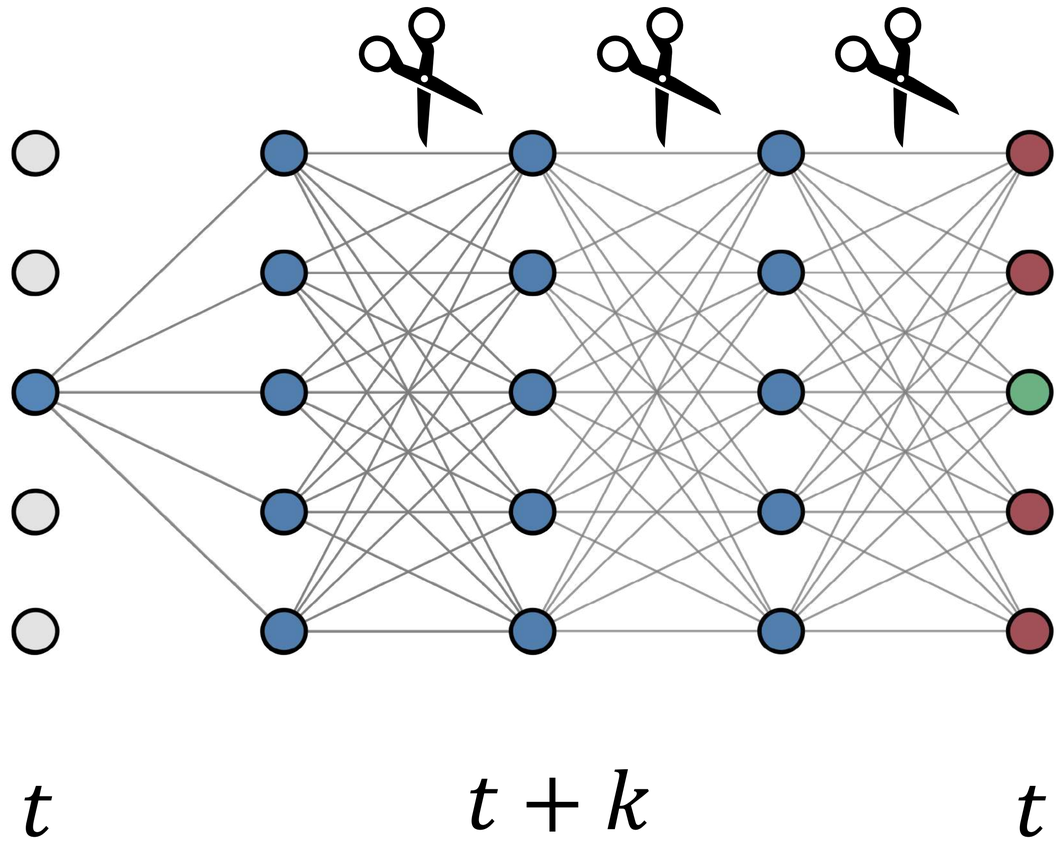
Ours

UVC [Li et al. 2019]

Ours



“Common Fate” by Edge Dropout

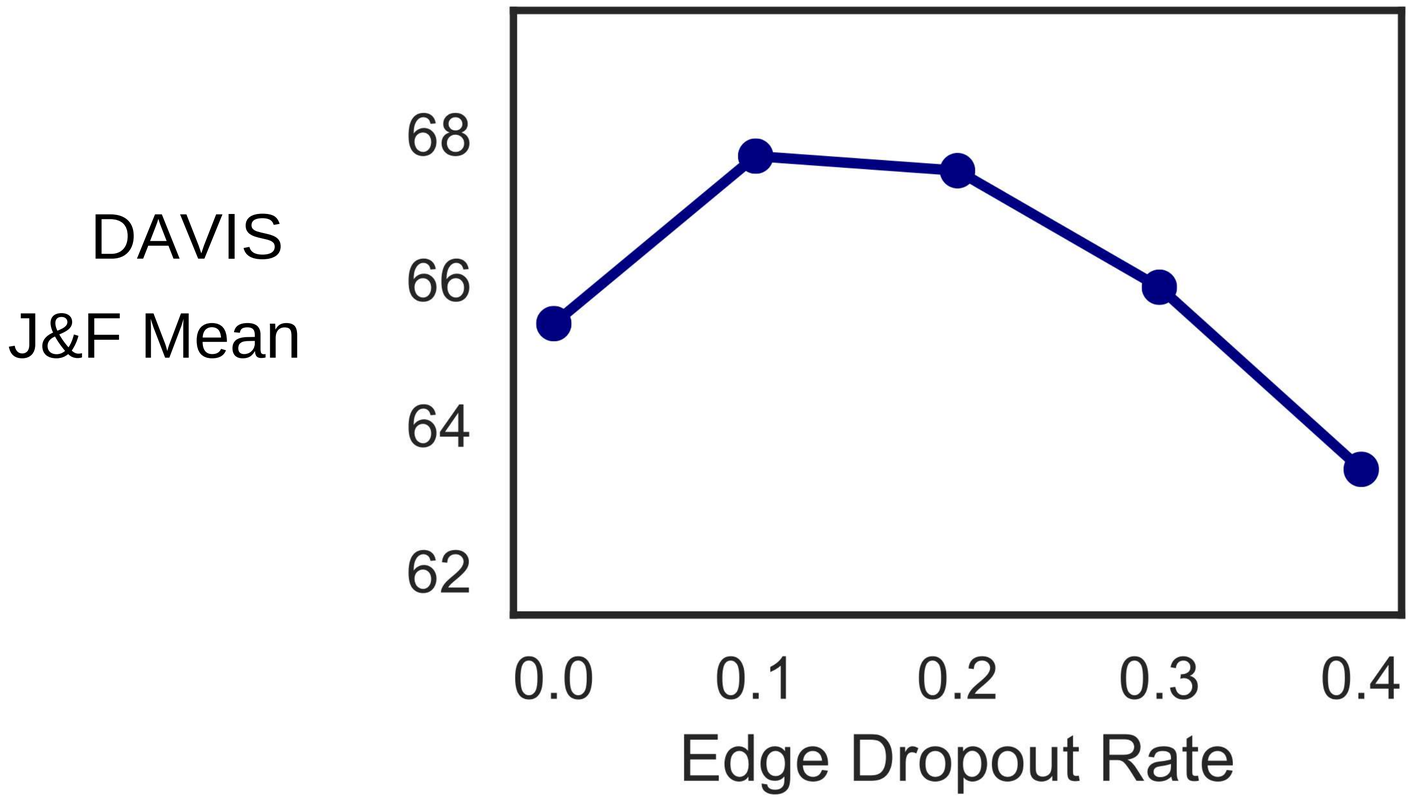


Higher-level correspondence

by modeling correlated paths

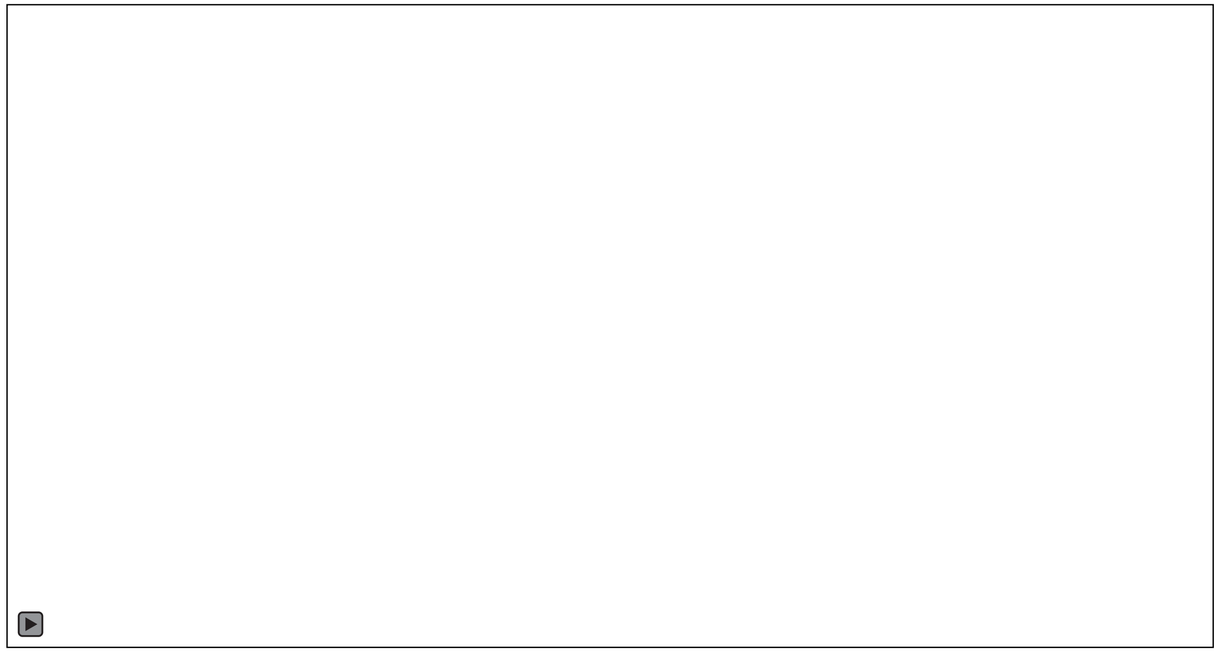
Laws of organization in perceptual forms.
Wehrtheimer (1938)

Edge Dropout

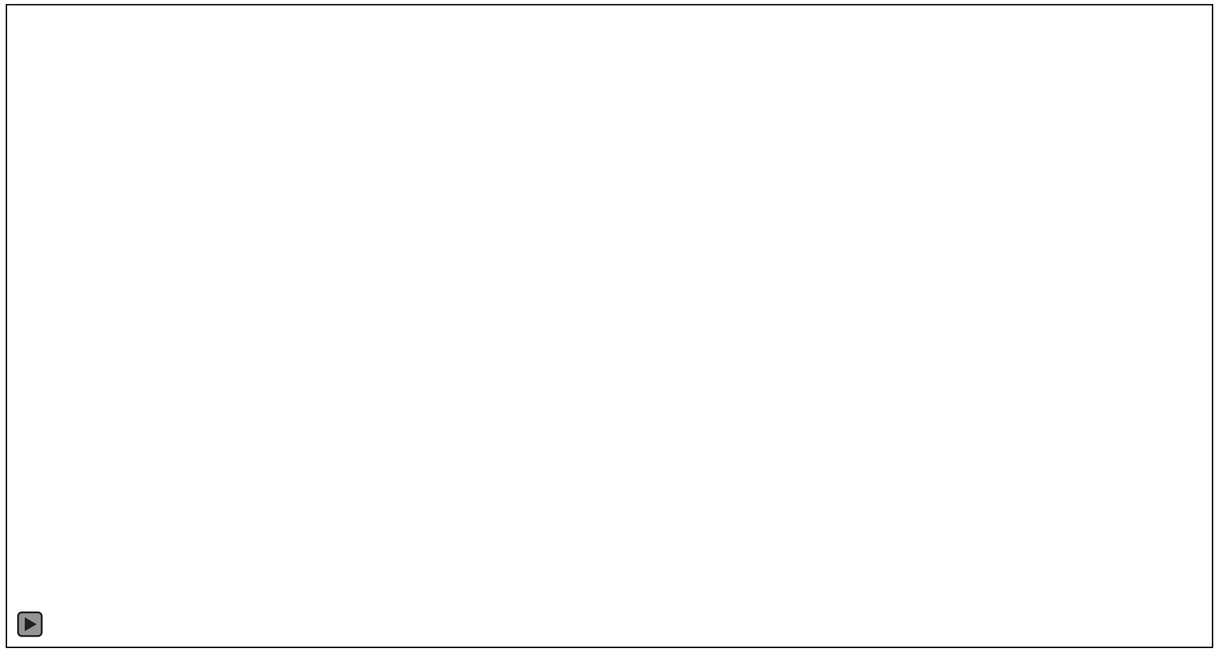


Video Object Propagation

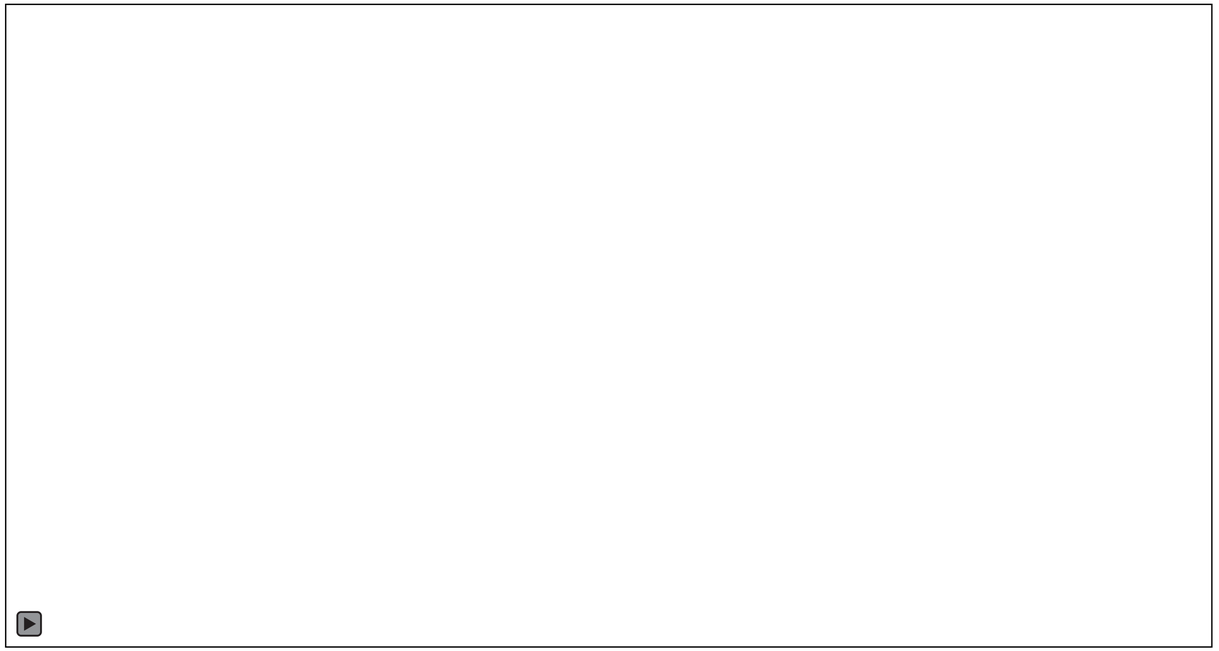
on DAVIS dataset



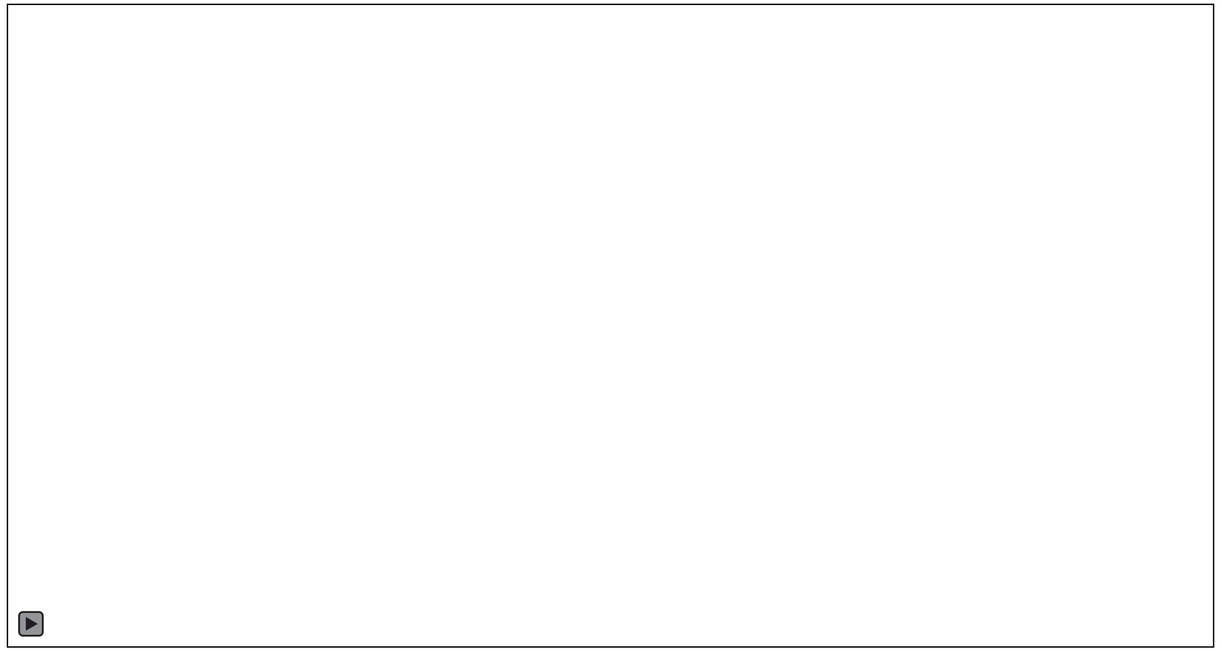
UVC [Li et al., 2019]



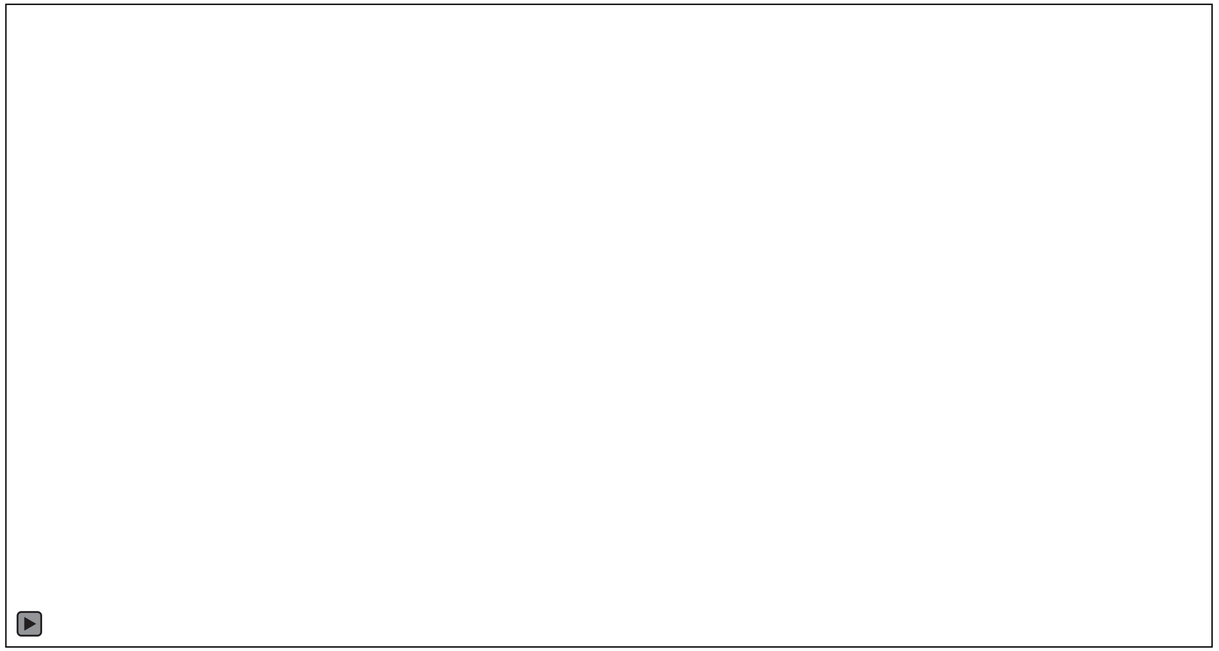
Ours



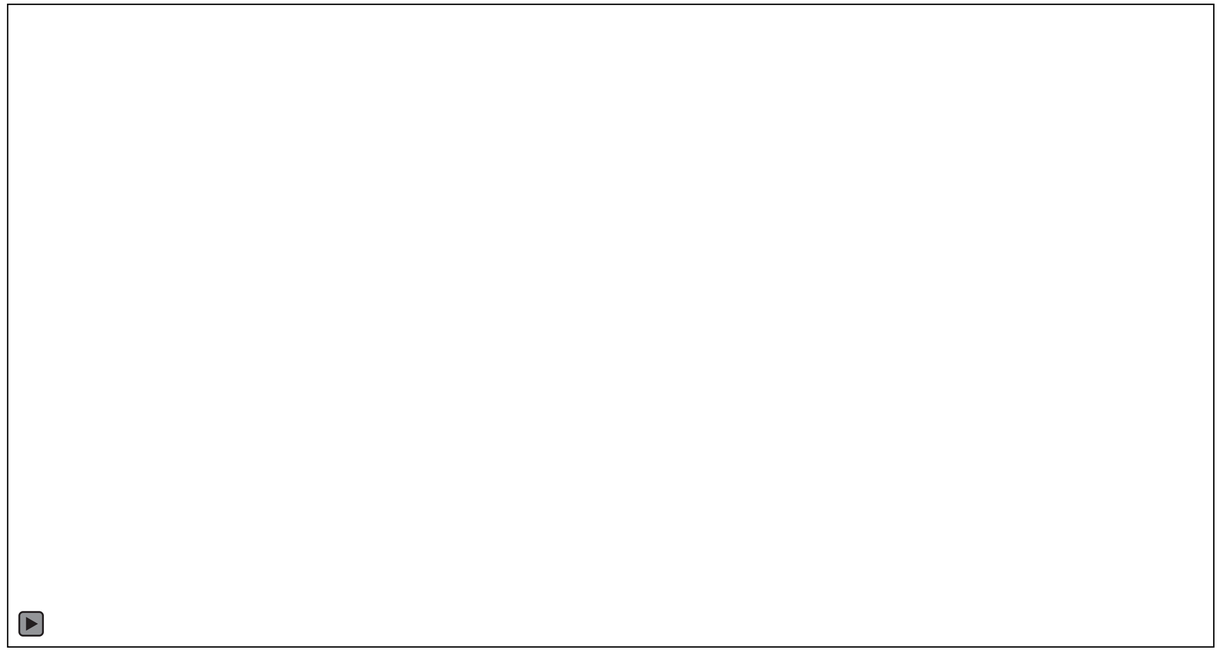
UVC [Li et al., 2019]



Ours



UVC [Li et al., 2019]

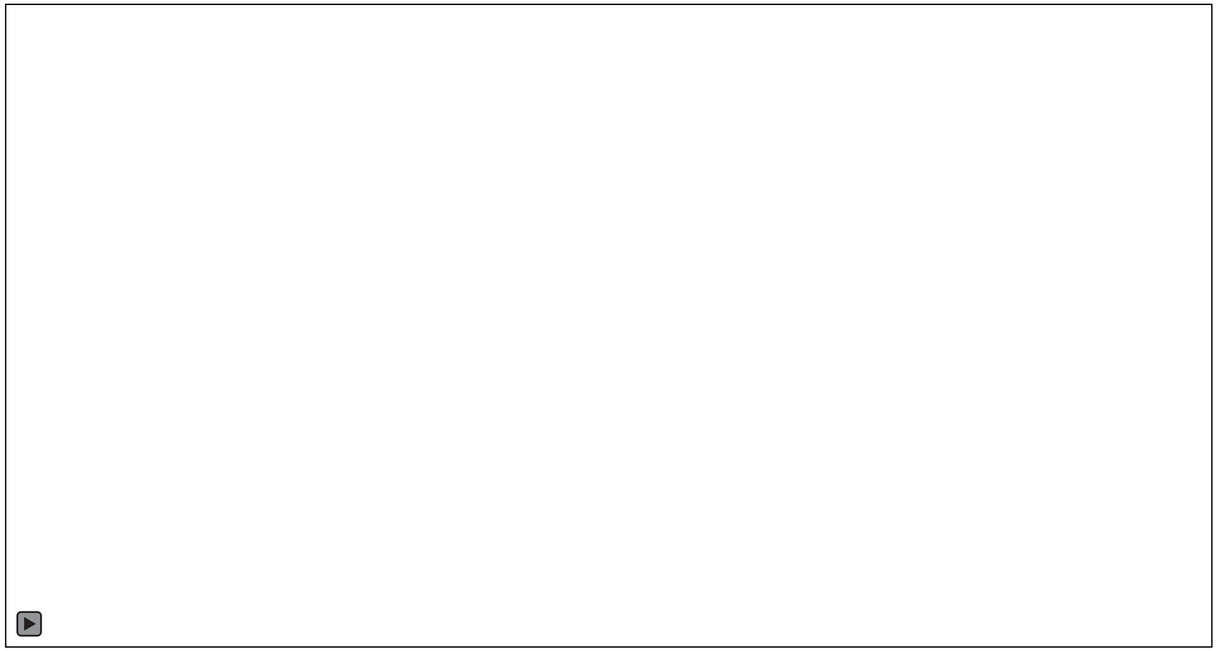


Ours

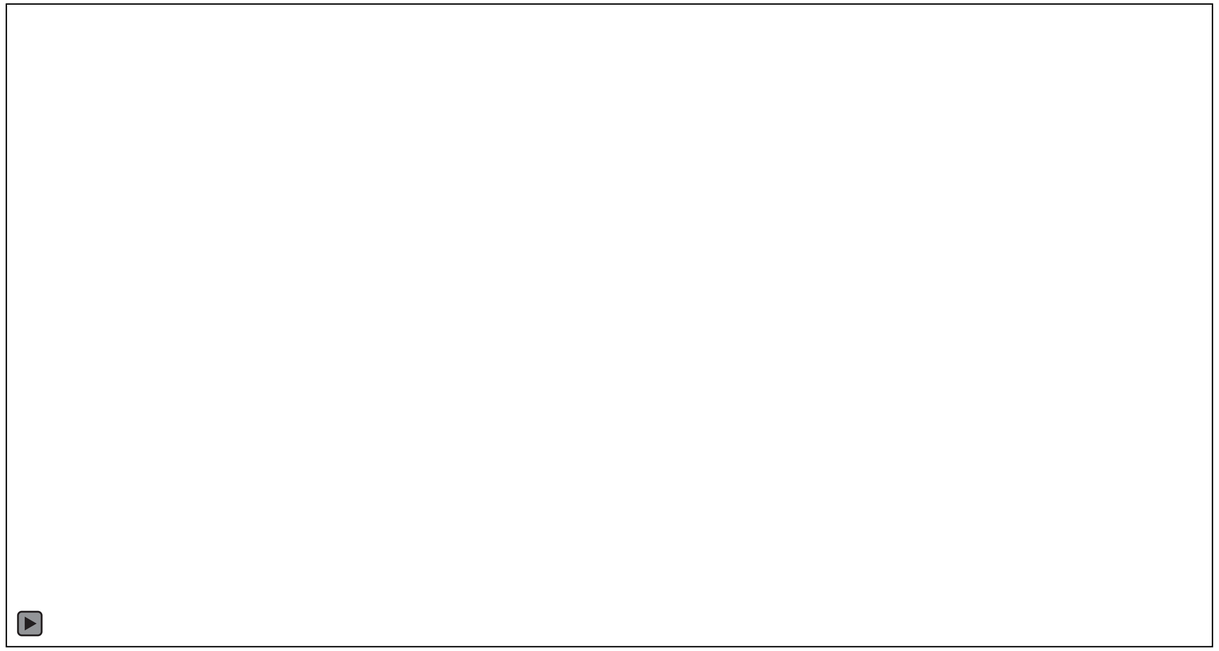
Still Frame Comparison

UVC [Li et al., 2019]

Ours

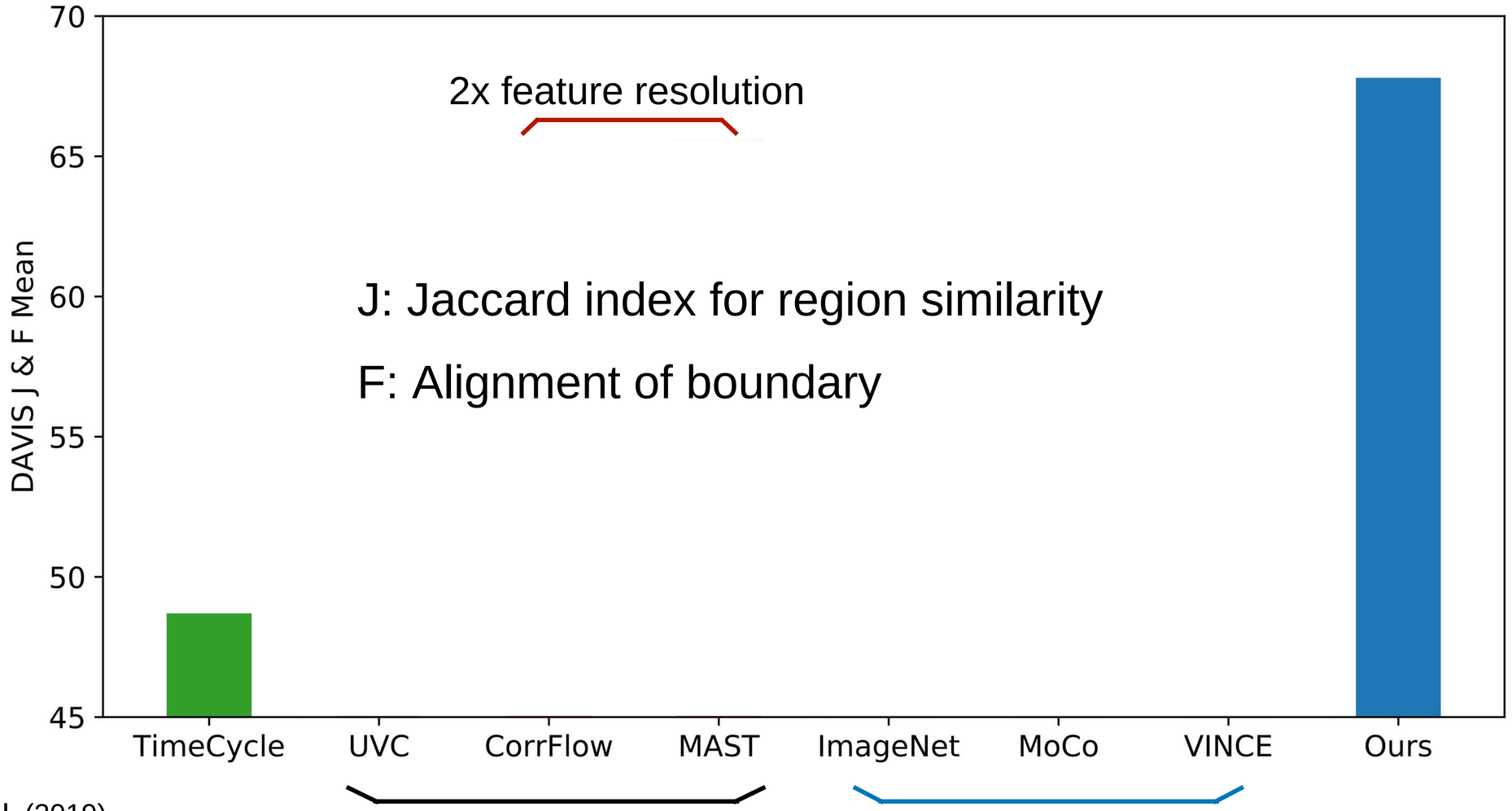


UVC [Li et al., 2019]



Ours

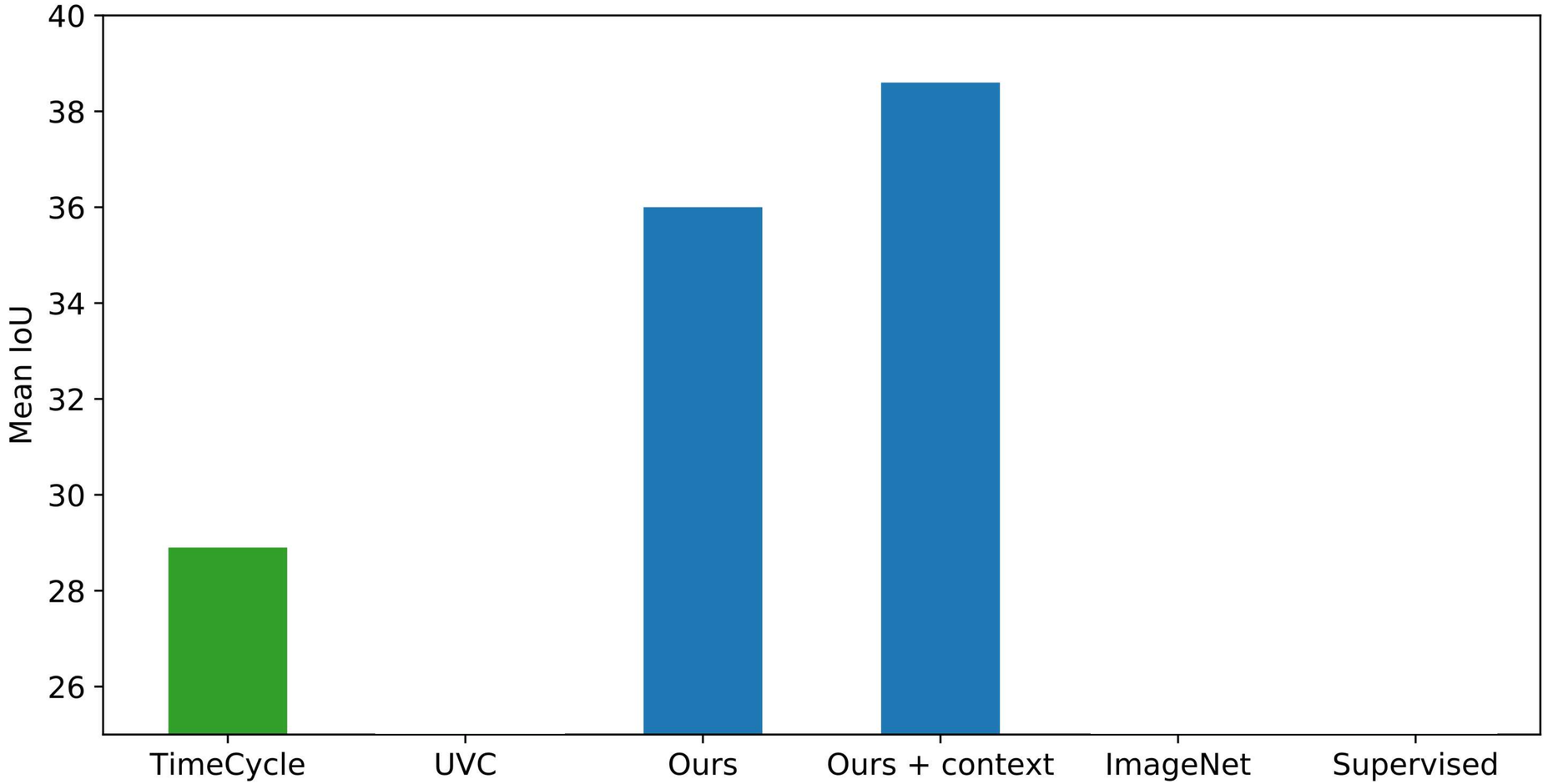
Quantitative Comparison: Self-Supervised State-of-the-Art



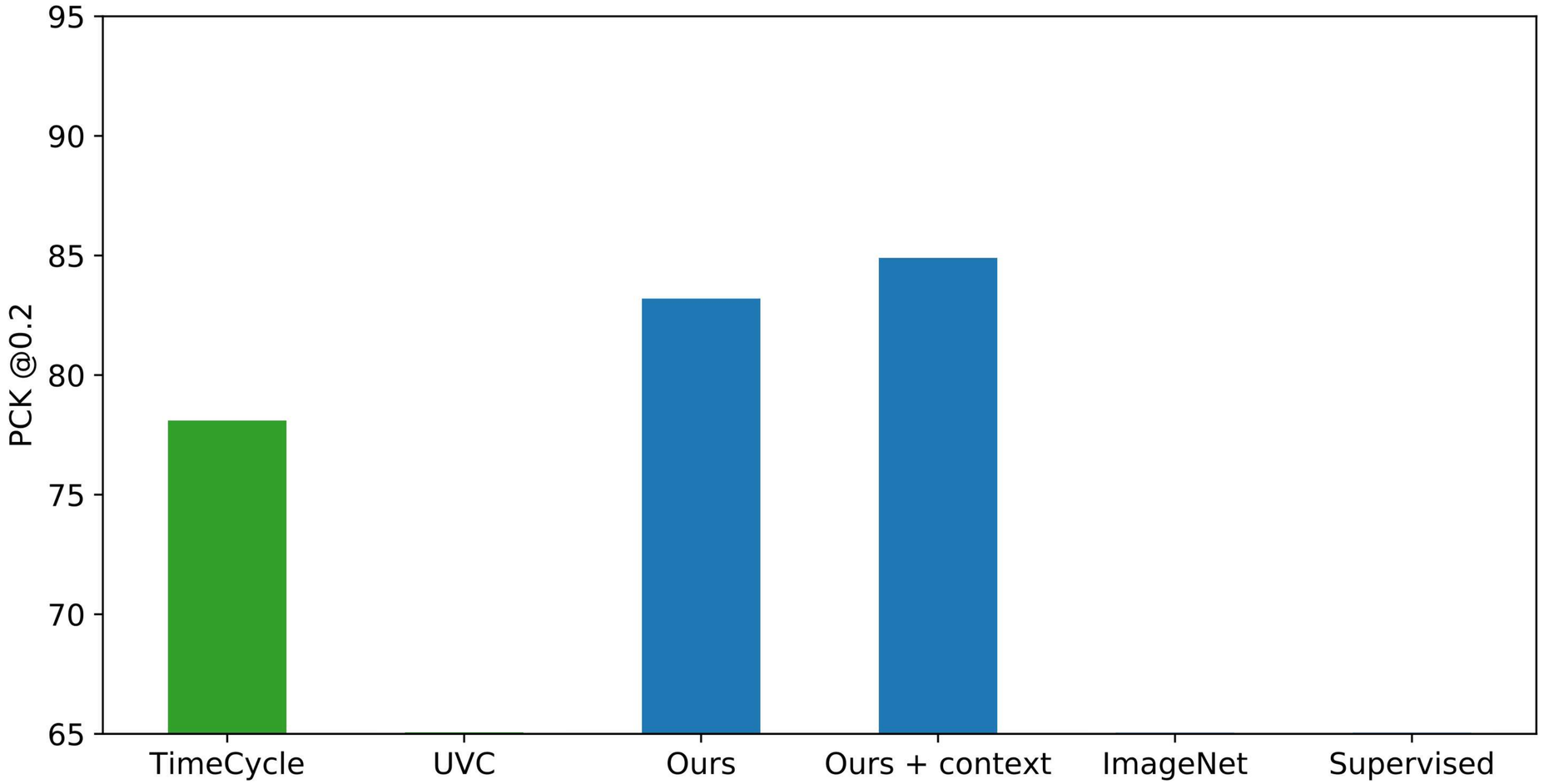
TimeCycle: Wang et al. (2019)
UVC: Li et al. (2019)
CorrFlow: Lai et al. (2019)
MAST: Lai et al. (2019)

VINCE: Gordon et al. (2020)
MoCo: He et al. (2019)

Quantitative Comparison: Semantic Part Propagation



Quantitative Comparison: Pose Propagation



So far…

Pre-Train → Test on Example X

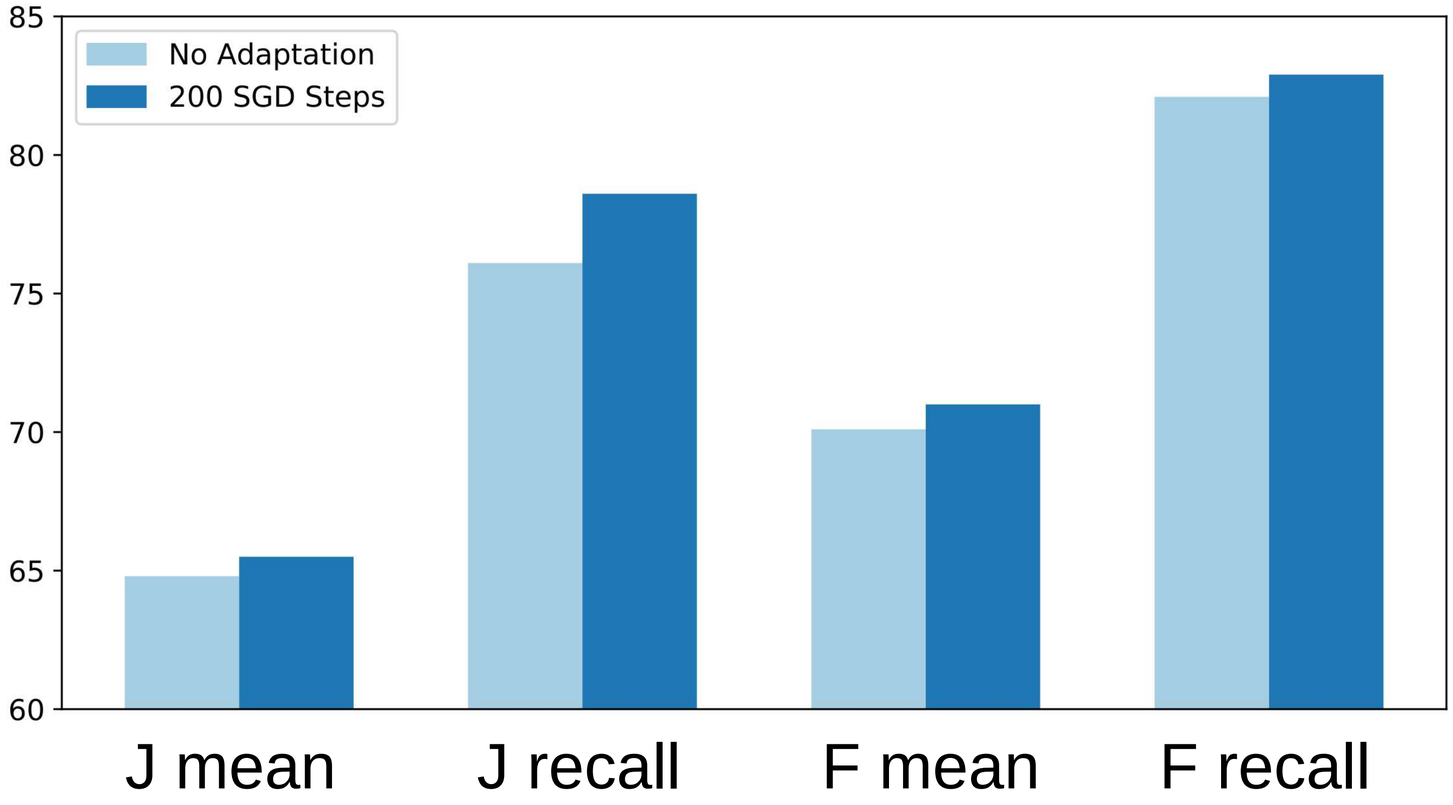
(No Adaptation)

Pre-Train → Self-sup Train on X → Test on X

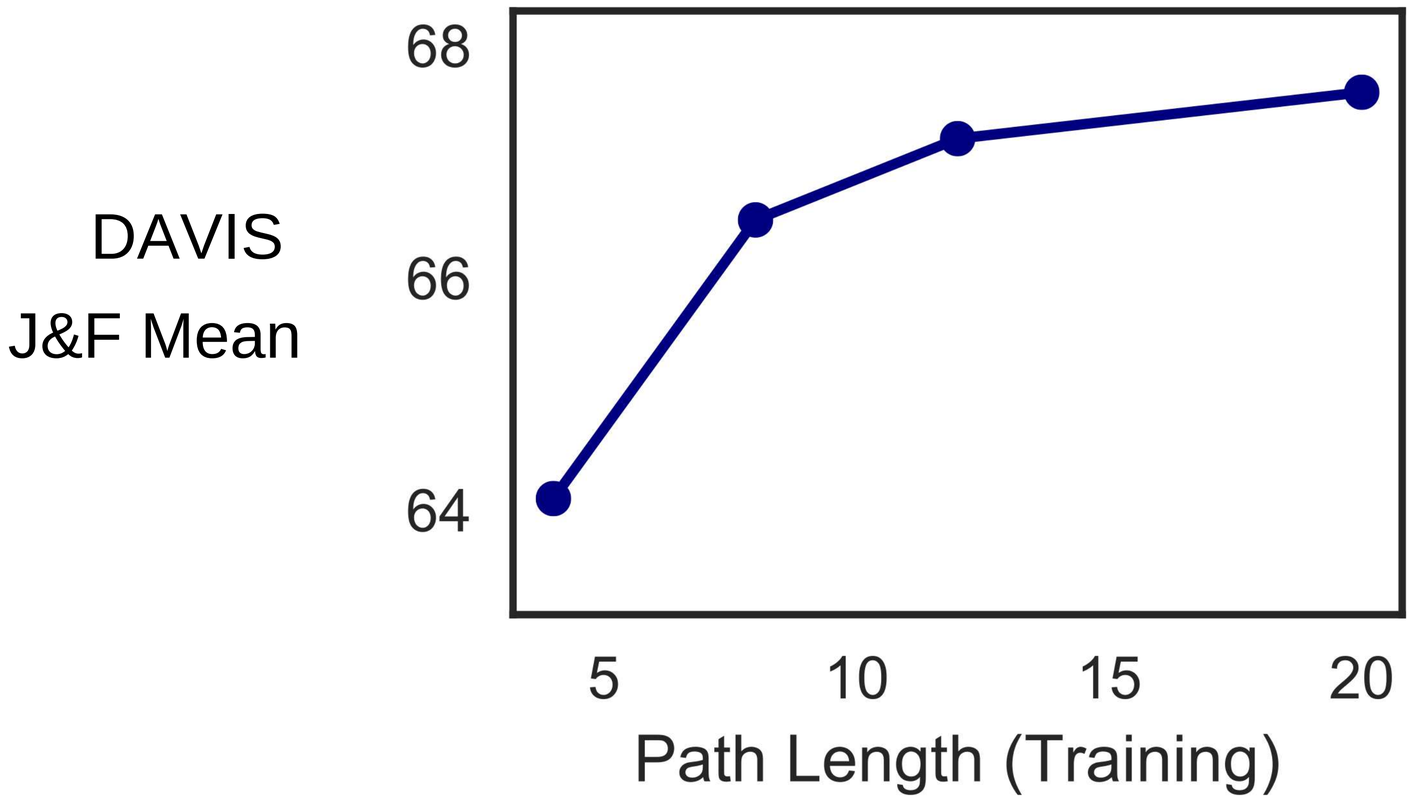
(Self-supervised Test-time Training)

Sun et al, ICML 2020

Test-time Training (DAVIS)



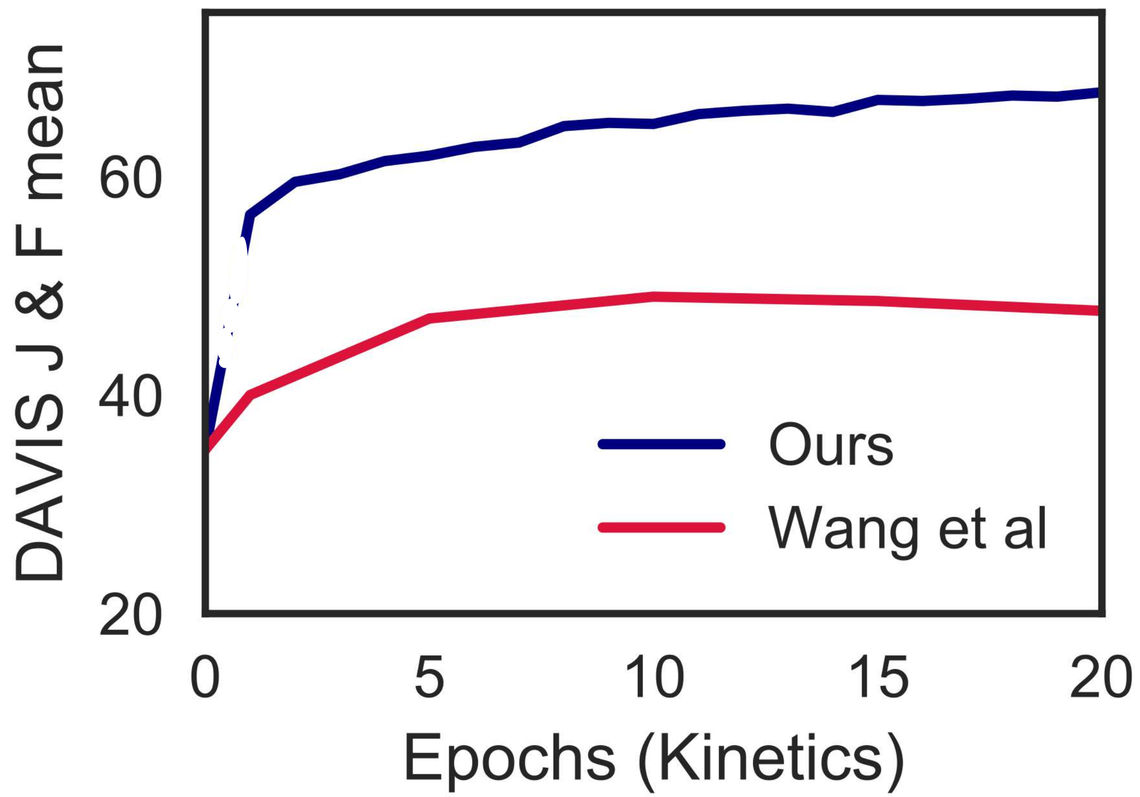
Path Length at Training



Learning Curve

DAVIS

J&F Mean



Conclusions

- We proposed a simple and effective formulation for learning correspondence in a scalable way from unlabeled video.
- The method outperforms self-supervised state of the art methods despite being more simple.