Carnegie Mellon University

Particle Video Revisited:

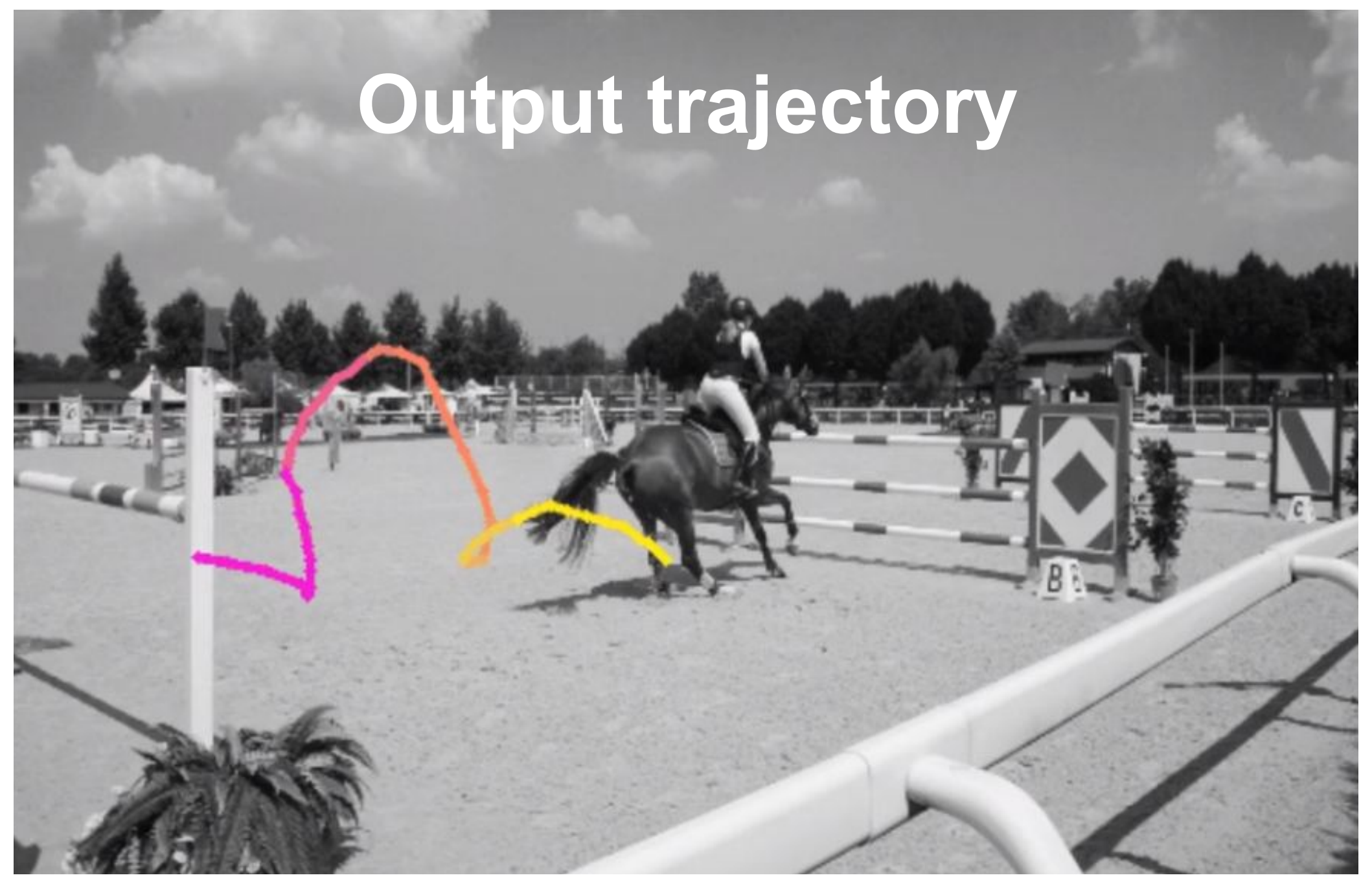
Tracking Through Occlusions Using Point Trajectories

Adam W. Harley, Zhaoyuan Fang, Katerina Fragkiadaki

https://particle-video-revisited.github.io
https://github.com/aharley/pips

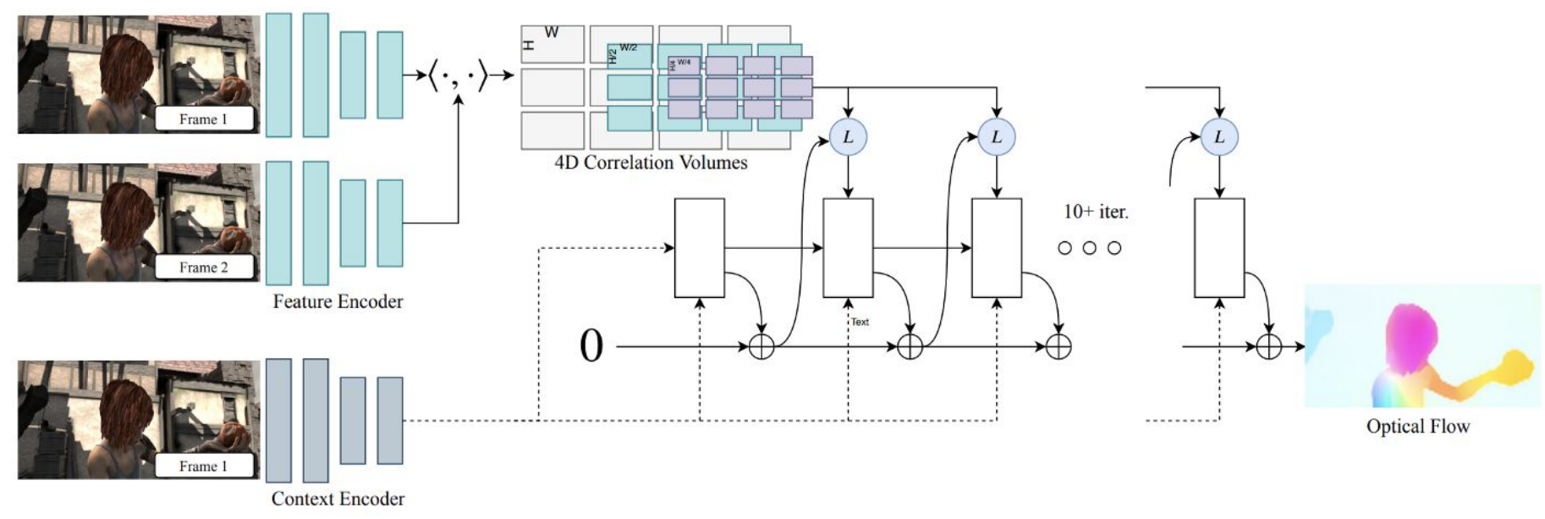
ECCV
TEL AVIV 2022

Goal: Given a target pixel specified on the first frame, track that pixel across the whole video.



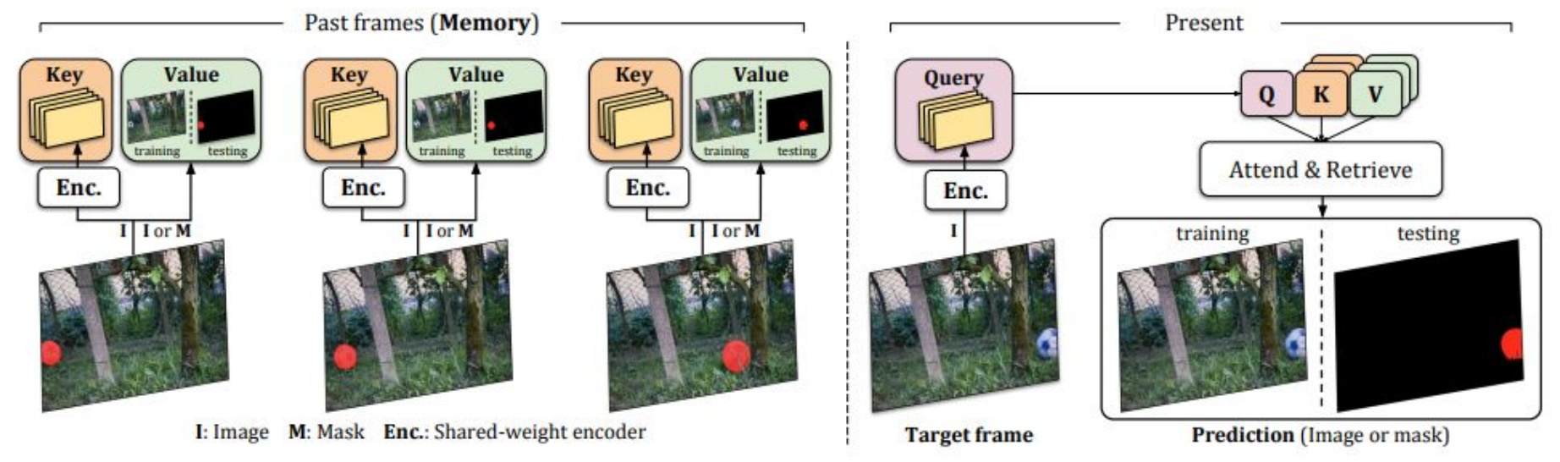
Related: Optical flow and feature matching are general-purpose, but have difficulty with occlusions.

Optical flow provides fine-grained motion, but with a narrow timespan.



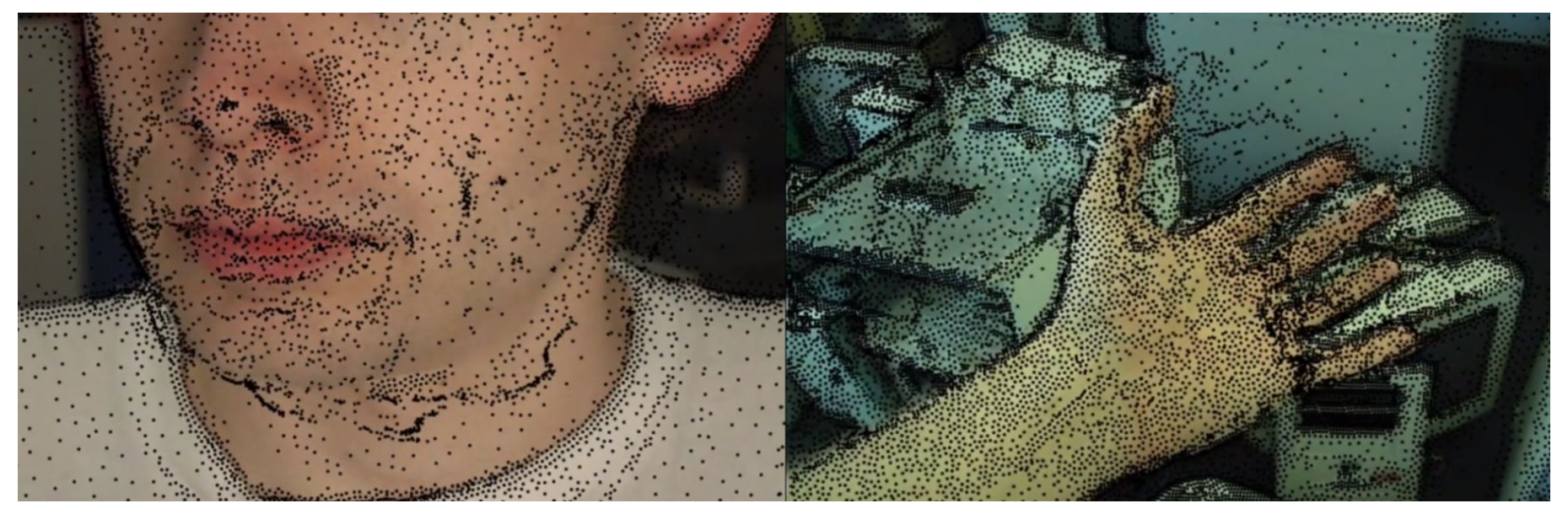
Teed & Deng. RAFT: Recurrent All Pairs Field Transforms for Optical Flow ECCV 2020

Feature matching provides long-range correspondence, but without a temporal prior.



Lai et al. MAST: A Memory-Augmented Self-Supervised Tracker. CVPR 2020.

Sand & Teller (2008) suggest to treat pixels like "particles". Let's revisit this!

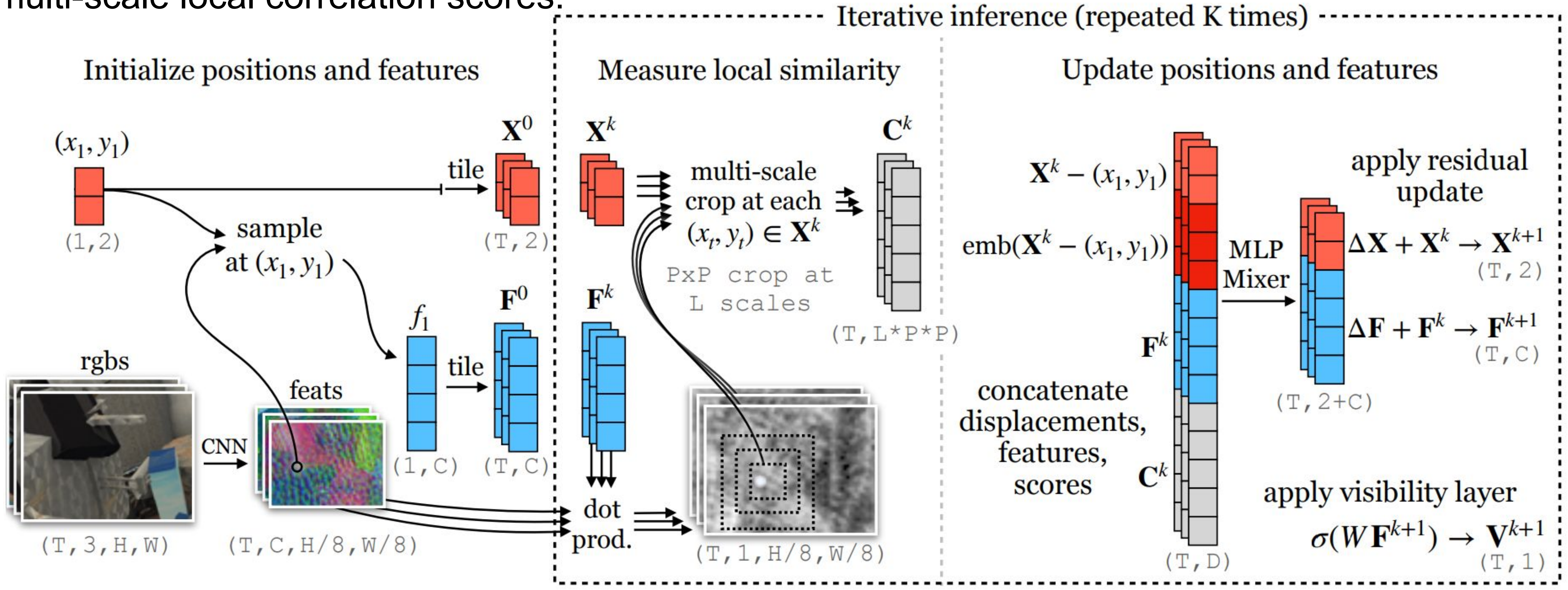


Sand & Teller. Particle Video: Long-Range Motion Estimation Using Point Trajectories. IJCV 2008.

Modern trackers for pedestrians and cars use strong temporal priors to help track through occlusions. Can we bring this power to pixel tracking?

Rajasegaran et al., Tracking People by Predicting 3D Appearance, Location & Pose. CVPR 2022.

***Persistent Independent Particles (PIPs)*:** Every particle is tracked independently. We initialize a zero-velocity trajectory for each particle, and iteratively refine it, inspecting multi-frame multi-scale local correlation scores.


Iterative inference (repeated K times)
Initialize positions and features
Measure local similarity
Update positions and features
(x_1, y_1)
(1,2)
tile
$\mathbf{X}^0$
(T,2)
sample at (x_1, y_1)
f_1
(1,C)
tile
$\mathbf{F}^0$
(T,C)
rgbs
(T,3,H,W)
CNN
feats
(T,C,H/8,W/8)
$\mathbf{X}^k$
$\mathbf{F}^k$
dot prod.
multi-scale crop at each $(x_t, y_t) \in \mathbf{X}^k$
PxP crop at L scales
$\mathbf{C}^k$
(T,L*P*P)
(T,1,H/8,W/8)
$\mathbf{X}^k - (x_1, y_1)$
emb$(\mathbf{X}^k - (x_1, y_1))$
$\mathbf{F}^k$
$\mathbf{C}^k$
concatenate displacements, features, scores
(T,D)
MLP Mixer
(T,2+C)
apply residual update
$\Delta\mathbf{X} + \mathbf{X}^k \rightarrow \mathbf{X}^{k+1}$
(T,2)
$\Delta\mathbf{F} + \mathbf{F}^k \rightarrow \mathbf{F}^{k+1}$
(T,C)
apply visibility layer
$\sigma(W\mathbf{F}^{k+1}) \rightarrow \mathbf{V}^{k+1}$
(T,1)

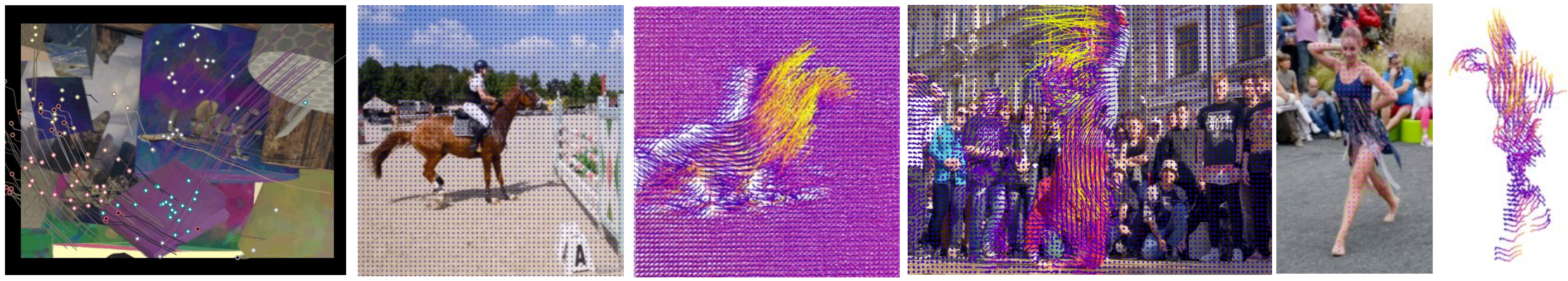
PIPs vs. RAFT

8-frame temporal window instead of 2-frame.

Deep temporal prior via MLP-Mixer, instead of shallow one by convGRU.

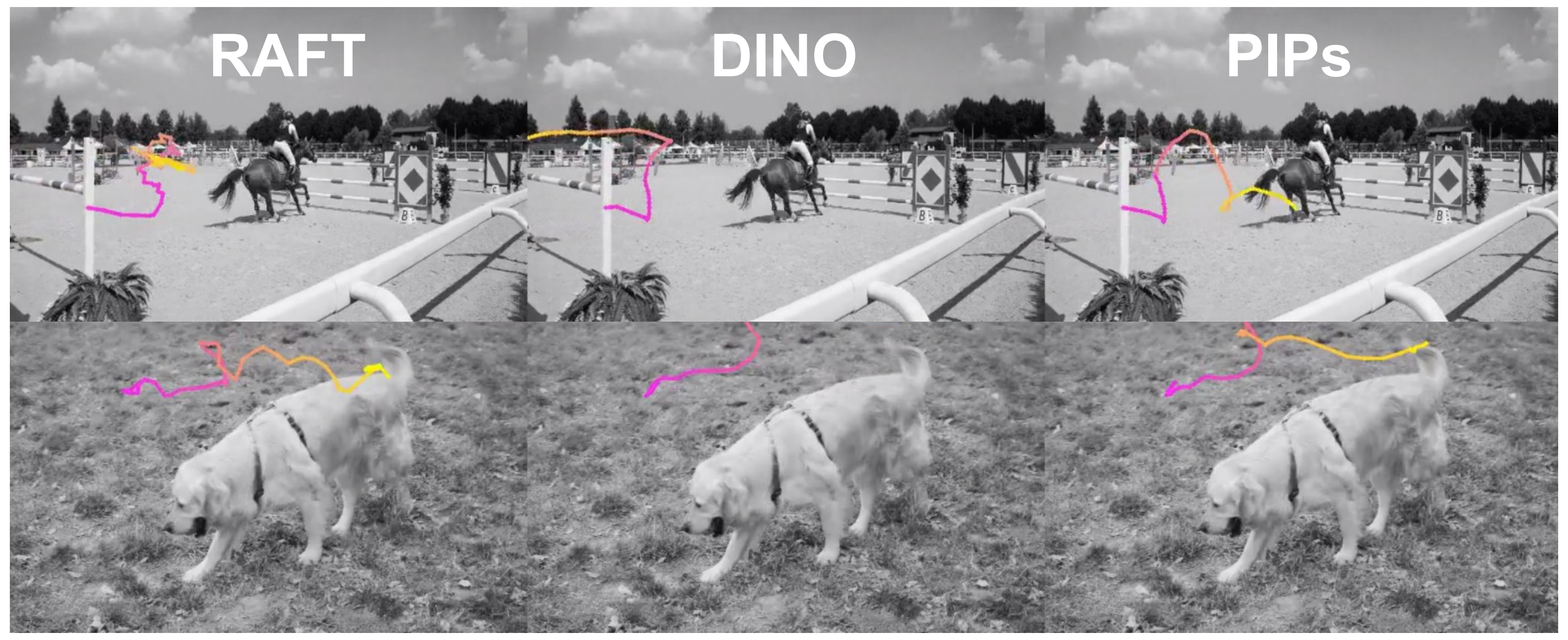
200ms for 8 frames for 256 particles, instead of 200ms/frame.

Results: Outperforms optical flow and feature matching. Despite training on synthetic data, generalizes to YouTube.



Training data consists of "Flying Things" with multi-frame occlusions. All experiments evaluate the same model.

At test time, we chain our 8-frame trajectories to track for arbitrary timespans.



We outperform flow and feature matching in trajectory estimation in FlyingThings, KITTI, CroHD, and BADJA, mostly due to handling occlusions.

Future work: multi-trajectory inference, recurrence.