Rethinking Temporal Consistency in Video Object-Centric Learning

From Prediction to Correspondence

Zhiyuan Li[1] Rongzhen Zhao[1] Wenyan Yang[1]
Wenshuai Zhao[2] Pekka Marttinen[2] Joni Pajarinen[1]

[1] Dept. of Electrical Engineering and Automation, Aalto University [2] Dept. of Computer Science, Aalto University, Finland

ICML 2026

Seeing a Scene as Objects, Not Pixels

We see a scene as separate things. Most networks see one big pixel grid.

Object-centric learning represents a scene as a set of slots — each slot grabs *one object*, learned with **no labels**.

In **video**, the hard part is keeping each object on the same slot as it moves frame to frame.

Figure 1: Each slot reconstructs one object on its own. [Locatello et al., NeurIPS 2020]

The Field's Fix Is Expensive — Is It Needed?

Guess objects from scratch

Slots start as random/blank vectors — many refinement steps just to *find* the objects.

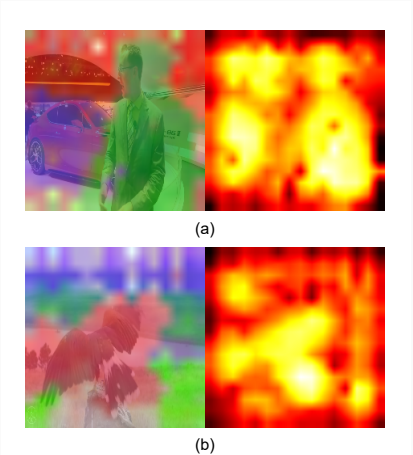
Learn to predict motion

A heavy Transformer/RNN forecasts where every object goes next — large compute cost.

Every modern method assumes **both** are necessary.

Are they?

Surprise 1: The Backbone Already Finds Objects



Modern self-supervised vision models (DINOv2) already highlight objects in their features — no extra training.

Idea: just place each slot *right where the backbone is already looking*.

Result: objects found in one step instead of many — **faster and a little better**.

Figure 2: A frozen self-supervised model already "lights up" on objects — for free.

Surprise 2: The Motion Predictor Barely Helps

We deleted the learned motion predictor and simply **carried each slot forward unchanged**.

Tracking module	Quality	
Heavy learned predictor	32.1	
Nothing (just copy)	**33.0**	✓

YouTube-VIS, segmentation quality (ARI). Higher is better.

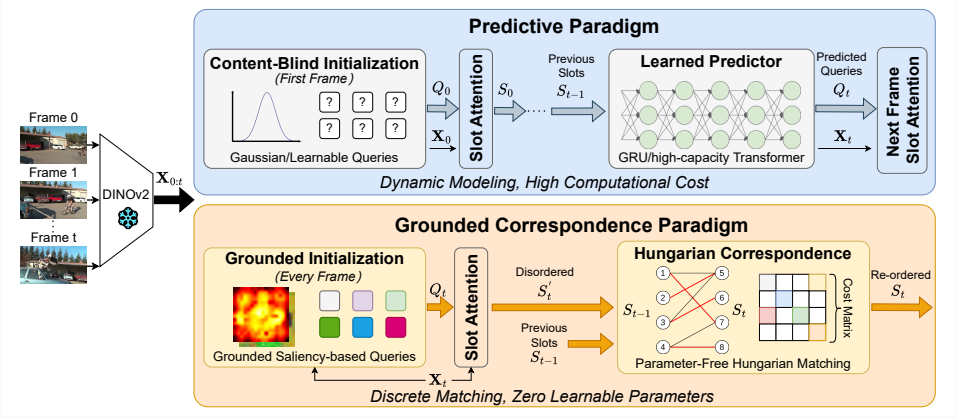
Why?

The predictor wasn't modeling physics — it was just keeping slots in order.

That is a **matching** job, not a prediction job.

Tracking objects = **matching**, not predicting.

Our Approach: Grounded Correspondence



1. Seed slots on backbone highlights → 2. Match slots between frames

Zero learned parameters for tracking.

It Works — With Zero Learned Tracking

Table 1: Object-discovery quality (higher is better). Ours uses **no learned tracking parameters**.

	MOVi-D			MOVi-E			YouTube-VIS		
	ARI	FG-ARI	mBO	ARI	FG-ARI	mBO	ARI	FG-ARI	mBO
Prior best (SlotContrast)	58.0	58.0	**30.0**	68.9	68.9	**26.6**	**32.1**	**36.3**	**29.9**
Ours	**73.7**	**73.7**	28.4	**75.7**	**75.7**	23.4	30.1	33.1	29.3

+15.7
MOVi-D (synthetic)

+6.8
MOVi-E (synthetic)

Competitive
YouTube-VIS (real video)

Seeing Is Believing

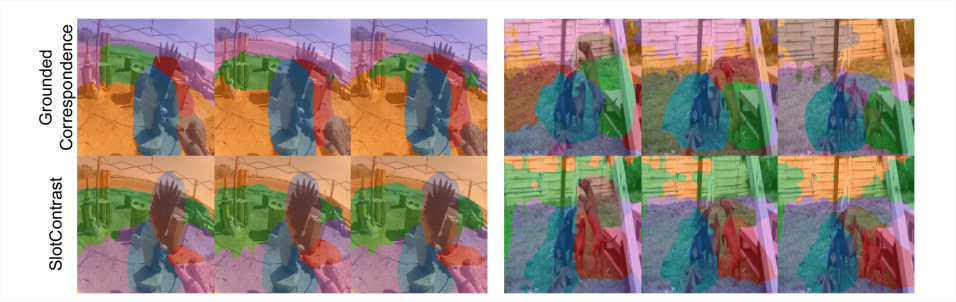


Figure 3: Real YouTube videos. Each color = one tracked object. Top: **ours**. Bottom: prior best.

Colors stay glued to each object across frames — with **no learned tracking**.

Takeaway

Tracking objects in video is a
matching problem, not a prediction problem.

What we showed:

- Backbones already locate objects — start there
- Learned motion predictors are mostly redundant
- Simple matching ⇒ **zero tracking parameters**

Honest limits:

- Objects lost after full occlusion
- Fixed number of slots
- Matching cost grows with #objects

Thank you!

Questions welcome.

Project page & code → scan the QR codes.

Funded by the Research Council of Finland (FCAI; 357301, 358246); compute via CSC / LUMI and Aalto Science-IT.

ICML International Conference On Machine Learning

Aalto University

FCAI

Project page

Code