Is ImageNet worth 1 video? Learning strong image encoders from 1 long unlabelled video

Shashanka Venkataramanan, Mamshad Nayeem Rizve, João Carreira, Yuki M. Asano, Yannis Avrithis

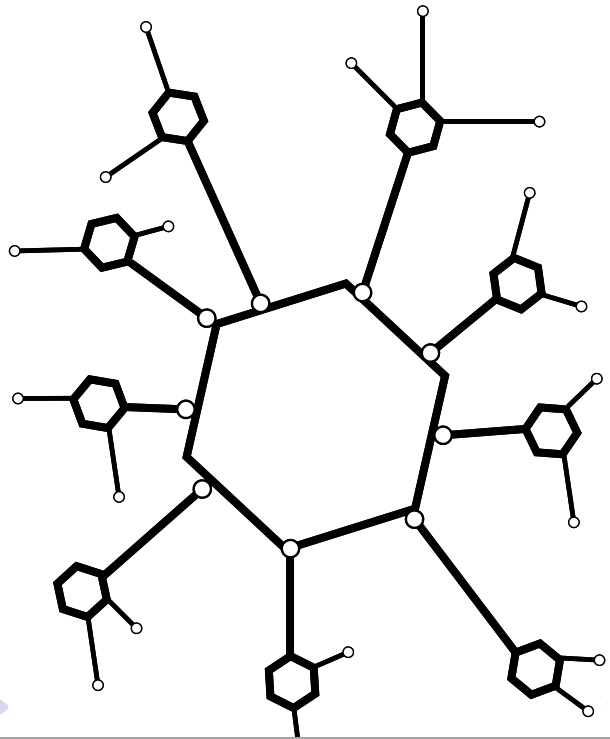
ICLR[2024]

Zakaria Akil & Adnane El Bouhali

Deep Learning for Computer Vision

31/01/2025

Contents



Introduction

Self Supervised learning (SSL)

SSL is used to pretrain encoders to learn meaningful representations of unlabeled data. Those learned representations can be subsequently used as input to various downstream tasks.

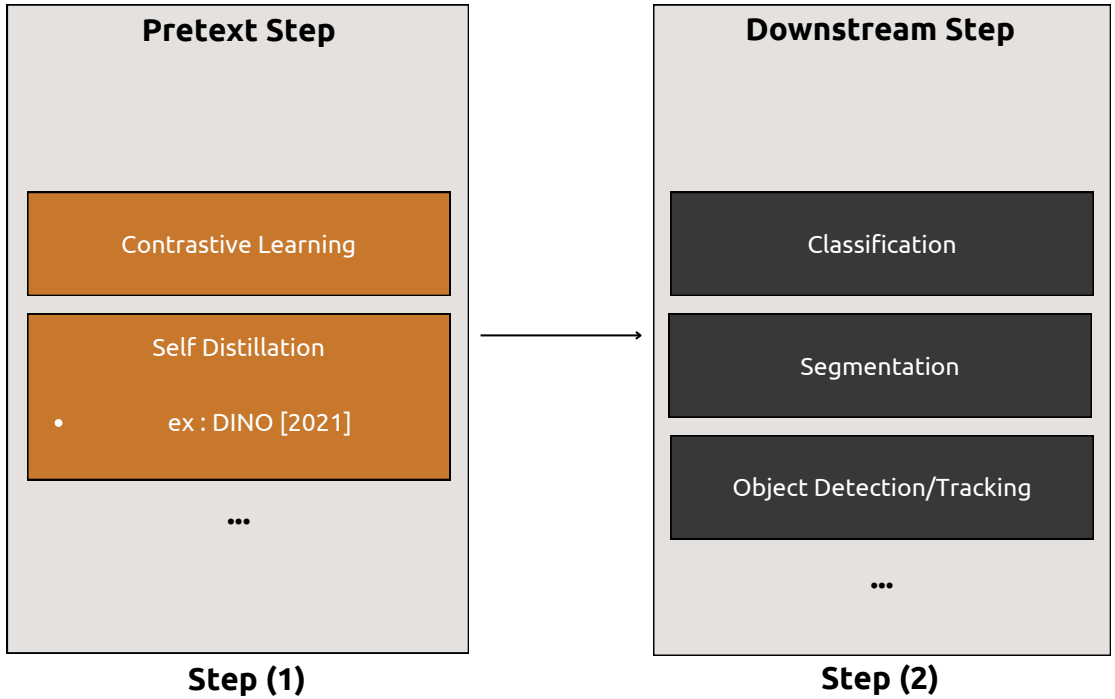


Figure: SSL principle and some methods

Problem statement

Problem statement

Self-supervised learning (SSL) has scaled to billions of images, but are we using the data efficiently?

Problem statement

Problem statement

Self-supervised learning (SSL) has scaled to billions of images, but are we using the data efficiently?

$\Longrightarrow$ Not really ...

Problem statement

Current SSL methods rely on large datasets like **ImageNet**.

Some disadvantages :

- Inefficient use of data
- High computational cost
- Domain mismatch
- Poor generalization

Paper Contributions

Walking Tours (WT) Dataset

WT : 10 continuous videos from cities such as Amsterdam, Bangkok, and Venice.

- High resolution
- 1~3h x 10
- Egocentric
- No special effects or transformations
- Natural transitions and no cuts

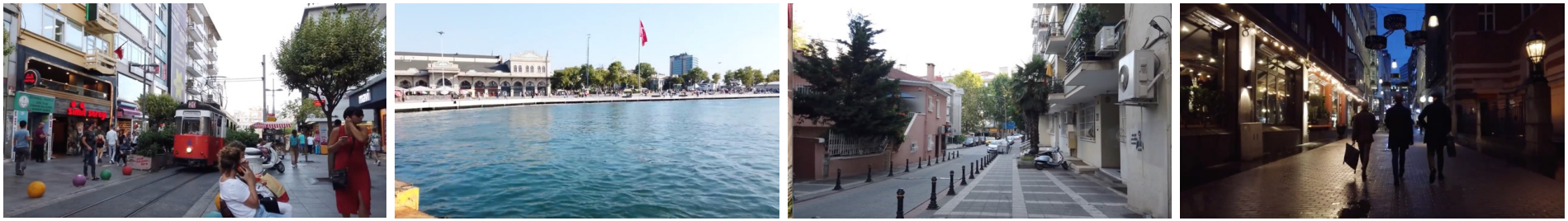


Figure: Some WT frames

Walking Tours (WT) Dataset

Dataset	Domain	Ego	Pre	Bal	Annot	Avg. Dur (sec)	Dur (hr)	#Videos	Frame resolution
Diverse Pretraining									
Kinetics-400 [?]	Actions	✗	✓	✓	**Class**	10.2	851	400	340 × 255
WebVid-2M [?]	Open	✗	✓	✗	**Weak**	18	13k	–	320 × 240
HowTo100M [?]	Instructions	✗	✓	✗	**Weak**	4	135k	–	–
Egocentric									
Epic-Kitchens [?]	Cooking	✓	✗	✗	**Loc.**	510	100	37	1920 × 1080
Ego-4D [?]	Daily	✓	✗	✗	**Loc.**	1446	120	931	1920 × 1080
Meccano [?]	Industry	✓	✗	✗	**Loc.**	1247	849	20	1920 × 1080
Assembly-101 [?]	Assembly	✓	✗	✗	**Loc.**	426	167	362	1920 × 1080
ImageNet-aligned									
R2V2 [?]	ImageNet	✗	✓	✓	**Class**	–	–	–	467 × 280
VideoNet [?]	ImageNet	✗	✓	✓	**Class**	10	3055	–	
Walking Tours (ours)	Urban	✓	✓	✗	**None**	5880	23	10	3840 × 2160

Table: *Walking Tours existing video datasets*. **Ego**: egocentric; **Pre**: used for pretraining; **Bal**: class balance control; **Annot**: annotation type. Weak: associated data per clip (text or other modality); Class: class label per frame or clip; Loc: localization per frame (bounding box, segmentation, mask 3D pose). **Avg. Dur**: average duration per video; **Dur**: total duration.

DoRA

DoRA (**Dis**c**o**ver and **RA**ck objects): A self-supervised pretraining method designed for continuous videos.

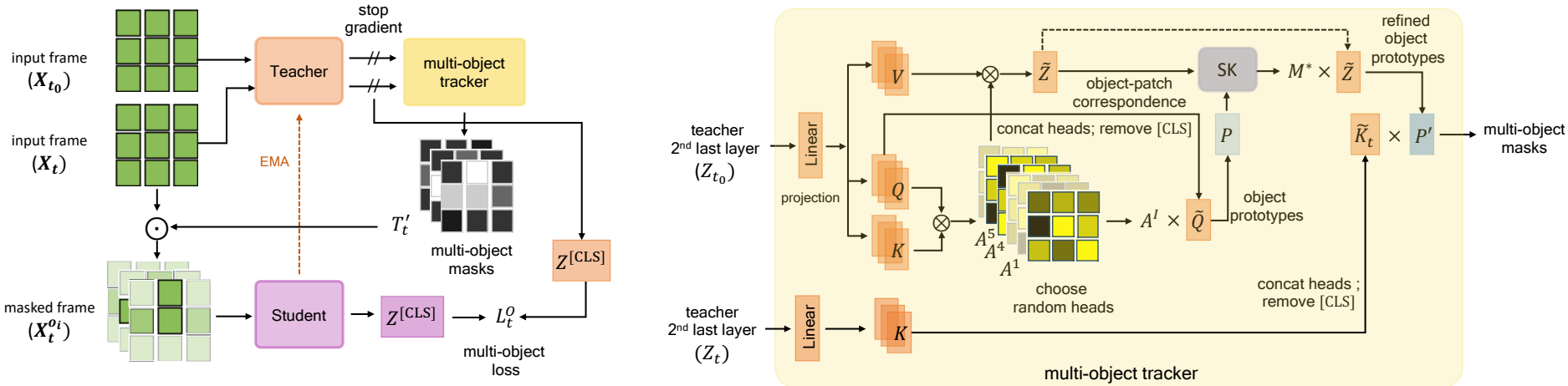


Figure: DoRA Architecture

High level architecture

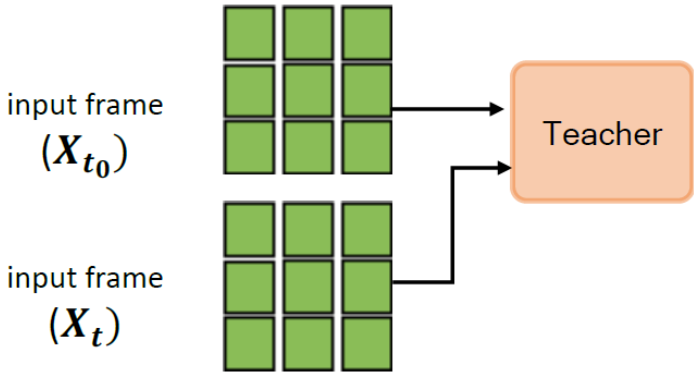


Figure: High level architecture

High level architecture

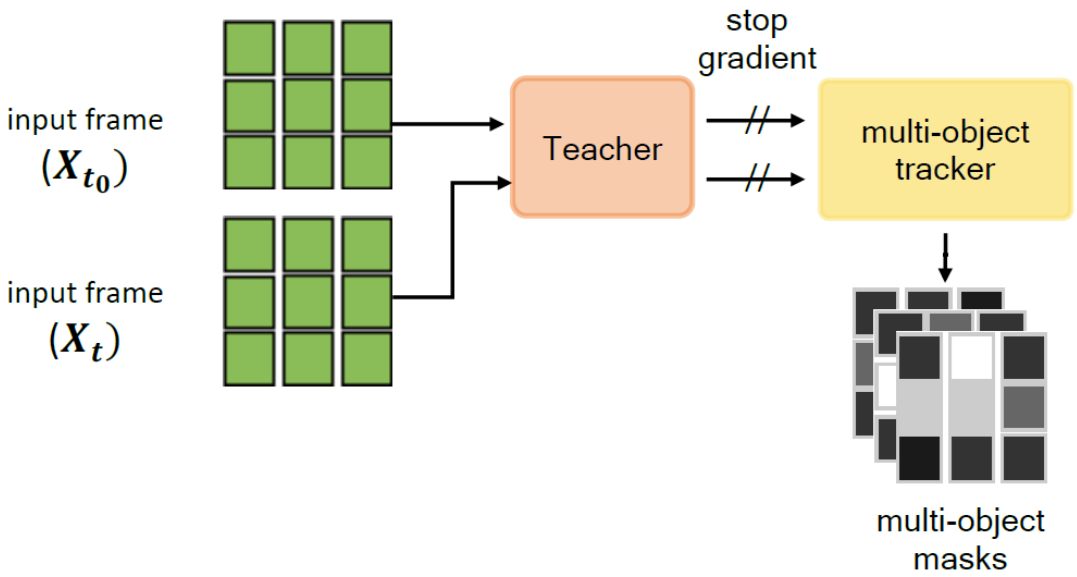


Figure: High level architecture

High level architecture

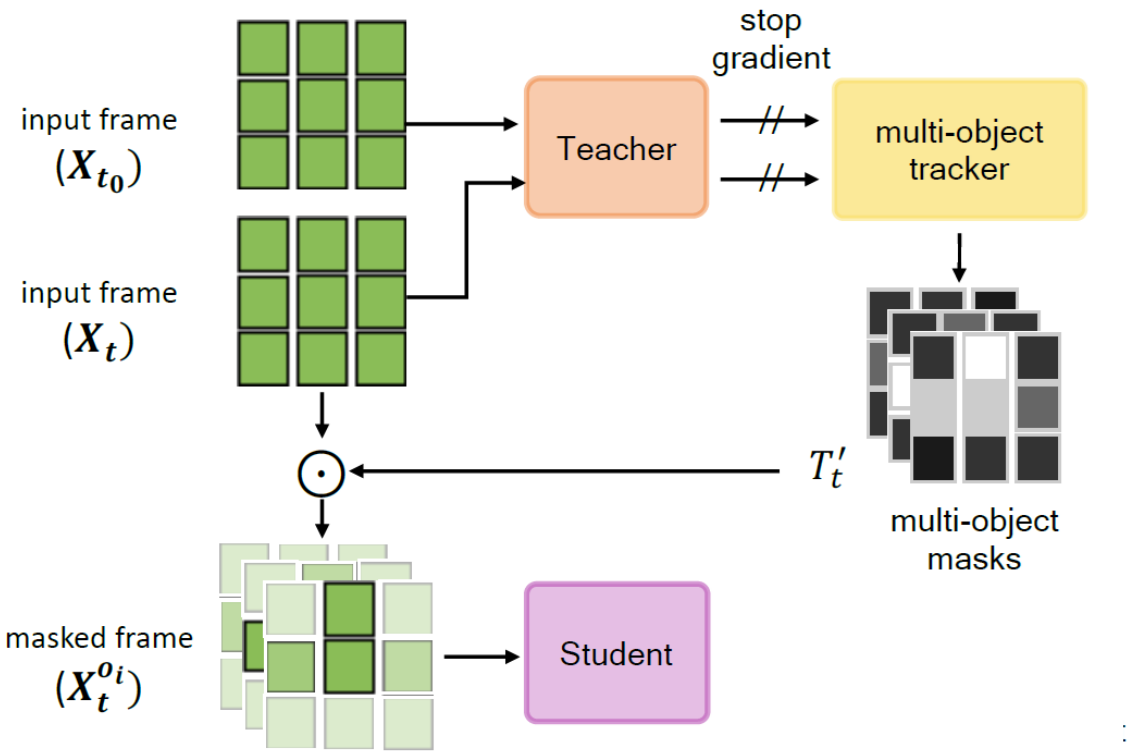


Figure: High level architecture

High level architecture

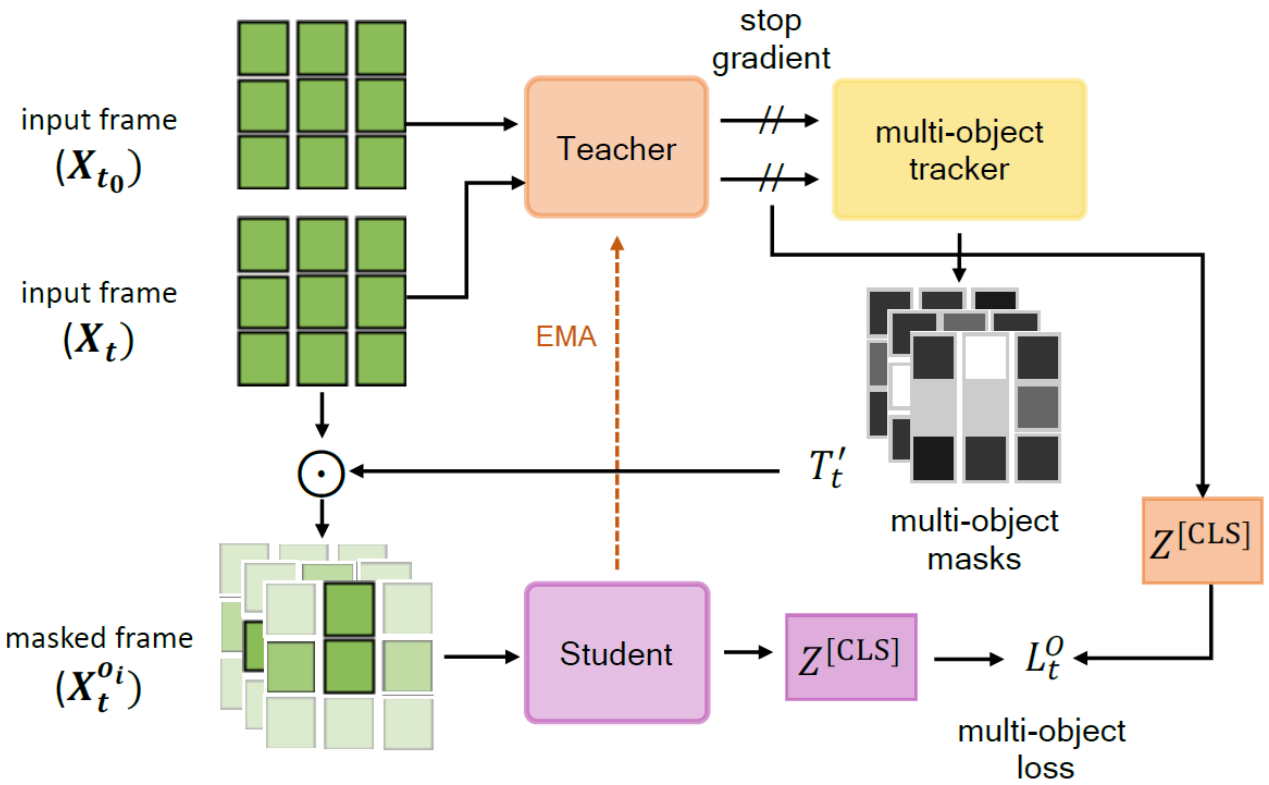


Figure: High level architecture

High level architecture

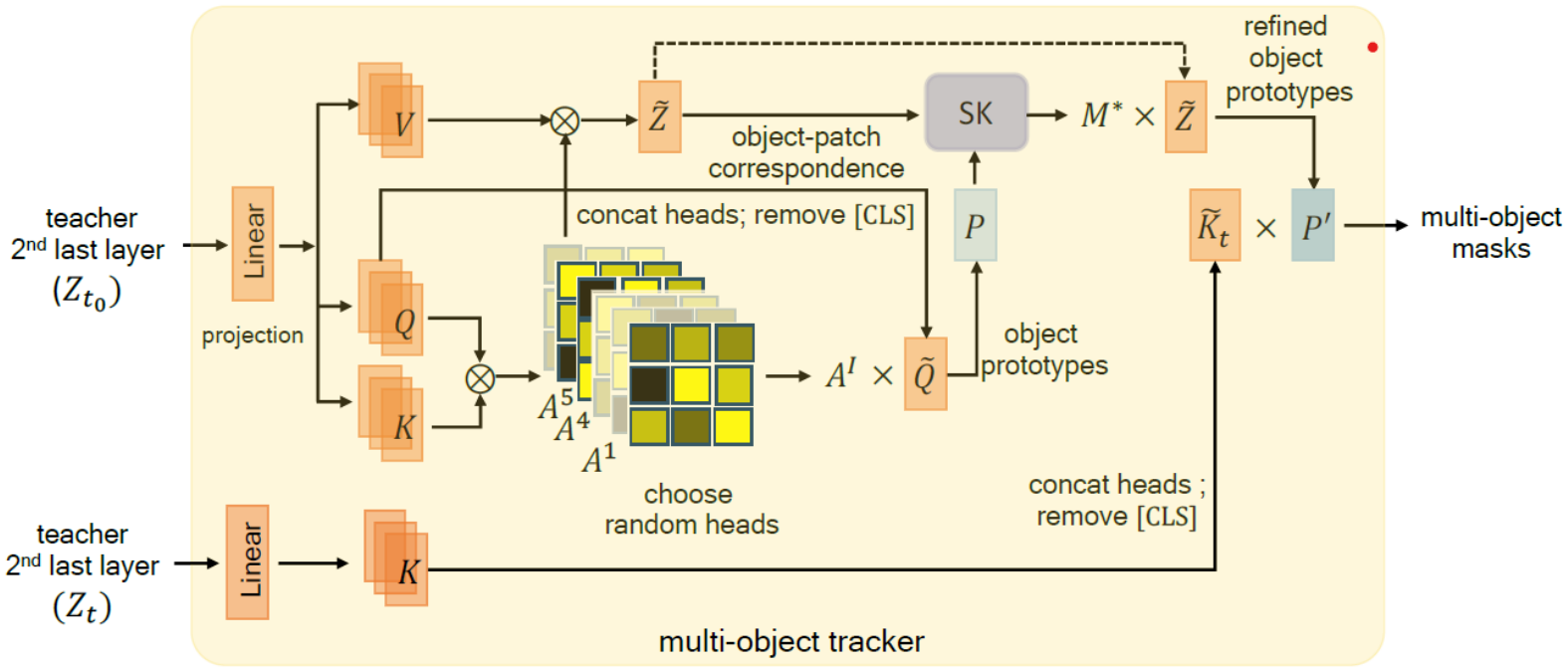


Figure: High level architecture

DoRA - SK refinement

- T'_t : Cross attention maps with Sinkhorn-Knopp refinement
- T_t : without SK refinement

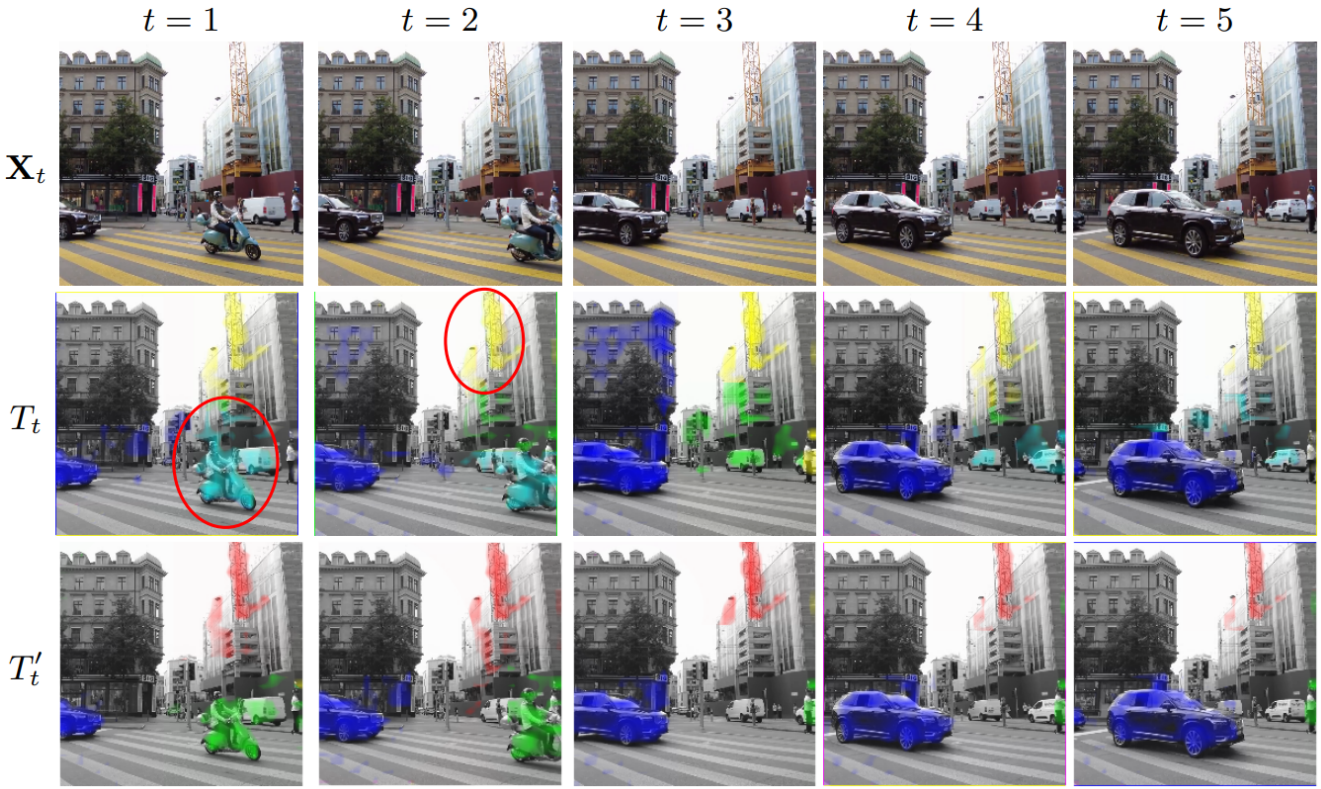
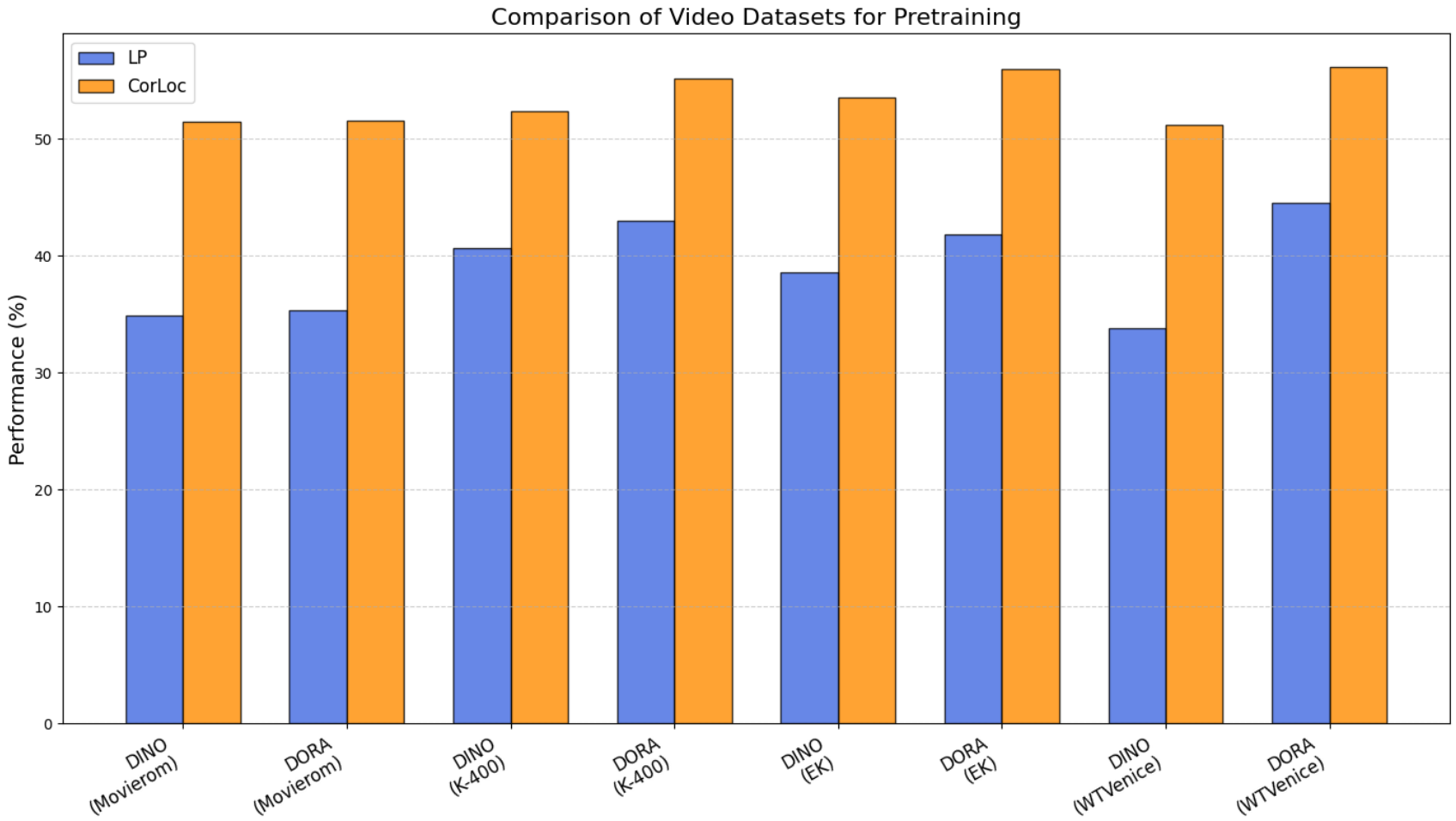


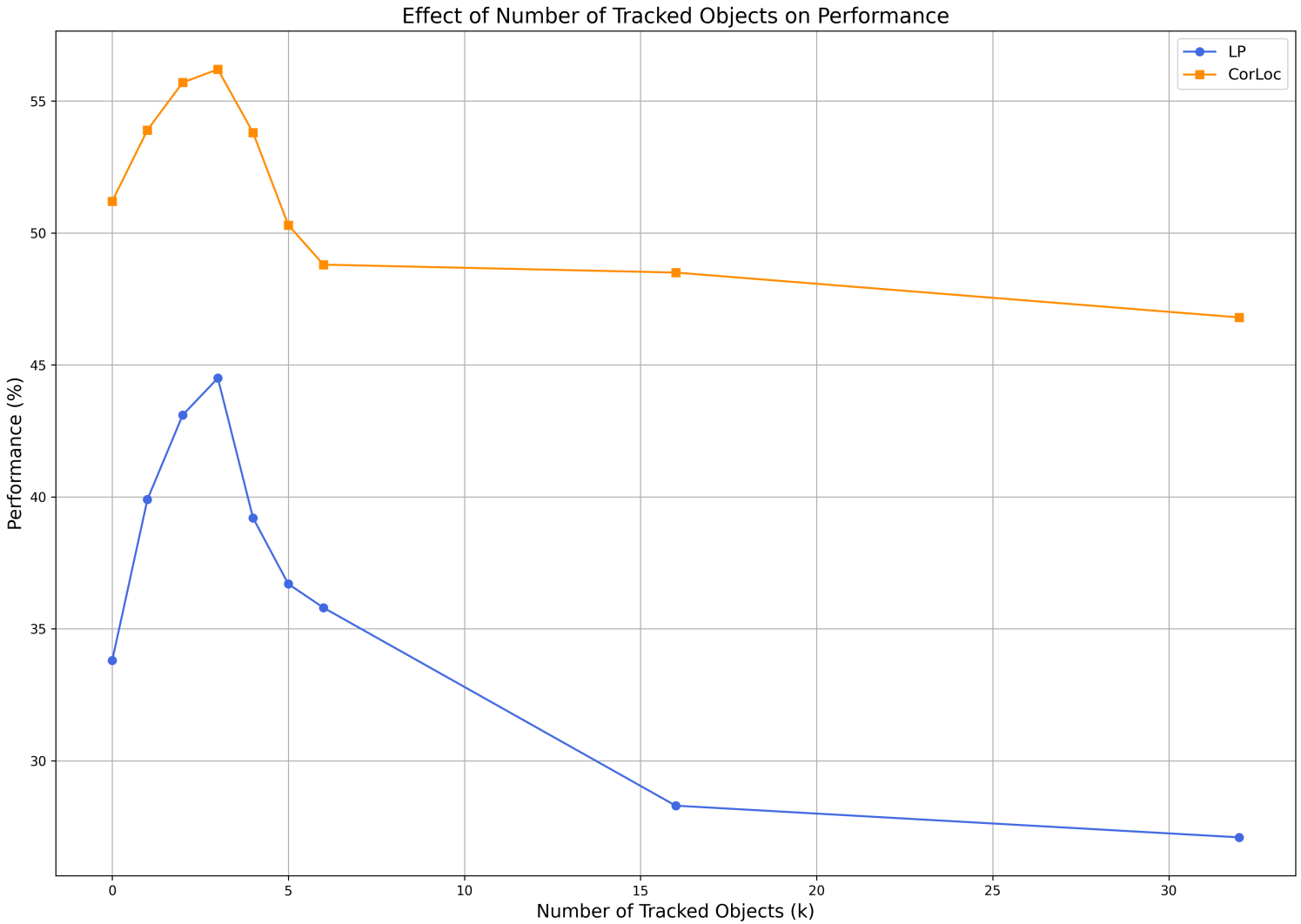
Figure: Reshaped and resampled cross-attention maps in code color (R, G and B)

Experiments & Results

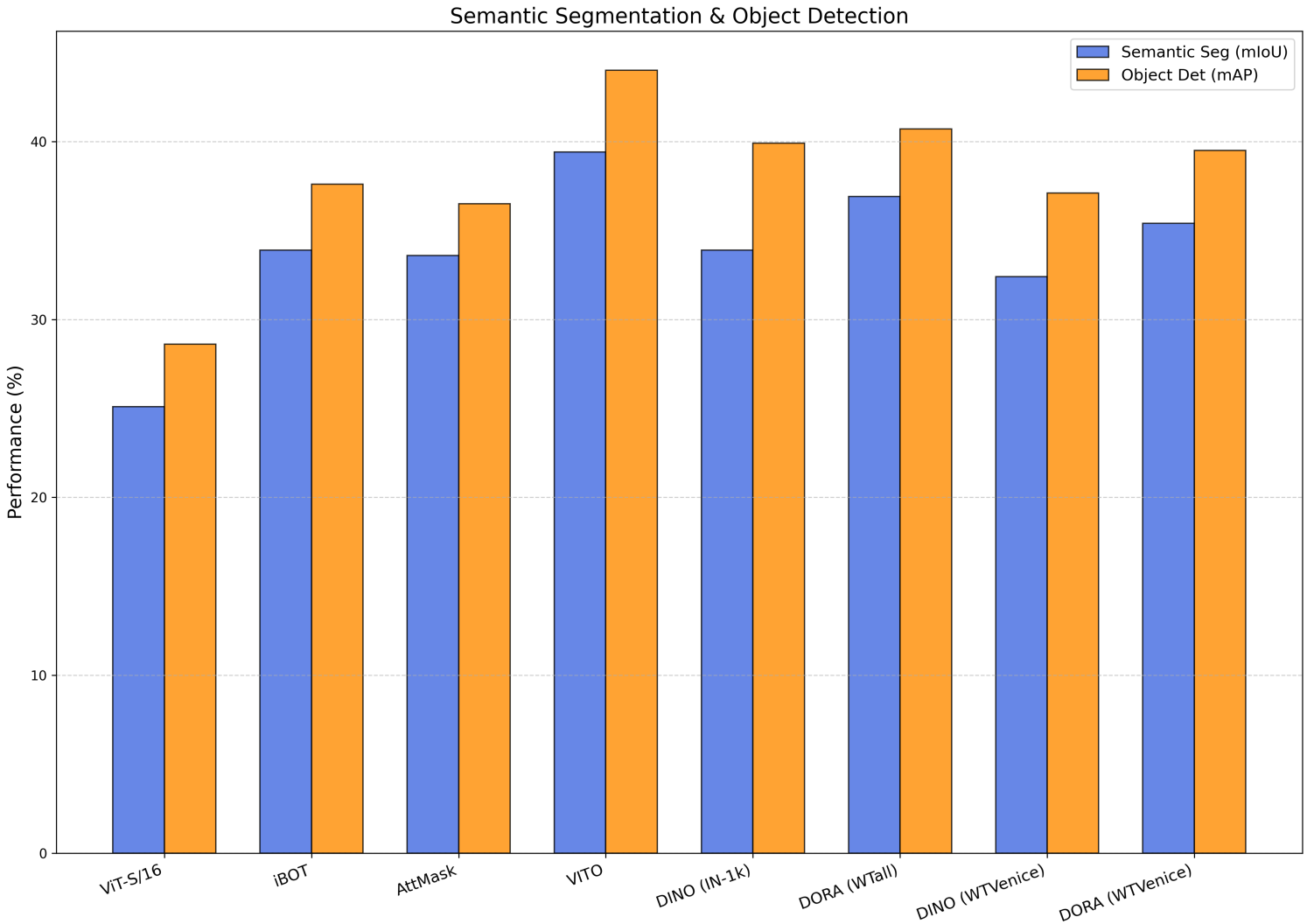
Effect of Pretraining video dataset



Effect of the Number of Tracked Objects



Segmentation and Object Detection Performance



Video Object Segmentation and Object Tracking Performance

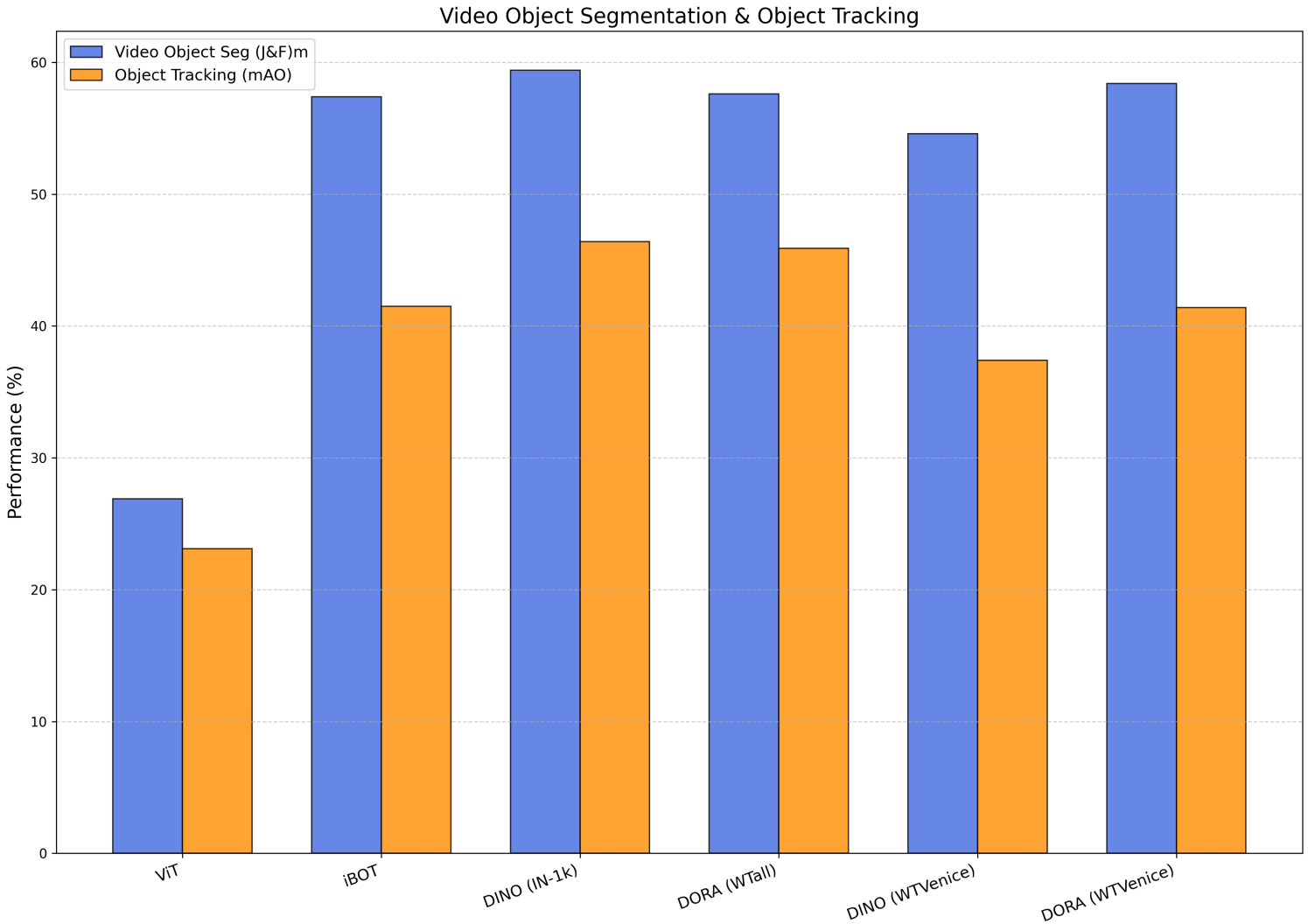
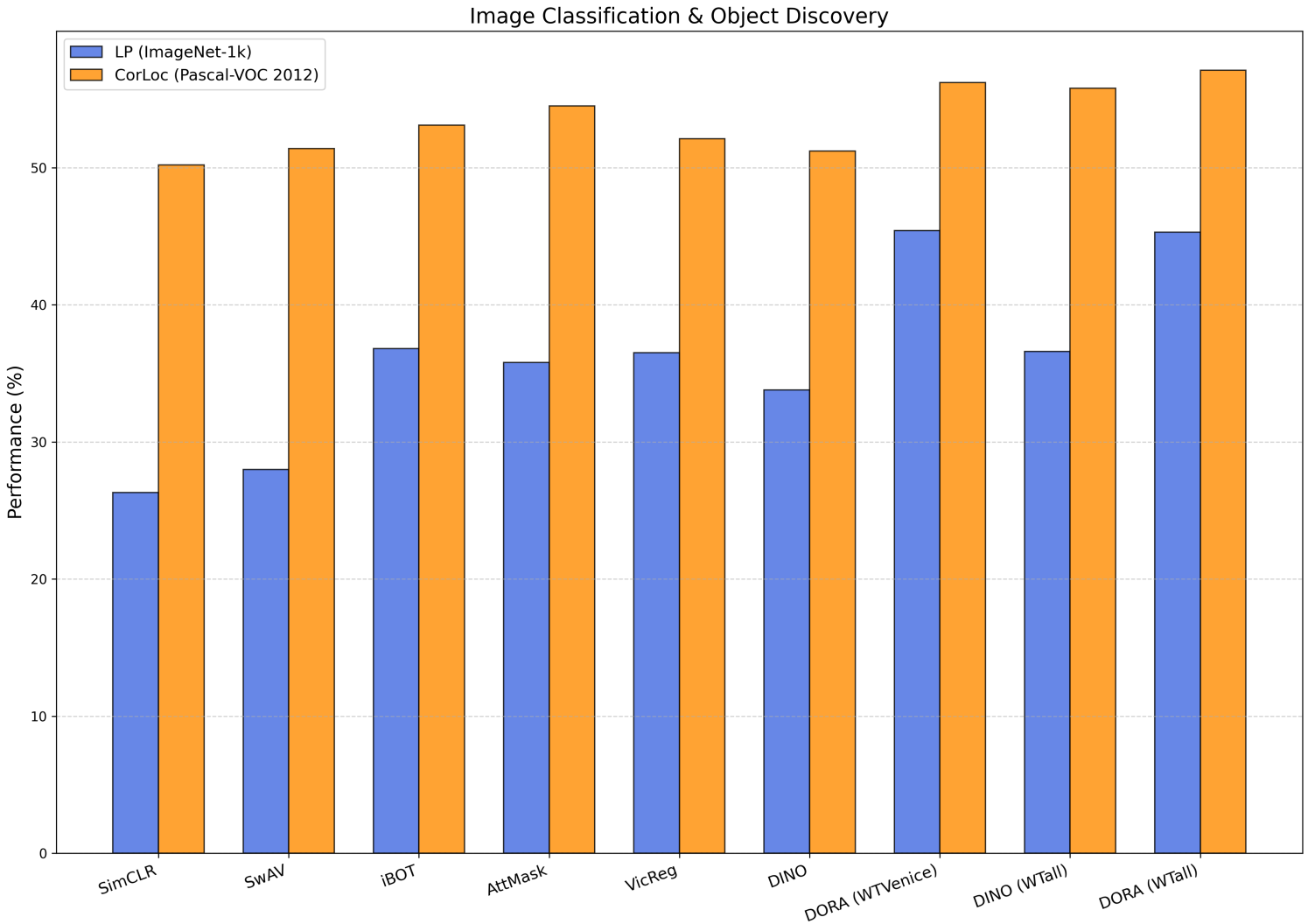


Image Classification and Object Discovery Performance



Conclusion

Conclusion

- **New Dataset**: 10 long, high-resolution, uncut walking tour videos.
- **Strong Representations**: Matches **ImageNet-level** performance on downstream tasks.
- **DORA (New SSL Method)**: Adapts **DINO** for video with **multi-object tracking**.
- **Efficient Learning**: Achieves results **without large-scale video datasets**.
- **Emergent Attention**: Transformer learns to track objects **through occlusions**.
- **Impact**: Advances **self-supervised learning** from video for real-world applications.

References

[1] [Reviewed Paper] Shashanka Venkataramanan, Mamshad Nayeem Rizve, João Carreira, Yuki M. Asano, Yannis Avrithis (2024), *"Is ImageNet worth 1 video? Learning strong image encoders from 1 long unlabelled video"*, https://arxiv.org/abs/2310.08584

Thank you!

Appendix

Contrastive Learning and Contrastive Loss

Contrastive Learning:

- Goal: Learn representations by contrasting positive pairs (similar samples) against negative pairs (dissimilar samples).
- **Contrastive Loss** (e.g., SimCLR):

$$\mathcal{L}_{\text{contrastive}} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(z_i, z_k)/\tau)}$$

where z_i, z_j are embeddings of positive pairs, sim is a similarity function (e.g., cosine similarity), and τ is a temperature parameter.

Self-Distillation and Distillation Loss

Self-Distillation:

- Goal: A student model learns from a teacher model (often an exponential moving average of the student).
- **Distillation Loss** (e.g., DINO):

$$\mathcal{L}_{\text{distill}} = -\sum_i P_{\text{teacher}}(z_i) \log P_{\text{student}}(z_i)$$

where P_{teacher} and P_{student} are the output distributions of the teacher and student models, respectively.

DoRA (Discover and TRack Objects):

- **Step 1: Patch Embedding**:

$$Z_t = g_\theta(X_t) \in \mathbb{R}^{(n+1)\times d}$$

where X_t is the input frame, g_θ is the transformer encoder, and Z_t contains patch embeddings and a [CLS] token.

- **Step 2: Multi-Head Attention**:

$$A^i = \text{softmax}\left(\frac{Q^i (K^i)^\top}{\sqrt{d}}\right)$$

where Q^i, K^i are query and key embeddings for head i, and A^i is the attention matrix.

- **Step 3: Object Prototypes**:

$$P = A^{\mathcal{I}} \tilde{Q} \in \mathbb{R}^{k\times d}$$

where $A^{\mathcal{I}}$ is a subset of attention heads, and P represents k object prototypes.

- **Step 4: Sinkhorn-Knopp (SK) for Object-Patch Correspondence**:

$$M^* = \text{SK}\left(\exp\left(\frac{P\tilde{Z}^\top}{\epsilon}\right)\right)$$

where M^* is the optimal transport plan, and SK ensures distinct object correspondences.

- **Step 5: Refined Object Prototypes**:

$$P' = M^*\tilde{Z} \in \mathbb{R}^{k\times d}$$

where P' are refined object prototypes after SK.

- **Step 6: Refined Cross-Attention**:

$$T_t' = \text{softmax}\left(\frac{P'\tilde{K}_t^\top}{\sqrt{d}}\right)$$

where T_t' tracks objects across frames using refined prototypes.

- **Step 7: Multi-Object Masking**:

$$X^{o_i} = X \odot T^i$$

where X^{o_i} is the masked frame for object i, and $\odot$ is the Hadamard product.

- **Step 8: Distillation Loss**:

$$\mathcal{L}_t^0 = \sum_{u,v \in V} \mathbb{1}_{u!=v} \sum_{i=1}^{k} f_{\theta'}(X_t^u)^{[\text{CLS}]} \log(f_\theta(X_t^{v,o_i})^{[\text{CLS}]})$$

where $f_{\theta'}$ and f_θ are the teacher and student models, respectively.