



UvA-DARE (Digital Academic Repository)

{MoSiC: Optimal-Transport Motion Trajectory for Dense Self-Supervised Learning

Salehidehnavi, S.; Venkataramanan, S.; Simion, I.; Gavves, E.; Snoek, C.G.M.; Asano, Y.M.

Publication date

2025

Document Version

Author accepted manuscript

Published in

International Conference on Computer Vision

[Link to publication](#)

Citation for published version (APA):

Salehidehnavi, S., Venkataramanan, S., Simion, I., Gavves, E., Snoek, C. G. M., & Asano, Y. M. (2025). {MoSiC: Optimal-Transport Motion Trajectory for Dense Self-Supervised Learning. In *International Conference on Computer Vision*

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

MoSiC: Optimal-Transport Motion Trajectory for Dense Self-Supervised Learning

Mohammadreza Salehi,^{1,3*} Shashanka Venkataramanan,^{2*} Ioana Simion,¹
Efstratios Gavves,¹ Cees G. M. Snoek,¹ Yuki M Asano³
¹ VIS Lab, UvA ² Valeo.ai ³ Fundamental AI Lab, UTN

Abstract

Dense self-supervised learning has shown great promise for learning pixel- and patch-level representations, but extending it to videos remains challenging due to the complexity of motion dynamics. Existing approaches struggle as they rely on static augmentations that fail under object deformations, occlusions, and camera movement, leading to inconsistent feature learning over time. We propose a motion-guided self-supervised learning framework that clusters dense point tracks to learn spatiotemporally consistent representations. By leveraging an off-the-shelf point tracker, we extract long-range motion trajectories and optimize feature clustering through a momentum-encoder-based optimal transport mechanism. To ensure temporal coherence, we propagate cluster assignments along tracked points, enforcing feature consistency across views despite viewpoint changes. Integrating motion as an implicit supervisory signal, our method learns representations that generalize across frames, improving robustness in dynamic scenes and challenging occlusion scenarios. By initializing from strong image-pretrained models and leveraging video data for training, we improve state-of-the-art by 1% to 6% on six image and video datasets and four evaluation benchmarks. The implementation is publicly available at our GitHub repository: github.com/MSMD75/MoSiC

1. Introduction

Dense self-supervised learning has emerged as a powerful paradigm for learning rich pixel- and patch-level representations without reliance on labeled data [24, 59, 71]. While significant progress has been made in the image domain, leveraging videos for self-supervised learning presents an even greater opportunity. Videos not only provide vast amounts of readily available data but also introduce a natural temporal dimension, which could be crucial for learning representations that extend beyond static imagery. However, directly applying image-based self-supervised learning techniques to

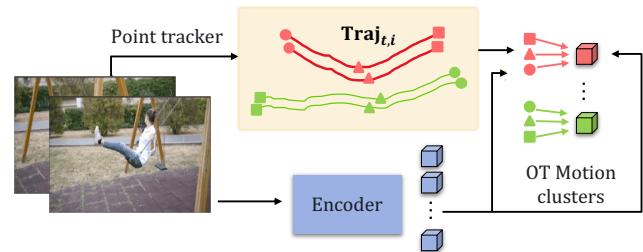


Figure 1. We introduce MoSiC, a self-supervised framework that uses optimal transport (OT) to enforce spatio-temporal consistency in dense visual representations. By tracking points across frames and clustering their motion trajectories, MoSiC encourages features of points that move together to be temporally coherent.

videos proves ineffective [4, 57]. In images, correspondences are often implicitly encoded by data augmentations—for instance, color transformations preserve pixel-wise associations. In contrast, videos exhibit complex motion patterns caused by object movement, camera shifts, and dynamic backgrounds, breaking such pixel-level correspondences.

In this work, we aim to learn temporally coherent dense representations by leveraging motion as an implicit supervisory signal. Rather than treating frames independently, we enforce feature consistency over time by clustering points that move together. This encourages representations that are not only spatially meaningful but also temporally stable, improving robustness to occlusions and viewpoint changes. Prior works have explored various strategies for learning dense representations in videos, including architectural modifications to track multiple objects in ViTs [61], joint-embedding architectures with specialized masking strategies [4, 7], and temporally coherent clustering [50]. In this work, we propose a new approach inspired by the Gestalt principle of "what moves together belongs together" [32]. We bring this principle to a much finer level of granularity and introduce MoSiC in Figure 1, a self-supervised Motion-based Sinkhorn Clustering framework that clusters points along motion tracks to learn spatiotemporally consistent representations.

Our approach begins by extracting dense point tracks that

*Equal Contribution. Correspondence: s.salehidehnavi@uva.nl

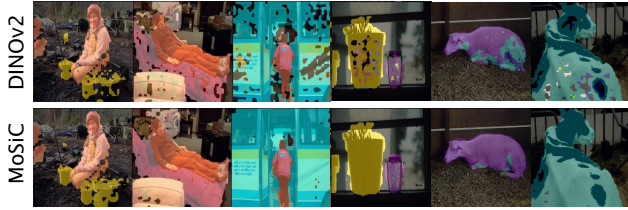


Figure 2. *In-context scene understanding on Pascal VOC* MoSiC improves DINOv2’s dense representations on unlabeled videos, resulting in precise segmentation boundaries and better object identification.

capture long-range motion trajectories across frames. These tracked points are then clustered using a momentum-encoder-based optimal transport mechanism, ensuring feature consistency while applying the loss only to visible points. By enforcing temporal coherence along motion trajectories, our model captures object permanence despite occlusions and viewpoint changes, resulting in more stable and semantically meaningful representations. Extensive evaluations across six image and video datasets demonstrate the effectiveness of MoSiC, showcasing its ability to learn robust and transferable dense features for tasks such as semantic segmentation and scene understanding. Notably, by leveraging the strong vision foundation model DINOv2 [42], MoSiC consistently improves its dense representations, as demonstrated qualitatively in Figure 2. This improvement is further validated quantitatively, achieving a 1% to 6% gain across four benchmarks—solely through training on video data.

2. Related Works

Video-to-Image self-supervised learning Video data provides rich temporal signals that naturally supervise learning by capturing object transformations and scene dynamics over time. Unlike static images, videos encode motion patterns and object persistence, offering essential cues for robust visual understanding. Early approaches explored various pretext tasks, including egomotion prediction [1, 26], pose estimation [13], dense prediction [37, 47], optical flow [39, 64], and visual correspondence learning [63].

Traditional self-supervised learning (SSL) methods, primarily developed on ImageNet, rely on augmentations for contrastive or predictive objectives [10, 14]. While extended to video [19, 45, 58, 62], direct transfer often leads to performance drops due to distribution shifts [45]. Recent works instead learn directly from videos, avoiding augmentation-based supervision. Carreira *et al.* [12] process continuous video streams without mini-batches, while DoRA [61] uses a single, long, unlabelled video to train strong image encoders without relying on traditional data augmentation techniques. MooG [60] models dynamic scene elements for improved motion tracking, and [11] scales SSL to larger models via

masked autoencoding for spatiotemporal learning.

Closest to our work, TimeTuning [50] leverages temporal cues but struggles with occlusions and long-range tracking under rapid camera motion. Unlike existing SSL methods, we incorporate dense point tracking to ensure robust temporal correspondences, improving performance on real-world video and image datasets.

Unsupervised object segmentation Unsupervised object segmentation has been widely studied, with many methods [2, 6, 22, 35, 53, 68] segmenting objects in images without labeled data. These approaches often use pretrained models to extract information and train another model for segmentation. Seitzer *et al.* [52] use slot attention to reconstruct DINO-pretrained features, clustering image regions into object slots. CroC [54] aligns cluster centers across views using a dense self-supervised loss, while Leopart [71] enhances representations with dense clustering. Hummingbird [5] leverages attention mechanisms within and across images for in-context scene understanding. CrIBo [36] enforces cross-image nearest-neighbor consistency, while NeCo [44] enforces patch-level nearest-neighbor consistency generating high-quality dense representations.

While these methods demonstrate the effectiveness of clustering and feature alignment for object segmentation, they primarily focus on static images, lacking the temporal consistency crucial for videos. In contrast, our method extends these ideas to the video domain by leveraging motion trajectories as signal. By propagating cluster assignments along tracked points, we ensure spatiotemporally coherent representations, bridging the gap between object-centric clustering and motion-aware learning. Tracktention [34] is another work that uses point trackers, which guides the attention for video prediction tasks to improve temporal consistency for depth. In contrast, we use point tracks to enforce temporal consistency for object segmentation across time.

3. MoSiC: Motion-Based Sinkhorn Clustering

We introduce MoSiC in Figure 3, a novel approach to dense self-supervised learning in videos that leverages motion as an implicit supervisory signal. Our method first extracts long-range motion trajectories using an off-the-shelf point tracker, ensuring robustness to occlusions and object permanence. To enforce spatiotemporal coherence, we perform clustering over these motion trajectories, aligning feature representations across frames. Unlike prior methods that rely purely on mask propagation or local frame-based learning, our approach establishes long-range correspondences, leading to spatiotemporally consistent features, considerably improving the quality of the learned representations.

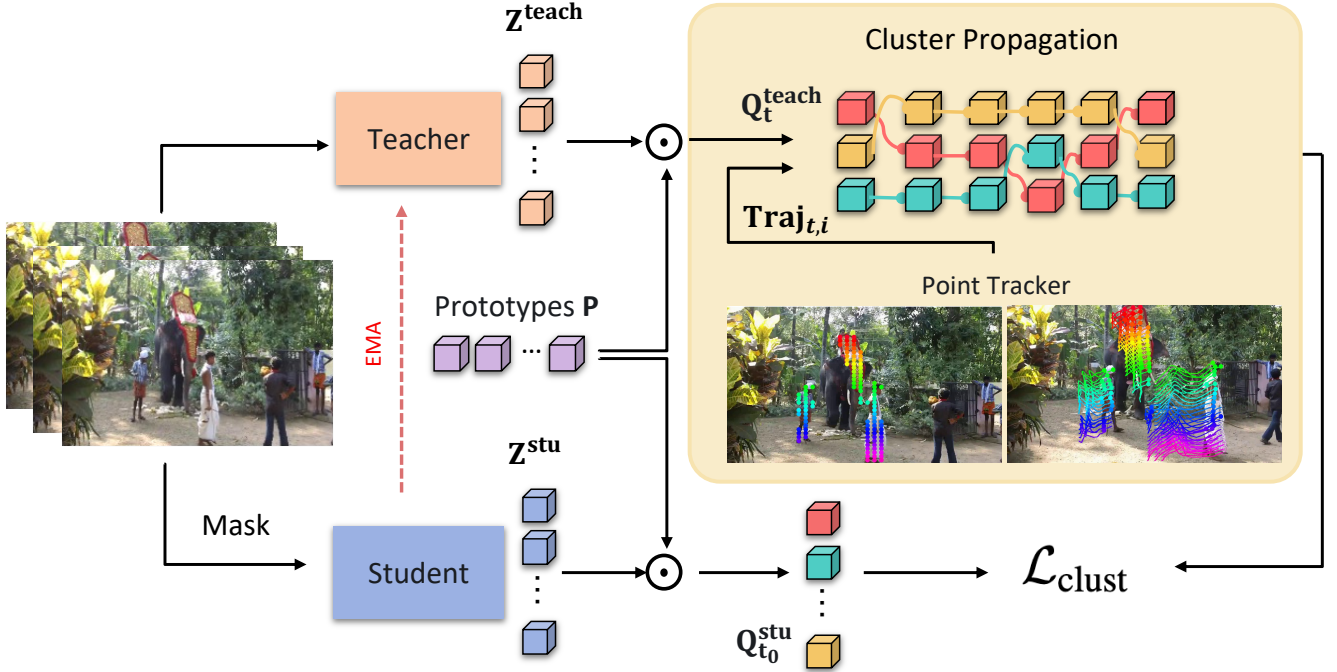


Figure 3. **MoSiC**: Motion-based Sinkhorn Clustering for dense self-supervised pretraining. A video clip is patchified, with masked patches passed through the student network to obtain $Z^{\text{stu}} \in \mathbb{R}^{n_s \times d \times T}$, while the teacher network processes the patches from raw clip to produce $Z^{\text{teach}} \in \mathbb{R}^{n \times d \times T}$. Features are clustered using Sinkhorn-Knopp [15] into prototypes P . The cluster assignments $Q_{t_0}^{\text{teach}}$ (eq. (6)) and $Q_{t_0}^{\text{stu}}$ (eq. (7)) are computed for the first frame ($t = t_0$). These assignments are then propagated along motion trajectories $\text{Traj}_{t,i}$ computed using CoTracker-v3 [29] to obtain Q_t^{teach} (eq. (8)). Finally, a cross-entropy loss is applied between $Q_{t_0}^{\text{stu}}$ and Q_t^{teach} (eq. (10)) for frames where $\text{Traj}_{t,i}$ remains visible, ensuring temporal consistency by clustering points along motion paths and preserving object identity over time.

3.1. Preliminaries

Given a video clip $X \in \mathbb{R}^{h \times w \times c \times T}$, where T is the number of frames, $h \times w$ is the spatial resolution, and c is the number of channels, we first divide each frame into $n = \frac{h \times w}{p^2}$ patches of size $p \times p$. These patches are then linearly projected into d -dimensional embeddings and passed through an encoder. We apply a binary mask $M \in \{0, 1\}^{n \times T}$ uniformly across all frames, randomly masking a fraction m of the tokens. Following Moutakanni *et al.* [41], we demonstrate that simple augmentations, such as cropping and masking, are sufficient for learning effective representations, eliminating the need for more complex augmentations like color jitter, grayscale, or blurring. We use a Vision Transformer (ViT) [17] in a teacher-student framework, where the student and teacher models are f_θ and g_θ . The representations are given by $Z^{\text{teach}} = g_\theta(X_t) \in \mathbb{R}^{n \times d}$ for the teacher and $Z^{\text{stu}} = f_\theta((1 - M) \odot X_t) \in \mathbb{R}^{n_s \times d}$ for the student networks, where \odot denotes the Hadamard product and n_s denotes the number of unmasked tokens.

3.2. Clustering Motion Trajectories

A major challenge in learning from real videos is the transient nature of objects due to occlusions, camera movement, and object permanence. Unlike images, video frames un-

dergo temporal deformations, making feature consistency difficult. Prior methods [50] propagate object masks across frames but struggle when objects temporarily disappear, leading to propagation errors. Additionally, long-range tracking suffers from drift accumulation, which compounds over time and degrades feature representations—an issue that worsens with longer sequences.

Motion Trajectories From a Point Tracker To address these challenges, we utilize an off-the-shelf point tracker capable of long-range tracking and robust to object permanence. The tracker samples points from a regular grid in the first frame and updates their positions across subsequent frames. Given our video clip X , we sample N points from the initial frame, forming a grid $= (x_i, y_i)_{i=1}^N$, where (x_i, y_i) are the coordinates of the i^{th} point. Using the video clip X and the grid, the tracker predicts the trajectories of these points across all frames as:

$$\text{Traj}_{t,i} := \text{Tracker}(X_t, (x_i, y_i)) \in \mathbb{R}^{T \times N \times 2}. \quad (1)$$

$\text{Traj}_{t,i}$ represents the trajectory of the i^{th} point across T frames, with each element encoding its coordinates at time t .

Optimal-Transport Based Clustering Although motion trajectories provide a strong signal for motion-based group-

ing, relying solely on them can be a challenge especially for objects with subtle or ambiguous motion. To address this, we perform clustering to enforce temporal coherence, ensuring that points belonging to the same object or part remain consistently clustered over time. Given motion trajectories from the point tracker, we incorporate semantic clustering to obtain a more discriminative representation of objects.

We first cluster the features from the student and teacher networks $Z^{\text{stu}} \in \mathbb{R}^{n_s \times d \times T}$, $Z^{\text{teach}} \in \mathbb{R}^{n \times d \times T}$ for the first frame using the Sinkhorn-Knopp algorithm [15]. This clustering solves an entropy-regularized optimal transport problem, iteratively refining the assignment of feature tokens to cluster prototypes while maintaining marginal constraints and has been used in various self-supervised works [3, 9, 42, 51].

Let $P^{\text{stu}}, P^{\text{teach}} \in \mathbb{R}^{K \times d}$ denote K cluster prototypes and $Z_{t_0}^{\text{stu}} \in \mathbb{R}^{n_s \times d}$, $Z_{t_0}^{\text{teach}} \in \mathbb{R}^{n \times d}$ represent the features extracted from the initial frame $t = t_0$. The transport cost between the patch features and the cluster prototypes $C^{\text{stu}} \in \mathbb{R}^{n_s \times K}$, $C^{\text{teach}} \in \mathbb{R}^{n \times K}$ are:

$$C^{\text{stu}} = -Z_{t_0}^{\text{stu}} P^{\text{stu}\top} \in \mathbb{R}^{n_s \times K} \quad (2)$$

$$C^{\text{teach}} = -Z_{t_0}^{\text{teach}} P^{\text{teach}\top} \in \mathbb{R}^{n \times K} \quad (3)$$

which denote the negative cosine similarity between the normalized patch features and normalized prototypes. The optimal transport plans $M^{\text{stu}*}$ and $M^{\text{teach}*}$ using eq. (2) and eq. (3) are obtained by solving:

$$M^{\text{stu}*} = \arg \min_{M^{\text{stu}} \in \mathcal{M}^{\text{stu}}} \langle M^{\text{stu}}, C^{\text{stu}} \rangle - \frac{\epsilon}{\lambda} H(M^{\text{stu}}) \quad (4)$$

$$M^{\text{teach}*} = \arg \min_{M^{\text{teach}} \in \mathcal{M}^{\text{teach}}} \langle M^{\text{teach}}, C^{\text{teach}} \rangle - \frac{\epsilon}{\lambda} H(M^{\text{teach}}) \quad (5)$$

where $M^{\text{stu}} \in \mathbb{R}^{n_s \times K}$, $M^{\text{teach}} \in \mathbb{R}^{n \times K}$ are the assignment matrices, $H(M)$ is an entropy regularization, and ϵ is the regularization coefficient that controls the smoothness of the assignment enforcing uniform marginal constraints $M^{\text{stu}} \mathbf{1}_K = \mathbf{1}_{n_s}/n_s$, $M^{\text{stu}\top} \mathbf{1}_{n_s} = \mathbf{1}_K/K$ and $M^{\text{teach}} \mathbf{1}_K = \mathbf{1}_n/n$, $M^{\text{teach}\top} \mathbf{1}_n = \mathbf{1}_K/K$.

3.3. Cluster Propagation using Motion Trajectories

After clustering the initial frame, we propagate these assignments across frames using the tracked motion trajectories from eq. (1). This allows us to maintain consistent object representations over time, even when objects undergo deformation. Given the tracked trajectories from eq. (1), we sample features from the student and teacher embeddings Z^{stu} , Z^{teach} at these locations.

To extract features at the tracked locations, we interpolate values from the feature maps using the continuous trajectory coordinates. Since these points do not always align with discrete pixel locations, we apply bilinear interpolation for the teacher network, computing feature values as a weighted sum of the four nearest neighbors. For the student network, we use nearest-neighbor interpolation, since the points are sampled along a simple uniformly spaced grid.

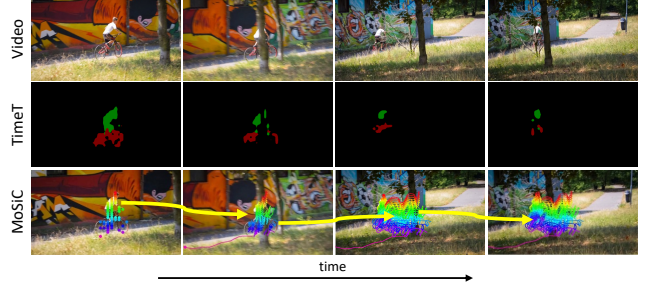


Figure 4. Previous methods such as TimeT lose precise object mapping after occlusions (e.g., a tree), rapid camera and object motion – resulting in degraded and coarse segmentation targets over long-range video clips. In contrast, MoSiC leverages stable point tracks, preserving consistency despite occlusions and motion via our trajectory based clustering loss eq. (10).

Cluster Propagation First, we compute the cluster assignment for the features from the teacher and the student networks using $M^{\text{stu}*}$ and $M^{\text{teach}*}$ computed in eq. (4) and eq. (5) as follows:

$$Q_{t_0}^{\text{stu}} = \arg \max_K M^{\text{stu}*} \cdot Z_{t_0}^{\text{stu}} \quad (6)$$

$$Q_{t_0}^{\text{teach}} = \arg \max_K M^{\text{teach}*} \cdot Z_{t_0}^{\text{teach}} \quad (7)$$

where the optimal transport plans assigns each feature to its nearest cluster prototype. We guide the cluster assignments $Q_{t_0}^{\text{teach}}$ using the motion trajectories $\text{Traj}_{t,i}$ obtained from eq. (1), where each tracked point retains its original cluster assignment as it moves along the trajectory:

$$Q_t^{\text{teach},i} = Q_{t_0}^{\text{teach},i}, \quad \forall (x_{i,t}, y_{i,t}) = \text{Traj}_{t,i} \quad (8)$$

3.4. Training Objective

To ensure temporal consistency, we aim to align features of different views of the same object over time. Although motion trajectories provide the correspondence of object locations across frames, relying solely on trackers may not guarantee that features from different views are assigned to the same cluster. For each frame $t \in T$, let $Q_{t_0}^{\text{stu},i}$ denote the cluster assignment for the i -th point in the initial frame t_0 from the student network, and $Q_t^{\text{teach},i}$ denote the propagated cluster assignment for the i -th point in frame t from the teacher network, as defined in eq (8).

We compute the cluster scores $S_{t_0}^{\text{stu},k,i}$ for each cluster k and point i in the frame t applying a softmax on the similarity between the $Z_{t_0}^{\text{stu},i}$ and the cluster prototypes $M^{\text{stu}*}$:

$$S_{t_0}^{\text{stu},k,i} = \frac{\exp(M^{\text{stu}*} \cdot Z_{t_0}^{\text{stu},i})}{\sum_{k'} \exp(M^{\text{stu}*} \cdot Z_{t_0}^{\text{stu},i})} \quad (9)$$

The teacher's assignments are one-hot vectors, where $\delta(Q_t^{\text{teach},i} = k)$ equals 1 if the assigned cluster is k and 0 otherwise. The clustering loss is formulated as a cross-entropy loss, averaged over all visible trajectories and frames:

$$\mathcal{L}_{\text{clust}}(i) = - \sum_{t=1}^T \sum_{k=1}^K v_{t,i} \cdot \delta(\mathcal{Q}_t^{\text{teach},i} = k) \cdot \log(S_{t_0}^{\text{stu},k,i}) \quad (10)$$

where v is a visibility flag indicating whether the i -th point is visible in frame t . This ensures that the loss is computed only for points that are visible in each frame, thereby enhancing the robustness of our model to occlusions. Therefore, as shown in Figure 4 we enforce that all patches along a trajectory, representing the same object or part as observed, are consistently clustered over time. This approach extends beyond individual images by maintaining consistency across pixel trajectories throughout the video sequence.

4. Experiments

4.1. Experimental setup

We follow TimeTuning [50] and initialize MoSiC with a pretrained DINOv2 [42] backbone and a frozen CoTracker-v3 [29] for point tracking. Training is conducted on YouTube-VOS [65], one of the largest video segmentation datasets. For evaluation, we discard the projection head and use the teacher network as the feature extractor, following DINO [10]. Unless specified, mean intersection over union (mIoU) serves as the primary metric.

We assess linear segmentation on Pascal VOC [18], COCO-Things [8], and ADE20K [69] by training a segmentation head on frozen spatial features, and perform end-to-end segmentation using the Segmenter head [55]. For frozen clustering-based evaluations, we apply K -Means to spatial tokens, setting K to both the number of ground-truth objects and larger values (300, 500), then match cluster maps to ground truth using Hungarian matching [33]. Dense nearest-neighbor retrieval is used to evaluate on-context scene understanding [5] for ADE20K and Pascal VOC. For unsupervised video semantic segmentation, we perform clustering and overclustering on YouTube-VOS and DAVIS [48], aligning clusters with ground truth via Hungarian matching.

We evaluate MoSiC across four dense benchmarks spanning images and videos, assessing unsupervised semantic segmentation across time and space, visual scene understanding, and feature transferability via linear segmentation. Additional experiments, including end-to-end finetuning results for object detection and segmentation, scaling to the large variant, image classification performance and pretraining with DINO, can be found in Appendix 6.1

4.2. Unsupervised Video Object Segmentation

To assess temporal coherence in MoSiC, we evaluate the clustering performance on DAVIS [48] and YouTube-VOS [65]. Following [50], clusters are aligned with ground-truth object masks using the Hungarian algorithm [33]. We also evaluate performance under Over-Clustering, where a

larger number of clusters is used. This is particularly relevant in dense self-supervised learning, as representations serve as feature descriptors for downstream tasks such as semantic segmentation and object detection.

From Table 2, MoSiC-S14 outperforms TimeT [50] by 8.7% and 9.4% mIoU on DAVIS and YouTube-VOS, respectively. Additionally, MoSiC surpasses both DINOv2 and DINOv2-R by 4.3% and 3.7% on average across clustering and overclustering evaluations. These results highlight MoSiC’s ability to produce temporally consistent segmentations and generalize across diverse video datasets.

4.3. Visual In-Context learning

We compare MoSiC on the Hummingbird Benchmark [5], which evaluates in-context reasoning in vision models. Unlike traditional segmentation approaches, this task does not involve fine-tuning; instead, segmentation maps are generated by matching patch-level feature similarities between validation (query) and training (key) images. As shown in Table 1, MoSiC consistently outperforms SOTA on Pascal-VOC [18] and ADE20K [69], particularly in low-data settings. On Pascal VOC, MoSiC-S14 outperforms DINOv2 by 6% and DINOv2-R by nearly 10% with 1/128th of the data and persists across all data regimes, with MoSiC-S14 surpassing DINOv2 by 1% even with full data. The larger variant, MoSiC-B14, further improves performance, outperforming DINOv2-R by 10% in the lowest-data setting and exceeding DINOv2 by 3% with full data.

Similarly, on ADE20K, MoSiC outperforms DINOv2 by 2% across both variants. These results demonstrate MoSiC’s ability to learn robust and transferable representations, especially in data-scarce scenarios where it significantly outperforms prior methods.

4.4. Frozen Clustering

We assess the quality of learned representations by clustering features and evaluating their ability to differentiate objects within datasets. Ideally, the patch features corresponding to the same object should be assigned to a single cluster. If representations capture finer details (e.g., hands or faces rather than entire persons), they should remain consistent across images. To analyze this, we extract dense features and apply K -Means clustering with varying K values. Cluster maps are then aligned with ground truth using Hungarian matching [33]. We evaluate performance when K matches the number of ground-truth objects, as well as under Over-Clustering, similar to subsection 4.2.

From Table 3a and Table 3b, MoSiC achieves the highest mIoU across all clustering granularities. For $K = 300$, it outperforms DINOv2 by 3.7% on Pascal VOC and 1.6% on COCO-Things, indicating its ability to learn structured, discriminative representations. Under Over-Clustering ($K = 500$), MoSiC surpasses CrIBo by 5.7% on Pascal VOC and 6.6% on COCO-Things, demonstrating its capacity to

Table 1. **In-context scene understanding benchmark.** We evaluate dense nearest neighbor retrieval performance across various training data proportions on ADE20K [69] and Pascal VOC [18]. Retrieved cluster maps are compared with the ground truth via Hungarian matching [33]. We report mIoU; higher is better.

METHOD	BACKBONE	PARAMS	ADE20K				PASCAL VOC			
			1/128	1/64	1/8	1/1	1/128	1/64	1/8	1/1
TRAINED ON IMAGES										
DINO [10]	ViT-S/16	21M	9.5	11.0	15.0	17.9	26.4	30.5	41.3	48.7
CrOC [54]	ViT-S/16	21M	8.7	10.8	15.2	17.3	34.0	41.8	53.8	60.5
SelfPatch [66]	ViT-S/16	21M	10.0	10.9	14.7	17.7	28.4	32.6	43.2	50.8
Leopart [71]	ViT-S/16	21M	12.9	14.8	19.6	23.9	44.6	49.7	58.4	64.5
CrIBo [36]	ViT-S/16	21M	14.6	17.3	22.7	26.6	53.9	59.9	66.9	72.4
DINOv2 [42]	ViT-S/14	21M	22.8	26.4	33.5	38.8	56.0	62.4	72.3	77.0
FINETUNED ON VIDEOS										
TimeT [50]	ViT-S/16	21M	12.1	14.1	18.9	23.2	38.1	43.8	55.2	62.3
MoSiC	ViT-S/14	21M	23.8	27.4	35.7	40.7	62.5	66.6	74.7	78.2
TRAINED ON IMAGES										
MAE [23]	ViT-B/16	85M	10.0	11.3	15.4	18.6	3.5	4.1	5.6	7.0
DINO [10]	ViT-B/16	85M	11.5	13.5	18.2	21.5	33.1	37.7	49.8	57.3
Leopart [71]	ViT-B/16	85M	14.6	16.8	21.8	26.7	50.1	54.7	63.1	69.5
Hummingbird [5]	ViT-B/16	85M	11.7	15.1	22.3	29.6	50.5	57.2	64.3	71.8
CrIBo [36]	ViT-B/16	85M	15.9	18.4	24.4	28.4	55.9	61.8	69.2	74.2
DINOv2 [42]	ViT-B/14	85M	24.2	27.6	34.7	39.9	55.7	61.8	72.4	77.1
FINETUNED ON VIDEOS										
MoSiC	ViT-B/14	85M	25.4	29.3	37.3	42.6	65.5	69.8	76.9	80.5

Table 2. **Unsupervised video semantic segmentation results** for clustering and over-clustering on DAVIS [48] and Youtube-VOS (YTVOS) [65]. For clustering, the Hungarian algorithm [33] matches clusters (K) to ground truth (GT) per frame (F), clip (C), or dataset (D). For over-clustering: K=10 (F, C), K=200 (D, DAVIS), K=500 (D, YTVOS). All the models are ViT-S16, except for DINOv2 variants and MoSiC, which are ViT-S14. We report mIoU and, unlike [50], do not use CBFE for any method to ensure consistency; higher is better.

	CLUSTERING						OVER-CLUSTERING					
	YTVOS			DAVIS			YTVOS			DAVIS		
	F	C	D	F	C	D	F	C	D	F	C	D
TRAINED ON IMAGES												
DINO [10]	39.1	37.9	1.9	30.2	31.0	1.6	66.2	65.4	4.0	56.9	54.9	17.9
Leopart [71]	39.2	37.9	11.7	30.3	30.2	16.5	64.5	62.8	15.5	54.9	54.4	26.7
DINOv2 [42]	56.3	55.5	12.8	57.4	57.4	13.6	62.8	62.5	17.3	57.9	58.5	25.5
DINOv2R [16]	56.8	55.4	14.7	57.3	57.8	14.3	64.0	63.5	21.5	58.1	59.5	27.2
FINETUNED ON VIDEOS												
STEGO [22]	41.5	40.3	2.0	31.9	31.0	3.2	58.1	54.3	5.1	47.6	46.3	10.4
DINO [10]	37.2	36.1	1.2	29.3	29.2	2.4	53.1	50.9	1.3	45.4	44.0	8.6
Leopart [71]	41.5	40.5	7.7	37.5	36.5	12.6	60.8	59.8	6.8	53.7	53.1	16.8
TimeT [50]	50.2	51.4	12.8	49.5	49.2	12.8	60.6	59.4	18.1	55.9	57.6	22.0
MoSiC	60.6	59.6	18.4	58.9	60.8	23.4	66.8	65.6	24.4	58.4	59.8	29.0

encode fine-grained object structures. These results show MoSiC’s effectiveness in distinguishing objects and their components, making it suitable for dense prediction tasks.

4.5. Linear Segmentation

We further evaluate MoSiC for supervised semantic segmentation, keeping the pretrained backbone frozen while training a linear layer on top. Output features are upsampled via bilinear interpolation to match input dimensions, allowing for pixel-wise cross-entropy loss. Unlike fine-tuning, this setup provides a more reliable measure of the backbone’s learned representations. As shown in Table 4, MoSiC consistently

outperforms SOTA segmentation models, including CrIBo, and surpasses DINOv2-R by up to 4% on Pascal VOC and 3.3% on ADE20K. Additionally, MoSiC outperforms DINOv2 across all datasets, underscoring its ability to capture semantically meaningful features that enable strong segmentation performance with just a linear probe.

4.6. Generalization to Diverse Vision Encoders

We explore the generalization of MoSiC to a diverse set of vision foundation models. We observe in Table 5, that MoSiC enhances both in-context scene understanding and linear classification for vision models, as well as vision-language

Table 3. **Frozen clustering-based evaluations.** (a) We evaluate the models using K -means with various clustering granularities K on the features of Pascal VOC [18] and COCO-Things [8]. Cluster maps are matched to the ground-truth via Hungarian matching [33], We report mIoU; higher is better. (b) We post-process these maps for unsupervised semantic segmentation on Pascal VOC. All the models are ViT-S16, except for DINOv2 variants and MoSiC, which are ViT-S14.

(a) Clustering							(b) Semantic segmentation	
METHOD	PASCAL VOC			COCO-THINGS			METHOD	mIoU
	K=100	K=300	K=500	K=100	K=300	K=500		
TRAINED ON IMAGES								
DINO [10]	10.1	13.9	17.3	14.4	18.8	19.2	MaskConstrast [59]	35.1
CrOC [54]	10.2	16.4	20.0	22.4	14.7	18.1	DINOv2R [16]	35.1
iBOT [70]	16.5	23.8	31.1	15.5	26.6	28.0	DINOv2 [42]	37.5
EVA-CLIP [56]	31.7	37.4	41.4	30.5	38.0	39.8	DeepSpectral [40]	37.2
DINOv2R [16]	34.8	46.7	49.5	32.0	38.9	41.2	DINOSAUR [52]	37.2
Leopart [71]	39.2	46.5	51.2	38.3	47.8	53.2	Leopart [71]	41.7
CrIBo [36]	40.3	51.3	54.5	40.2	46.0	48.3	COMUS [67]	50.0
DINOv2 [42]	43.2	55.1	58.6	43.8	51.6	53.1	FINETUNED ON VIDEOS	
FINETUNED ON VIDEOS								
TimeT [50] [†]	34.6	43.6	46.2	34.9	42.7	44.6	TimeT [50]	41.1
MoSiC	50.0	58.8	60.2	45.8	53.2	54.9	MoSiC	51.2

Table 4. **Linear segmentation performance.** A linear segmentation head is trained on top of frozen spatial features. We report mIoU scores on 4 validation datasets.

METHOD	ARCH.	COCO-THINGS	COCO-STUFF	PASCAL VOC	ADE20K
TRAINED ON IMAGES					
DINO [10]	ViT-S/16	43.9	45.9	50.2	17.5
iBOT [70]	ViT-S/16	58.9	51.5	66.1	21.8
CrOC [54]	ViT-S/16	64.3	51.2	67.4	23.1
CrIBo [36]	ViT-S/16	64.3	49.1	71.6	22.7
DINOv2 [42]	ViT-S/14	81.4	58.3	78.9	37.9
FINETUNED ON VIDEOS					
TimeT [50]	ViT-S/16	58.2	48.7	66.3	20.7
MoSiC	ViT-S/14	82.3	61.0	79.7	39.6
TRAINED ON IMAGES					
DINO [10]	ViT-B/16	55.8	51.2	62.7	23.6
MAE [23]	ViT-B/16	38.0	38.6	32.9	5.8
iBOT [70]	ViT-B/16	69.4	55.9	73.1	30.1
CrIBo [36]	ViT-B/16	69.6	53.0	73.9	25.7
EVA-CLIP [56]	ViT-B/16	75.9	48.0	70.4	34.6
DINOv2 [42]	ViT-B/14	84.0	58.9	80.3	42.6
FINETUNED ON VIDEOS					
MoSiC	ViT-B/14	85.8	61.4	81.5	43.6

models such as EVA-CLIP [56], achieving improvements of 2% and 3% on average, respectively, using only video-based training. This demonstrates that videos can serve as a complementary source of information to further enrich image and language datasets.

Table 5. **Performance on In-context visual scene understanding (IC) and linear classification (LC) benchmarks with and without MoSiC.** MoSiC improves various baselines across segmentation datasets.

METHOD	IC(I/I)		LC	
	ADE20K	PASCAL VOC	ADE20K	PASCAL VOC
DINO-S16	17.9	48.7	17.5	50.2
+ MoSiC	24.9	64.5	22.3	67.3
EVA-CLIP-B14 [56]	32.9	69.0	34.6	70.4
+ MoSiC	35.0	70.0	40.1	73.4
DINOv2R-B14	39.5	78.8	43.0	80.2
+ MoSiC	43.7	79.3	44.4	81.1

4.7. End-to-end Full Fine-tuning

In Table 6 and Table 10, we evaluate the performance of MoSiC when used as the backbone for object detection and semantic segmentation in the full finetuning setting. For object detection ViT-Det [38] and for semantic segmentation Segmenter [55] frameworks are used. As shown, our method consistently outperforms DINOv2 across all datasets and evaluation protocols, setting a new state-of-the-art—despite being fine-tuned solely on video data that differs in distribution from the evaluation datasets. Since full finetuning updates all model parameters, the superior performance of MoSiC indicates that learning a more structured, object-aware feature space is achievable even without any image dataset, by finetuning on videos alone.

Table 6. **MoSiC vs. DINOv2 on object detection.** We use ViTDet [38] on COCO [8] and report Average Precision (AP) on bounding-box object detection (AP^{box}) and instance segmentation (AP^{mask}).

Method	Backbone	Params	AP^{box}	AP^{mask}
DINOv2	ViT-S/14	21M	42.5	36.7
MoSiC	ViT-S/14	21M	42.5	36.8
DINOv2	ViT-B/14	85M	46.1	41.9
MoSiC	ViT-B/14	85M	46.4	42.0
DINOv2	ViT-L/14	307M	51.6	45.9
MoSiC	ViT-L/14	307M	51.8	46.0

4.8. Ablations

We conduct ablations on key parameters of MoSiC using the DINO [10] backbone trained on YouTube-VOS for 50 epochs with an image resolution of 224. We evaluate visual in-context learning on Pascal VOC12 and ADE20K, using a

Table 7. **Ablating the key parameters of MoSiC** by training a linear layer on top of the frozen representations (Lin.) or using the in-context (IC) evaluation [5] on the validation images of Pascal VOC (PVOC) and ADE20K.

(a) Mask Ratio			(b) EMA Teacher			(c) Grid Size			(d) Crop Scale		
RATIO	PVOC	ADE20K	TEACHER	PVOC	ADE20K	GRID	PVOC	ADE20K	INTERVAL	PVOC	ADE20K
No Mask	51.1	18.2	\times	50.5	18.2	8×8	49.2	17.5	No crop	49.1	18.0
10%	51.5	18.6	\checkmark	51.5	18.6	16×16	51.5	18.6	[0.2, 1]	50.8	18.5
20%	51.3	18.8				32×32	50.5	18.2	[0.4, 1]	51.5	18.6
40%	49.9	18.5							[0.6, 1]	50.5	18.4

(e) Number of Prototypes			(f) Optimized points			(g) Clip length in second			(h) Number of frames.		
PROTOTYPES	PVOC	ADE20K	BATCH	PVOC	ADE20K	T	PVOC	ADE20K	#FRAMES	PVOC	ADE20K
50	46.5	15.3	All	50.5	18.0	0.5	47.6	16.2	2	39.3	13.1
100	51.5	18.6	Visible	51.5	18.6	1	50.5	18.2	4	48.0	16.4
200	51.0	18.7				1.6	51.0	18.2	8	50.4	18.2
300	50.1	18.1				3.2	51.5	18.6	12	51.5	18.6

reduced memory size of 1/100 of the original setting in our primary experiments.

Masking ratio. We analyze the effect of different masking ratios in Table 7a. Moderate masking enhances semantic learning by preventing reliance on a limited set of patches. However, excessively high masking (e.g., 40%) degrades performance by reducing available input information. We select a 10% masking ratio due to its superior training stability over 20%, despite both achieving similar performance.

Crop scale. Table 7d evaluates the impact of different cropping scales in augmentations. Moderate cropping yields the highest improvements, particularly due to the non-object-centric nature of YouTube-VOS, where aggressive cropping can make it difficult to extract meaningful regions consistently across frames. This differs from static-image datasets [71].

EMA teacher. We assess the impact of an Exponential Moving Average (EMA) teacher in Table 7b. Incorporating a teacher network improves performance by 1% on Pascal VOC and 0.4% on ADE20K, aligning with findings from [10], where EMA stabilization led to improved training consistency in self-supervised frameworks.

Number of prototypes. Table 7e shows that increasing the number of prototypes significantly enhances performance up to a threshold. The best results are achieved with 100 prototypes, beyond which improvements plateau. MoSiC remains robust across a wide range (100–300 prototypes), indicating insensitivity to fine-tuning this parameter.

Grid size. Table 7c shows the effect of different grid sizes for tracking initialized points. Denser grids improve performance, peaking at 16×16 , after which performance slightly drops at 32×32 . This decline is likely due to an incompatibility between patch-based feature extraction and point-wise tracking, where excessive density leads to misleading correspondences during training.

Clip length. We evaluate the effect of clip length in Table 7g, considering sequences of 12 frames. Longer clips improve performance by exposing the model to richer motion cues and diverse object transformations. We select a 3.2-second clip length as the optimal setting for all

Number of clip frames. Table 7h explores the impact of clip frame count while keeping the time step per frame fixed at 3.2/12 seconds. Increasing the frame count consistently improves performance, as it provides richer spatial and temporal signals during training. We use 12 frames as the optimal setting.

Optimized points. To improve tracking accuracy, we apply loss only to points that remain visible throughout the sequence. As shown in Table 7f, this approach reduces false similarity enforcement, improving performance by up to 1

5. Conclusion

In this paper, we introduced MoSiC, a motion-based self-supervised learning approach that leverages dense point tracking to enforce temporal consistency in feature learning. By propagating cluster assignments along motion trajectories, our method effectively mitigates occlusions and long-range tracking drift, enhancing representation quality for dense prediction tasks. Extensive evaluations across unsupervised video-object segmentation, in-context learning, frozen clustering, and linear segmentation demonstrate that MoSiC achieves state-of-the-art performance across both image and video benchmarks. Our results highlight the importance of motion cues in self-supervised representation learning and suggest that point tracking can serve as a powerful supervisory signal—enabling robust feature learning without requiring dense annotations.

Acknowledgements

Shashanka was supported by HPC resources from GENCI-IDRIS Grant 2024-AD011016023.

References

- [1] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *ICCV*, 2015. 2
- [2] Nikita Araslanov, Simone Schaub-Meyer, and Stefan Roth. Dense unsupervised learning for video segmentation. *NeurIPS*, 2021. 2
- [3] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020. 4
- [4] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, 2023. 1
- [5] Ivana Balazevic, David Steiner, Nikhil Parthasarathy, Relja Arandjelović, and Olivier Henaff. Towards in-context scene understanding. *NeurIPS*, 2023. 2, 5, 6, 8, 14, 15
- [6] Zhipeng Bao, Pavel Tokmakov, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert. Object discovery from motion-guided tokens. In *ICCV*, 2023. 2
- [7] Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *TMLR*, 2024. 1
- [8] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocomp: Thing and stuff classes in context. In *CVPR*, 2018. 5, 7, 12, 13, 14, 16
- [9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *NeurIPS*, 33:9912–9924, 2020. 4
- [10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 2, 5, 6, 7, 8, 13, 14, 15
- [11] João Carreira, Dilara Gokay, Michael King, Chuhan Zhang, Ignacio Rocco, Aravindh Mahendran, Thomas Albert Keck, Joseph Heyward, Skanda Koppula, Etienne Pot, et al. Scaling 4d representations. *arXiv preprint arXiv:2412.15212*, 2024. 2
- [12] João Carreira, Michael King, Viorica Patraucean, Dilara Gokay, Catalin Ionescu, Yi Yang, Daniel Zoran, Joseph Heyward, Carl Doersch, Yusuf Aytar, et al. Learning from one continuous video stream. In *CVPR*, 2024. 2
- [13] Prabhuddha Chakraborty and Vinay P Namboodiri. Learning to estimate pose by watching videos. *arXiv preprint arXiv:1704.04081*, 2017. 2
- [14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2
- [15] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *NeurIPS*, 2013. 3, 4
- [16] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *ICLR*, 2024. 6, 7, 14, 15
- [17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3, 13
- [18] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>, 2012. 5, 6, 7, 14, 15, 16
- [19] Daniel Gordon, Kiana Ehsani, Dieter Fox, and Ali Farhadi. Watching the world go by: Representation learning from unlabeled videos. *arXiv preprint arXiv:2003.07990*, 2020. 2
- [20] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent—a new approach to self-supervised learning. *NeurIPS*, 2020. 13
- [21] Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. Accelerating large-scale inference with anisotropic vector quantization. In *ICML*, 2020. 15
- [22] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snaveley, and William T Freeman. Stego: Unsupervised semantic segmentation by distilling feature correspondences. In *ICLR*, 2022. 2, 6, 15
- [23] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 6, 7, 14, 15
- [24] Olivier J Hénaff, Skanda Koppula, Jean-Baptiste Alayrac, Aaron Van den Oord, Oriol Vinyals, and Joao Carreira. Efficient visual pretraining with contrastive detection. In *ICCV*, 2021. 1
- [25] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 13
- [26] Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *ICCV*, 2015. 2
- [27] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *ICCV*, 2019. 16
- [28] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 2019. 16
- [29] Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker3: Simpler and better point tracking by pseudo-labelling real videos. *arXiv preprint arXiv:2410.11831*, 2024. 3, 5
- [30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014. 13
- [31] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 16
- [32] Kurt Koffka. *Principles of Gestalt psychology*. routledge, 2013. 1
- [33] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955. 5, 6, 7, 14, 15, 16

- [34] Zihang Lai and Andrea Vedaldi. Tracktention: Leveraging point tracking to attend videos faster and better. *arXiv preprint arXiv:2503.19904*, 2025. 2
- [35] Mengcheng Lan, Xinjiang Wang, Yiping Ke, Jiaying Xu, Litong Feng, and Wayne Zhang. Smooseg: smoothness prior for unsupervised semantic segmentation. *NeurIPS*, 2023. 2
- [36] Tim LeBailly, Thomas Stegmüller, Behzad Bozorgtabar, Jean-Philippe Thiran, and Tinne Tuytelaars. Cribot: Self-supervised learning via cross-image object-level bootstrapping. In *ICLR*, 2024. 2, 6, 7, 14, 15
- [37] Xueting Li, Sifei Liu, Shalini De Mello, Xiaolong Wang, Jan Kautz, and Ming-Hsuan Yang. Joint-task self-supervised learning for temporal correspondence. In *NeurIPS*, 2019. 2
- [38] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *ECCV*, 2022. 7, 12
- [39] Aravindh Mahendran, James Thewlis, and Andrea Vedaldi. Cross pixel optical-flow similarity for self-supervised learning. In *ACCV*, 2019. 2
- [40] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Deep spectral methods: A surprisingly strong baseline for unsupervised semantic segmentation and localization. In *CVPR*, 2022. 7, 14
- [41] Théo Moutakanni, Maxime Oquab, Marc Szafraniec, Maria Vakalopoulou, and Piotr Bojanowski. You don't need domain-specific data augmentations when scaling self-supervised learning. *NeurIPS*, 2025. 3
- [42] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *TMLR*, 2024. 2, 4, 5, 6, 7, 14, 15
- [43] Valentinos Pariza, Mohammadreza Salehi, and Yuki Asano. Hummingbird evaluation for vision encoders, 2024. 15
- [44] Valentinos Pariza, Mohammadreza Salehi, Gertjan J. Burghouts, Francesco Locatello, and Yuki M Asano. Near, far: Patch-ordering enhances vision foundation models' scene understanding. In *The Thirteenth International Conference on Learning Representations*, 2025. 2
- [45] Nikhil Parthasarathy, SM Eslami, João Carreira, and Olivier J Hénaff. Self-supervised video pretraining yields human-aligned visual representations. In *NeurIPS*, 2023. 2
- [46] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019. 12
- [47] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, 2017. 2
- [48] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 5, 6, 15, 17
- [49] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 13
- [50] Mohammadreza Salehi, Efstratios Gavves, Cees GM Snoek, and Yuki M Asano. Time does tell: Self-supervised time-tuning of dense image representations. In *ICCV*, 2023. 1, 2, 3, 5, 6, 7, 14, 15, 20
- [51] Mohammadreza Salehi, Michael Dorckenwald, Fida Mohammad Thoker, Efstratios Gavves, Cees GM Snoek, and Yuki M Asano. Sigma: Sinkhorn-guided masked video modeling. In *ECCV*, 2024. 4
- [52] Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, et al. Bridging the gap to real-world object-centric learning. In *ICLR*, 2023. 2, 7, 14
- [53] Oriane Siméoni, Chloé Sekkat, Gilles Puy, Antonín Vobecký, Éloi Zablocki, and Patrick Pérez. Unsupervised object localization: Observing the background to discover objects. In *CVPR*, 2023. 2
- [54] Thomas Stegmüller, Tim LeBailly, Behzad Bozorgtabar, Tinne Tuytelaars, and Jean-Philippe Thiran. Croc: Cross-view online clustering for dense visual representation learning. In *CVPR*, 2023. 2, 6, 7, 14, 15
- [55] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *ICCV*, 2021. 5, 7, 12, 13
- [56] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 7, 14, 15
- [57] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *NeurIPS*, 2022. 1
- [58] Michael Tschannen, Josip Djolonga, Marvin Ritter, Aravindh Mahendran, Neil Houlsby, Sylvain Gelly, and Mario Lucic. Self-supervised learning of video-induced visual invariances. In *CVPR*, 2020. 2
- [59] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *ICCV*, 2021. 1, 7, 14, 16
- [60] Sjoerd van Steenkiste, Daniel Zoran, Yi Yang, Yulia Rubanova, Rishabh Kabra, Carl Doersch, Dilara Gokay, Etienne Pot, Klaus Greff, Drew Hudson, et al. Moving off-the-grid: Scene-grounded video representations. *NeurIPS*, 2025. 2
- [61] Shashanka Venkataramanan, Mamshad Nayeem Rizve, João Carreira, Yuki M Asano, and Yannis Avrithis. Is imagenet worth 1 video? learning strong image encoders from 1 long unlabelled video. In *ICLR*, 2024. 1, 2
- [62] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, 2015. 2
- [63] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, 2019. 2
- [64] Yuwen Xiong, Mengye Ren, Wenyan Zeng, and Raquel Urtasun. Self-supervised representation learning from flow equivariance. In *ICCV*, 2021. 2
- [65] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos:

- A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018. [5](#), [6](#), [15](#), [17](#)
- [66] Sukmin Yun, Hankook Lee, Jaehyung Kim, and Jinwoo Shin. Patch-level representation learning for self-supervised vision transformers. In *CVPR*, 2022. [6](#), [14](#)
- [67] Andrii Zadaianchuk, Matthaeus Kleindessner, Yi Zhu, Francesco Locatello, and Thomas Brox. Unsupervised semantic segmentation with self-supervised object-centric representations. In *ICLR*, 2023. [7](#), [14](#)
- [68] Andrii Zadaianchuk, Maximilian Seitzer, and Georg Martius. Object-centric learning for real-world videos by predicting temporal feature similarities. *NeurIPS*, 2023. [2](#)
- [69] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. [5](#), [6](#), [14](#), [15](#), [16](#), [17](#)
- [70] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. In *ICLR*, 2022. [7](#), [14](#), [15](#)
- [71] Adrian Ziegler and Yuki M Asano. Self-supervised learning of object parts for semantic segmentation. In *CVPR*, 2022. [1](#), [2](#), [6](#), [7](#), [8](#), [14](#), [15](#), [16](#)