University of Liège
School of Engineering
Montefiore Institute

LEARNING WITH UNLABELED DATA

a PhD dissertation
by Renaud Vandeghen

Advisor:
Professor Marc Van Droogenbroeck
February 2026

This dissertation has been submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy in Engineering Sciences and Technology.

JURY MEMBERS

Vân Anh Huynh-Thu, Professor, Université de Liège, Belgium (President).

Marc Van Droogenbroeck, Professor, Université de Liège, Belgium (Advisor).

Gilles Louppe, Professor, Université de Liège, Belgium.

Christophe De Vleeschouwer, Professor, Université catholique de Louvain, Belgium.

Fida M. Thoker, Doctor, King Abdullah University of Science and Technology, Saudi Arabia.

ABSTRACT

Deep supervised learning has drastically improved many computer vision tasks but relies on large amounts of labeled data. In dense vision tasks such as object detection and semantic segmentation, labeled data is scarce and expensive to obtain. However, unlabeled data is often abundant and cheap to collect. The first part of this work focuses on the *semi-supervised learning* paradigm, leveraging unlabeled data to improve the performance of models trained with only a small amount of labeled data. The second part focuses on a more constrained setting, *self-supervised learning*, where no labeled data is available during pretraining.

In the first part, we explore semi-supervised learning techniques for object detection, using the pseudo-labeling paradigm. The first contribution explores how to account for the uncertainty of pseudo-labels during training. In particular, we propose to scale the loss contribution of each pseudo-labeled example by a factor related to its confidence score. We show that linearly scaling the loss by the confidence score is the most effective strategy compared to the baseline model, especially in very low-label regimes. Then we explore how to obtain a robust threshold value to select pseudo-labels without the need for a costly hyperparameter search. We propose to use an adaptive thresholding strategy, where the threshold is determined by the distribution of confidence scores. We show that this heuristic can be used for each class independently and that it matches the performance of the greedy threshold at no computational cost. This new thresholding strategy is therefore particularly useful since it can be applied across different domains. We also add a refinement stage in the teacher-student framework, where the student model is fine-tuned on the labeled data only before being used as a new teacher.

In the second part, we explore self-supervised learning in the context of masked image modeling. In the first work, we show that reconstructing a highly masked, randomly resized crop of an image is an effective pretraining task for object-centric representation learning. In particular, we show that this new pretraining strategy based on crops yields better performance than learning on video frames while also being more computationally efficient. Finally, in our last contribution, we improve video representation learning by combining masked video modeling with both pixel and trajectory signals. This dual reconstruction task encourages the model to learn both spatial and temporal information. During pretraining, the learning objective of the model is to reconstruct both masked spatial information, either in pixel or latent space, as well as the masked trajectory, obtained by an off-the-shelf point tracker. The trajectory information is also used to build a motion-aware masking strategy, which further improves the learned representations. We show that both signals are complementary and that their combination leads to state-of-the-art results, especially in motion-centric tasks.

ACKNOWLEDGMENTS

Acknowledgments will be available in the final version of the manuscript.

CONTENTS

1 INTRODUCTION 3
1.1 Motivation . . . 3
1.2 Research question . . . 4
1.3 Outline . . . 4
1.4 Contributions . . . 4

I SEMI-SUPERVISED LEARNING 9
2 LEARNING PARADIGMS 11
2.1 Supervised learning . . . 11
2.2 Semi-supervised learning . . . 12
2.2.1 Pseudo-labeling in classification . . . 14
2.2.2 Pseudo-labeling in object detection . . . 17
3 UNCERTAINTY IN PSEUDO-LABELS 21
3.1 Prologue . . . 21
3.2 Epilogue . . . 33
4 AUTOMATIC THRESHOLD SELECTION FOR PSEUDO-LABELING 35
4.1 Prologue . . . 35
4.2 Epilogue . . . 48

II SELF-SUPERVISED LEARNING 51
5 MASKED MODELING 53
5.1 Masked image modeling . . . 53
5.2 Masked video modeling . . . 54
6 OBJECT-CENTRIC REPRESENTATION LEARNING WITH MASKED IMAGE MODELING 57
6.1 Prologue . . . 57
6.2 Epilogue . . . 77
7 MOTION-CENTRIC REPRESENTATION LEARNING WITH MASKED VIDEO MODELING 79
7.1 Prologue . . . 79
7.2 Epilogue . . . 98

III CONCLUSION 99
8 CONCLUSION 101

"The beautiful thing about learning is that no one can take it away from you."

B.B. King

1 INTRODUCTION

1.1 MOTIVATION

Learning is a natural process, yet it is difficult to define precisely. A common functional view is that learning is an enduring change in behavior or knowledge caused by experience (Bransford et al. [2000]; De Houwer et al. [2013]). Experience can take many forms, *e.g.* observation, practice, instruction, and feedback.

Evidence from human development highlights complementary mechanisms. First, learners can extract structure directly from raw sensory streams. In controlled experiments, 8-month-old infants segmented continuous speech using statistical relations between syllables, even with brief exposure (Saffran et al. [1996]). Second, learning is active and constructive: early intelligence develops through action and progressive organization of sensorimotor schemes (Piaget [1952]). Third, learning is social: interaction, language, and guidance shape what is learned and when it becomes possible, as emphasized in Vygotsky [1978]. Social learning theory similarly mentions that people learn by observing others and the consequences of their actions (Bandura [1977]).

Taken together, these perspectives suggest that explicit instruction is only one component of learning. A substantial part of learning relies on extracting regularities from unlabeled experience, integrating new observations with prior knowledge, and forming abstractions that support generalization and transfer (Bransford et al. [2000]; Tenenbaum et al. [2011]).

Machine learning (ML) pursues an analogous objective through explicit computational procedures. A learning algorithm receives data and an objective, then adapts model parameters to improve performance (Bishop [2006]; Hastie et al. [2009]; Shalev-Shwartz and Ben-David [2014]).

In *supervised learning*, training data consist of input-output examples, and the model learns to predict targets that generalize to unseen samples (Bishop [2006]; Shalev-Shwartz and Ben-David [2014]). This setting is close to the explicitly guided component of human learning, where targets are provided by instruction or correction.

Supervised learning has driven major progress because the learning signal is direct and evaluation against ground truth is straightforward. Its practical limitation is that labels are costly, whereas real-world observations are abundant and mostly unlabeled.

This mismatch motivates paradigms that use unlabeled data more directly. In *semi-supervised learning*, a small labeled set is combined with a larger unlabeled set. In *self-supervised learning*, training targets are constructed from the data itself. At a high level,

these paradigms leverage the same intuition suggested by human learning evidence: useful structure can be learned from experience even when explicit targets are limited.

1.2 RESEARCH QUESTION

The motivation above highlights a central challenge. In many real-world settings, data are abundant, but explicit supervision is limited. This limitation is especially common when high-quality annotations are costly, slow to produce, or require expert knowledge. Consequently, progress is constrained not only by modeling choices and computation, but also by the availability, cost, and biases of labeled data.

This thesis is centered on one question:

How can we leverage unlabeled data to learn efficiently?

The contributions of this thesis provide concrete methods to address this question, with a focus on semi-supervised learning for object detection and self-supervised learning for images and video model pretraining.

1.3 OUTLINE

The manuscript is structured in two main technical parts, corresponding to the two learning paradigms covered in this thesis.

Part I introduces supervised and semi-supervised learning in Chapter 2. Chapters 3 and 4 each include a prologue, the associated manuscript, and an epilogue.

Part II covers self-supervised learning. Chapter 5 introduces masked modeling for images and videos. Chapters 6 and 7 follow the same structure as in the previous part.

Finally, Part III provides a general conclusion in Chapter 8.

1.4 CONTRIBUTIONS

Publications reproduced in this thesis

- **Renaud Vandeghen**[*], Anthony Cioppa[*], and Marc Van Droogenbroeck. *Semi-Supervised Training to Improve Player and Ball Detection in Soccer.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop (CVPRW), pages 3480-3489, June 2022. https://doi.org/10.1109/CVPRW56347.2022.00392

- **Renaud Vandeghen**, Gilles Louppe, and Marc Van Droogenbroeck. *Adaptive Self-Training for Object Detection.* In IEEE/CVF International Conference on

Computer Vision Workshops (ICCVW), pages 914-923, October 2023. https://doi.org/10.1109/ICCVW60793.2023.00098

- Alexandre Eymaël*, **Renaud Vandeghen***, Anthony Cioppa, Silvio Giancola, Bernard Ghanem, and Marc Van Droogenbroeck. *Efficient Image Pre-training with Siamese Cropped Masked Autoencoders.* In European Conference on Computer Vision (ECCV), pages 348-366, October 2024. https://doi.org/10.1007/978-3-031-73337-6_20

- **Renaud Vandeghen***, Fida Mohammad Thoker*, Bernard Ghanem, and Marc Van Droogenbroeck. *TrackMAE: Video Representation Learning via Track Mask and Predict.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2026.

Publications not reproduced in this thesis

- Sébastien Piérard, Anthony Cioppa, Anaïs Halin, **Renaud Vandeghen**, Maxime Zanella, Benoît Macq, Saïd Mahmoudi, and Marc Van Droogenbroeck. *Mixture Domain Adaptation to Improve Semantic Segmentation in Real-World Surveillance.* In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW), pages 22-31, January 2023. https://doi.org/10.1109/WACVW58289.2023.00007

- Anaïs Halin, Sébastien Piérard, **Renaud Vandeghen**, Benoît Gérin, Maxime Zanella, Martin Colot, Jan Held, *et al.*. *Physically Interpretable Probabilistic Domain Characterization.* In Proceedings of the Asian Conference on Computer Vision Workshops (ACCVW), pages 17-35, 2024. https://doi.org/10.1007/978-981-96-2641-0_2

- Jan Held*, **Renaud Vandeghen***, Adrien Deliège, Abdullah Hamdi, Silvio Giancola, Anthony Cioppa, *et al.*. *Triangle Splatting for Real-Time Radiance Field Rendering.* In Proceedings of the International Conference on 3D Vision (3DV), March 2026.

- Jan Held*, **Renaud Vandeghen***, Abdullah Hamdi*, Adrien Deliège, Anthony Cioppa, Silvio Giancola, Andrea Vedaldi, Bernard Ghanem, and Marc Van Droogenbroeck. *3D Convex Splatting: Radiance Field Rendering with 3D Smooth Convexes.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 21360-21369, June 2025. https://doi.org/10.1109/CVPR52734.2025.01990

- Jan Held, **Renaud Vandeghen**, Sanghyun Son, Daniel Rebain, Matheus Gadelha, Yi Zhou, Ming C. Lin, *et al.*. *Triangle Splatting+: Differentiable*

Rendering with Opaque Triangles. Preprint, 2025.
https://doi.org/10.48550/arXiv.2509.25122

- Jan Held, Sanghyun Son, **Renaud Vandeghen**, Daniel Rebain, Matheus Gadelha, Yi Zhou, Anthony Cioppa, *et al.*. *MeshSplatting: Differentiable Rendering with Opaque Meshes.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2026.

*Equal contribution.

"Tell me and I forget, teach me and I may remember, involve me and I learn."

Xunzi

Part I

SEMI-SUPERVISED LEARNING

2

LEARNING PARADIGMS

Outline

This chapter introduces the background necessary to understand supervised and semi-supervised learning. Section 2.1 summarizes supervised learning as the reference setting. Section 2.2 motivates semi-supervised learning and details pseudo-labeling for image classification and object detection.
We assume that the reader is familiar with the basics of machine learning, deep learning, and computer vision.

2.1 SUPERVISED LEARNING

Supervised learning remains the dominant paradigm for training deep models in computer vision, enabled by large-scale annotated datasets for image classification (Russakovsky et al. [2015]), object detection Lin et al. [2014], semantic segmentation (Cordts et al. [2016]), and action recognition (Carreira and Zisserman [2017]). Formally, given labeled pairs $(\boldsymbol{x}_i, y_i)$, the objective is to learn a parameterized model f_θ that maps $\boldsymbol{x}$ to y by minimizing a task-specific loss over the training set. Beyond fitting the labeled data, the central requirement is that the learned model generalizes to unseen samples drawn from the same data distribution.

In this thesis, the input is typically an image or a video, $\boldsymbol{x} \in \mathbb{R}^{H\times W\times T\times C}$, where H, W, T, and C denote height, width, temporal length (omitted for still images), and channels, respectively. Depending on the task, the output can be a class label for image classification Deng et al. [2009]; Krizhevsky [2009]; Russakovsky et al. [2015], a set of bounding boxes and class labels for object detection (Everingham et al. [2010]; Lin et al. [2014]), a pixel-level class assignment for semantic segmentation (Cordts et al. [2016]; Zhou et al. [2018a]), or an action label for video clips (Carreira and Zisserman [2017]; Goyal et al. [2017b]).

Across these tasks, performance is measured with standard task-specific metrics. For classification tasks, we report accuracy. For object detection, we use mAP with $\text{AP}_{50:95}$, *i.e.* average precision first averaged over IoU thresholds from 0.50 to 0.95, then averaged over classes. For semantic segmentation, we report mIoU, the mean intersection-over-union across classes. For video object segmentation and propagation, we use the DAVIS-style J&F score (Pont-Tuset et al. [2017]), defined as the average of region similarity J_m (IoU-based overlap) and contour accuracy F_m (boundary F-measure). For pose propa-

gation, we report PCK (percentage of correct keypoints), where different thresholds are used depending on the evaluation setup, *e.g.* PCK@0.1 and PCK@0.2. In terms of usage across this thesis, mAP is the main metric in Chapters 3 and 4 for semi-supervised object detection, while J&F, J_m, F_m, mIoU, and PCK are used in Chapter 6 for propagation benchmarks. Accuracy is primarily reported in Chapter 7 for action recognition evaluation.

As an introduction to supervised learning and to motivate the semi-supervised paradigm, we consider a standard image classification task on the CIFAR-10 dataset (Krizhevsky [2009]). CIFAR-10 consists of natural images distributed over ten semantic classes and has been extensively used to benchmark methods in computer vision.

In this experiment, we perform conventional supervised training on the training split and report performance on the test set, following best practice. To explicitly assess the dependency of supervised learning on labeled data, we repeat the experiment multiple times while progressively reducing the number of labeled training samples. Figure 1 presents the test accuracy as a function of the proportion of labeled data. As expected, performance degrades monotonically as the amount of supervision decreases, highlighting the strong reliance of supervised models on large annotated datasets.

Although this experimental setup is intentionally simplified, it captures a fundamental limitation of supervised learning. Its effectiveness is tightly coupled to the availability of labeled data. In many computer vision scenarios, acquiring large-scale annotations is costly and time-consuming. This is particularly evident in dense prediction tasks such as semantic segmentation on datasets like Cityscapes (Cordts et al. [2016]) or Microsoft COCO (Lin et al. [2014]), where pixel-accurate or instance-level labels require substantial human effort. The challenge is further amplified in specialized domains such as medical imaging (Jha et al. [2019]) or remote sensing (Li et al. [2020a]). These considerations motivate learning paradigms that leverage unlabeled data to reduce the dependency on extensive manual annotation.

2.2 SEMI-SUPERVISED LEARNING

In many practical settings, unlabeled images are abundant, while labeled data are limited. Semi-supervised learning addresses this imbalance by combining a small labeled set with a larger unlabeled set to improve generalization compared with supervised training on labeled data only.

In computer vision, semi-supervised learning has been developed primarily on image classification. Two major families dominate the literature: *consistency regularization* (Bachman et al. [2014]; Berthelot et al. [2019, 2020]; Laine and Aila [2017]; Sohn et al. [2020a]; Xie et al. [2020a]; Zhai et al. [2019b]) and *self-training* (Amini et al. [2025]; Chen et al. [2023a]; Lee [2013]; Pham et al. [2021]; Sohn et al. [2020a]; Wang et al. [2023c]; Xie et al. [2020b]; Zhang et al. [2021]). Consistency-based regularization methods encour-

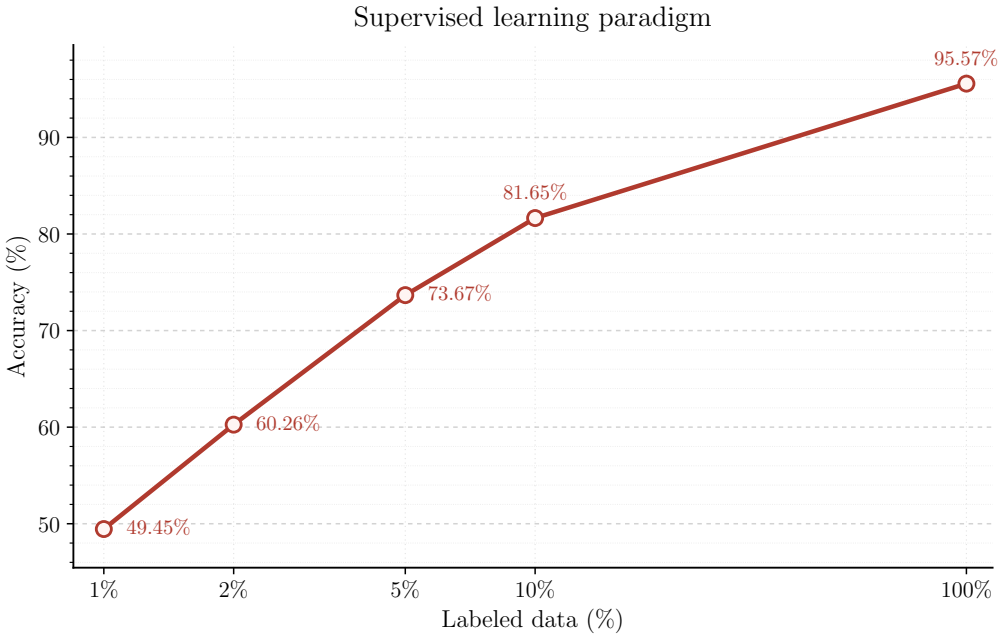


Figure 1: **Supervised learning**. CIFAR-10 test accuracy of a supervised ResNet-18 (He et al. [2016]) as a function of labeled-data availability. Performance increases monotonically from 49.45% at 1% labels to 95.57% at 100% labels, quantifying the strong dependence of supervised learning on annotation volume.

age models to produce stable predictions under input or model perturbations. In contrast, self-training leverages the model itself to generate a pseudo-supervision signal by converting predictions on unlabeled data into training targets. The latter family of approaches is examined in detail in the remainder of this section.

A general problem statement about semi-supervised learning is defined hereafter.

Problem statement

Let $\mathcal{D}_l = \{(\boldsymbol{x}_i^l, y_i^l)\}_{i=1}^{N_l}$ be a labeled dataset and $\mathcal{D}_u = \{\boldsymbol{x}_j^u\}_{j=1}^{N_u}$ an unlabeled dataset, with $N_u \gg N_l$. Semi-supervised learning optimizes a supervised loss term on $\mathcal{D}_l$ and an unsupervised loss term on $\mathcal{D}_u$, weighted by a λ_u factor:

$$\min_{\theta} \frac{1}{N_l} \sum_{i=1}^{N_l} \ell_{\text{sup}}(f_\theta(\boldsymbol{x}_i^l), y_i^l) + \lambda_u \frac{1}{N_u} \sum_{j=1}^{N_u} \ell_{\text{unsup}}(\boldsymbol{x}_j^u; \theta) \ .$$

In this thesis, we assume that labeled and unlabeled samples are drawn from the same data distribution.

2.2.1 *Pseudo-labeling in classification*

In this thesis, we focus on self-training through *pseudo-labeling*. An introduction to this paradigm is first presented in the context of image classification in the remainder of this section.

Given an unlabeled sample $\boldsymbol{x}^u$, a model produces an output distribution $p(c \mid \boldsymbol{x}^u)$ over class indices $c \in \mathcal{C}$. A pseudo-label is then formed as a target-confidence pair, typically

$$\hat{y}^u = \arg\max_c p(c \mid \boldsymbol{x}^u), \qquad s = \max_c p(c \mid \boldsymbol{x}^u) \ ,$$

where $\hat{y}^u$ is the selected class, c is the class index and s is its confidence score.

A standard semi-supervised training pipeline is:

1. Train a *teacher* model on labeled data.
2. Use the teacher on unlabeled data to generate pseudo-labels and associated confidence scores.
3. Train a *student* model on labeled and pseudo-labeled data.
4. Optionally replace the teacher with the student and iterate.

Implementation details vary across methods, but the central mechanism is unchanged: unlabeled samples are converted into training targets, and target quality improves as the teacher improves.

Two broad pseudo-labeling regimes are used in the literature: *offline* (Pham et al. [2021]; Xie et al. [2020b]) and *online* (Chen et al. [2023a]; Lee [2013]; Wang et al. [2023c]; Zhang et al. [2021]).

> *Offline vs online pseudo-labeling*
>
> In offline pseudo-labeling (Fig. 2), the teacher first generates pseudo-labels for the entire unlabeled dataset in a separate inference stage. The student is then trained on the union of labeled and pseudo-labeled data. After convergence, the student can replace the teacher, and the procedure may be repeated in successive rounds.
> In online pseudo-labeling (Fig. 3), pseudo-labels are produced on-the-fly during training. The teacher is typically updated as an exponential moving average of the student parameters. Pseudo-label generation and student optimization therefore occur simultaneously within a single training process, and the final student model is used for inference.
> Practically, the offline setting splits pseudo-label generation from optimization and is conceptually simple to implement, whereas the online setting is usually more compute-efficient and allows pseudo-labels to evolve continuously with the model.

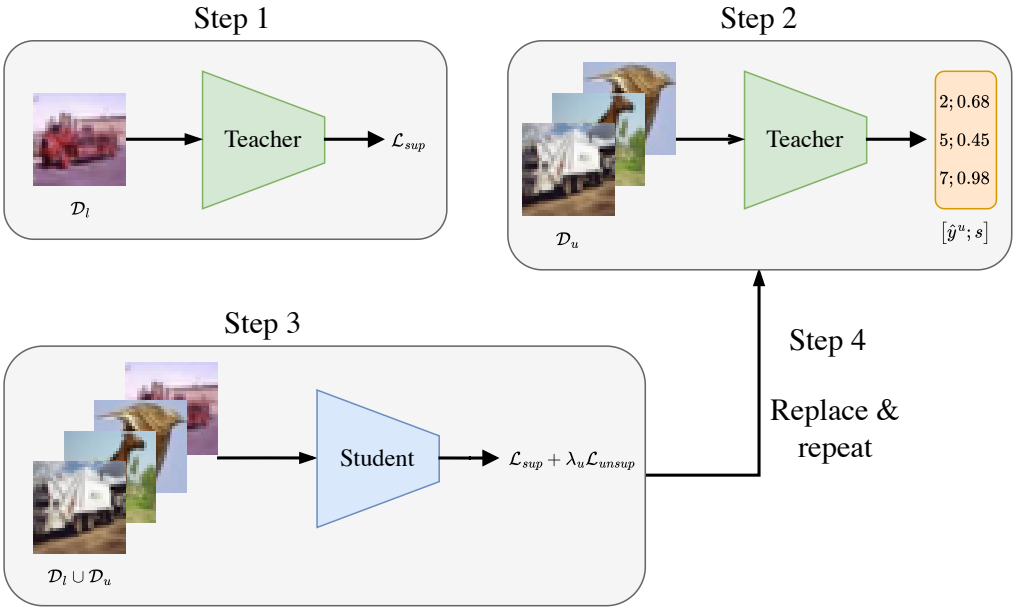


Figure 2: **Offline pseudo-labeling pipeline**. (Step 1) A teacher model is trained on labeled data $\mathcal{D}_l$ with a supervised loss $\mathcal{L}_{\text{sup}}$. (Step 2) The teacher infers pseudo-labels on the unlabeled set $\mathcal{D}_u$ in a separate inference stage. (Step 3) A student is trained on $\mathcal{D}_l \cup \mathcal{D}_u$ using labeled and pseudo-labeled supervision $\mathcal{L}_{\text{sup}} + \lambda_u \mathcal{L}_{\text{unsup}}$. (Step 4) The student replaces the teacher and the process is repeated iteratively.

For online methods, it is important that the teacher and student process different augmentations of each unlabeled sample to avoid degenerate solutions. A common strategy is to use weak augmentation for the teacher input and stronger augmentation for the student input (Cubuk et al. [2019, 2020]; Sohn et al. [2020a]). Weak augmentation preserves prediction quality for pseudo-label generation, while strong augmentation improves robustness during student training.

A standard pseudo-labeling objective, following Lee (Lee [2013]), assigns each unlabeled sample a hard pseudo-label derived from the model prediction. For each unlabeled sample $\boldsymbol{x}_j^u$, we define

$$\hat{c}_j^u = \arg\max_c p(c \mid \boldsymbol{x}_j^u), \qquad \hat{\boldsymbol{y}}_j^u = \text{onehot}(\hat{c}_j^u) \ ,$$

where $\hat{c}_j^u$ is the predicted class index and $\hat{\boldsymbol{y}}_j^u$ is the corresponding one-hot pseudo-label vector.

The unlabeled pseudo-label loss is then written as

$$\mathcal{L}_{\text{unsup}} = \frac{1}{|B_u|} \sum_{j \in B_u} \text{CE}\big(p(\cdot \mid \boldsymbol{x}_j^u), \hat{\boldsymbol{y}}_j^u\big) \ ,$$

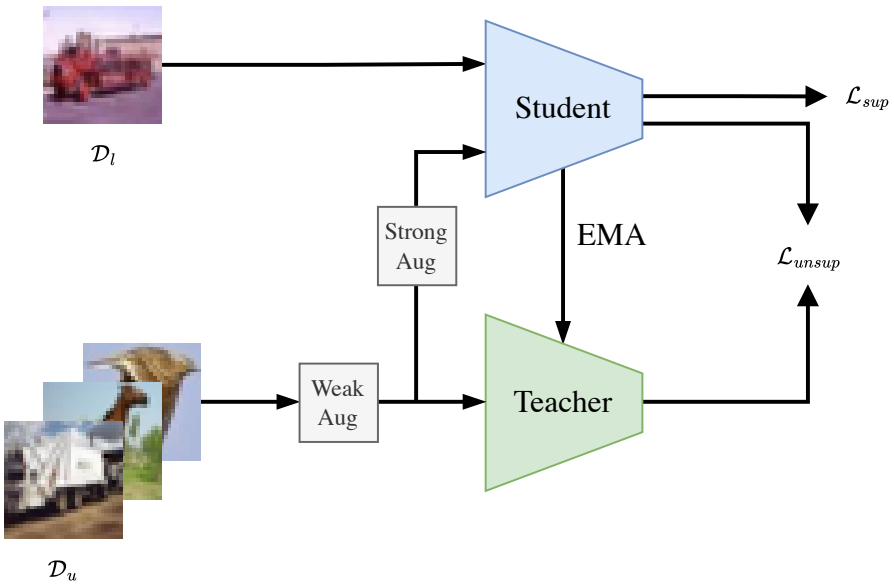


Figure 3: **Online pseudo-labeling pipeline**. Labeled samples from $\mathcal{D}_l$ supervise the student through $\mathcal{L}_{\text{sup}}$, while unlabeled samples from $\mathcal{D}_u$ are weakly augmented for teacher pseudo-label generation and strongly augmented for student training with $\mathcal{L}_{\text{unsup}}$. The teacher is updated as an exponential moving average (EMA) of student parameters, so pseudo-labels evolve continuously during training.

where CE denotes the cross-entropy loss between the model prediction and the pseudo-label.

The full training objective is

$$\mathcal{L} = \mathcal{L}_{\text{sup}} + \lambda_u \mathcal{L}_{\text{unsup}} \ .$$

Because this formulation uses all pseudo-labels without confidence filtering, it is sensitive to uncertain predictions.

Subsequent methods (Chen et al. [2023a]; Pham et al. [2021]; Sohn et al. [2020a]; Wang et al. [2023c]; Xie et al. [2020b]; Zhang et al. [2021]) improve reliability by introducing confidence-aware selection. In the unsupervised loss formulation used by FixMatch (Sohn et al. [2020a]), only pseudo-labels above a confidence threshold contribute:

$$\mathcal{L}_{\text{unsup}} = \frac{1}{|B_u|} \sum_{j \in B_u} \mathbf{1}[s_j \geq \tau] \ \text{CE}\big(p_s(\cdot \mid a_s(\boldsymbol{x}_j^u)), \hat{\boldsymbol{y}}_j^u\big) \,, \qquad s_j = \max_c p_t(c \mid a_w(\boldsymbol{x}_j^u)) \ .$$

Here, p_t and p_s denote the teacher and student output distributions, and a_w and a_s denote weak and strong data augmentations, respectively. The loss CE uses $\hat{\boldsymbol{y}}_j^u$ as one-hot target. Compared to the previous formulation, this loss introduces a threshold value

τ used as the confidence cutoff that determines whether a pseudo-label is retained (*i.e.* when $s_j \geq \tau$) or discarded. This parameter is a central element in this thesis. Finally, the full training loss is

$$\mathcal{L} = \mathcal{L}_{\text{sup}} + \lambda_u \mathcal{L}_{\text{unsup}} \ .$$

Threshold value

The threshold value controls the acceptance of pseudo-labels and directly governs the balance between false positives and false negatives in the unlabeled set.
If the threshold is too low, unreliable predictions are accepted. This increases false positives, where incorrect pseudo-labels are treated as ground truth and reinforce confirmation bias during student training. If the threshold is too high, many correct but moderately confident predictions are rejected. This increases false negatives, where valid supervisory signals are removed and effective data utilization decreases.
The threshold therefore defines a trade-off between label noise and supervision sparsity.

In image classification, conservative high-threshold strategies are often effective because each sample has a single semantic target and no localization ambiguity. However, excessively strict filtering under-utilizes unlabeled data, slows learning, and can amplify class imbalance. As a result, most methods still rely on a discrete grid search to select the threshold.

Figure 4 illustrates this behavior in a representative semi-supervised classification threshold sweep. The best performance is obtained at an intermediate confidence level. Lower thresholds admit too many noisy pseudo-labels, while higher thresholds discard too many informative unlabeled samples.

This trade-off is most critical in the low-label regime introduced earlier in this chapter, where supervised performance degrades as labeled data become scarce. Figure 5 shows that semi-supervised learning yields the largest relative gains in this setting, where unlabeled data provides the strongest complementary signal. As the labeled fraction increases, the relative gain decreases, but semi-supervised learning remains consistently better than supervised training across all evaluated labeled-data fractions.

2.2.2 *Pseudo-labeling in object detection*

Pseudo-labeling in semi-supervised object detection (SSOD) follows the same teacher-student mechanism introduced for classification: a teacher predicts targets on unlabeled data, and a student is trained on labeled and pseudo-labeled samples. The core concepts therefore transfer directly, including augmentation strategies and pseudo-label filtering. The key difference is that detection produces multi-instance predictions, so uncertainty must be handled at the instance level. The central challenge is therefore not only how to

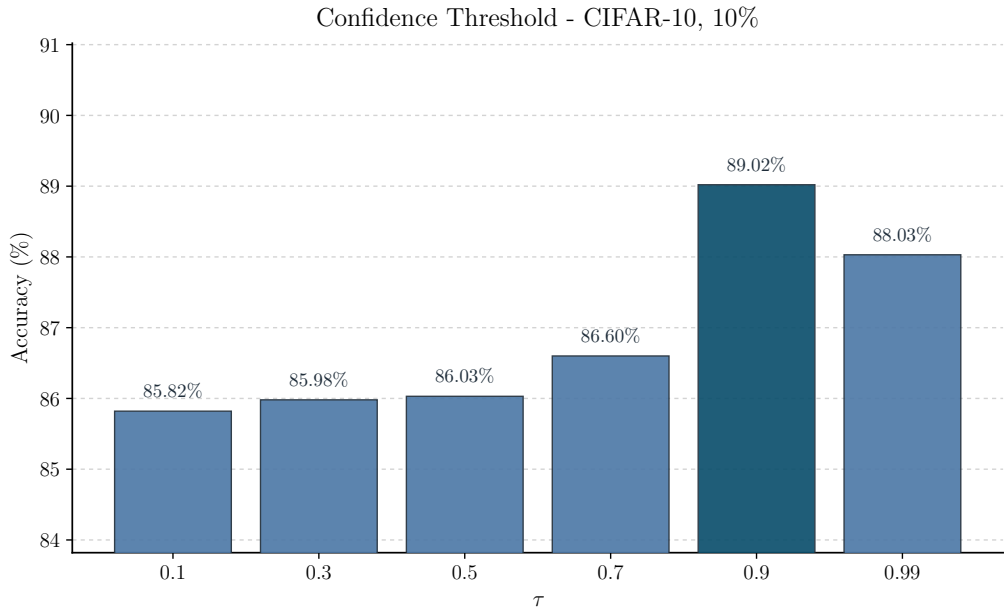


Figure 4: **Confidence threshold sweep**. Sensitivity of semi-supervised CIFAR-10 performance (10% labeled setting) to the pseudo-label confidence threshold τ. Accuracy peaks at $\tau = 0.9$, while lower thresholds admit noisier pseudo-labels and a very high threshold ($\tau = 0.99$) over-filters unlabeled supervision (88.03%).

generate pseudo-labels but how to select predictions that are reliable enough to supervise the student.

Object detection is a central semi-supervised learning setting because dense box annotation is expensive. Early works (Li et al. [2020b]; Sohn et al. [2020b]) introduced offline pseudo-labeling strategies for semi-supervised object detection (SSOD), while subsequent methods (Li et al. [2022a]; Liu et al. [2021b, 2022a]; Sha et al. [2020]) moved toward online teacher-student pipelines.

Compared with image classification, pseudo-label quality in detection depends on several coupled factors: confidence thresholding, duplicate removal, class imbalance, and localization accuracy. A pseudo-label can be class-correct but spatially inaccurate, and such noise directly affects the box regression signal. Conversely, strict filtering can remove ambiguous but informative instances, especially for small or rare objects. Because filtering operates at the instance level, it inevitably rejects some correct predictions, resulting in pseudo-labeled images that contain both incorrect and missing boxes.

Most SSOD methods (Liu et al. [2021b, 2022a]; Sha et al. [2020]; Sohn et al. [2020b]) still rely on empirical threshold search, and reported optimal values vary across datasets and methodologies. This motivates the subsequent chapters. The question around the threshold value is addressed in Chapter 3 through a loss parametrization that accounts for

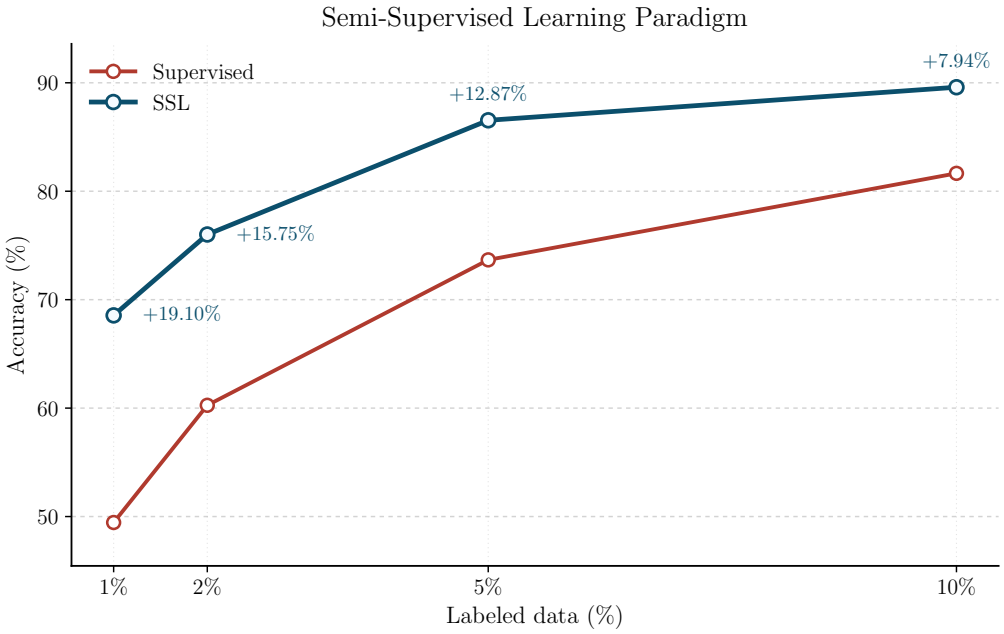


Figure 5: **Supervised vs semi-supervised learning**. Comparison of supervised and semi-supervised CIFAR-10 test accuracy across low-label regimes (1%, 2%, 5%, and 10% labels). Semi-supervised learning consistently outperforms supervised training, with absolute gains of +19.10, +15.75, +12.87, and +7.94 points, respectively, showing that unlabeled data is most beneficial when labels are scarce.

pseudo-label uncertainty in the unsupervised loss, and in Chapter 4 through an adaptive thresholding strategy that removes the need for grid search.

3

UNCERTAINTY IN PSEUDO-LABELS

Outline

This chapter presents the following publication: ***Vandeghen, R.***, Cioppa, A.*, and Van Droogenbroeck, M. Semi-Supervised Training to Improve Player and Ball Detection in Soccer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2022.*

Building on Chapter 2, the chapter studies semi-supervised object detection for soccer broadcast footage and evaluates three loss parameterizations for pseudo-labeled supervision. The central idea is that pseudo-label confidence should not only decide which samples are kept, but also how strongly each retained sample supervises the student.

3.1 PROLOGUE

This work studies uncertainty in teacher predictions during pseudo-labeling for object detection. The main hypothesis is that pseudo-label confidence should affect the training signal used by the student. We therefore evaluate loss parameterizations that explicitly modulate pseudo-labeled contributions as a function of confidence.

Chapter 2 emphasized that pseudo-label filtering in SSOD is usually binary and therefore sensitive to threshold choice. Here, we keep the same offline teacher-student pipeline inspired by early SSOD work (Sohn et al. [2020b]), but we refine the unsupervised supervision signal after thresholding. Instead of treating all retained pseudo-labels equally, we scale their contribution by confidence.

Empirically, this confidence-aware weighting improves both training stability and final detection performance. We also observe that, in constrained in-domain settings such as soccer broadcast footage, conservative thresholding remains beneficial. This chapter therefore addresses the first half of the threshold problem introduced in Chapter 2: how to reduce the impact of uncertain pseudo-labels once a threshold has been applied.

Author contribution

As lead author, I formulated the method, implemented the publicly available codebase (`https://github.com/rvandeghen/SST`), ran the experiments, and wrote the manuscript.

Anthony Cioppa and Marc Van Droogenbroeck supervised the work and contributed to scientific discussion and writing.

Semi-Supervised Training to Improve Player and Ball Detection in Soccer

Renaud Vandeghen* University of Liège
Anthony Cioppa* University of Liège
Marc Van Droogenbroeck University of Liège

Abstract

Accurate player and ball detection has become increasingly important in recent years for sport analytics. As most state-of-the-art methods rely on training deep learning networks in a supervised fashion, they require huge amounts of annotated data, which are rarely available. In this paper, we present a novel generic semi-supervised method to train a network based on a labeled image dataset by leveraging a large unlabeled dataset of soccer broadcast videos. More precisely, we design a teacher-student approach in which the teacher produces surrogate annotations on the unlabeled data to be used later for training a student which has the same architecture as the teacher. Furthermore, we introduce three training loss parametrizations that allow the student to doubt the predictions of the teacher during training depending on the proposal confidence score. We show that including unlabeled data in the training process allows to substantially improve the performances of the detection network trained only on the labeled data. Finally, we provide a thorough performance study including different proportions of labeled and unlabeled data, and establish the first benchmark on the new SoccerNet-v3 detection task, with an mAP of 52.3%. *Our code is available at [https://github.com/rvandeghen/SST].*

1. Introduction

Sports analytics has been steadily growing over the last decade [22], pushed by the development of advanced artificial intelligence and computer tools. Last year, the market was estimated at more than 1 billion dollars, with most indicators pointing out a growth by 500% within the next 5-10 years [14, 18]. Therefore, sports analytics will become even more central for the sports industry in the coming years. Some companies already offer analytics services to clubs with the purpose to improve their playing performances and ascend the championship ladder, thus generating more revenues from ticket sales, advertisements and merchandising.

(*) Denotes equal contributions. Contacts: r.vandeghen@uliege.be and anthony.cioppa@uliege.be.

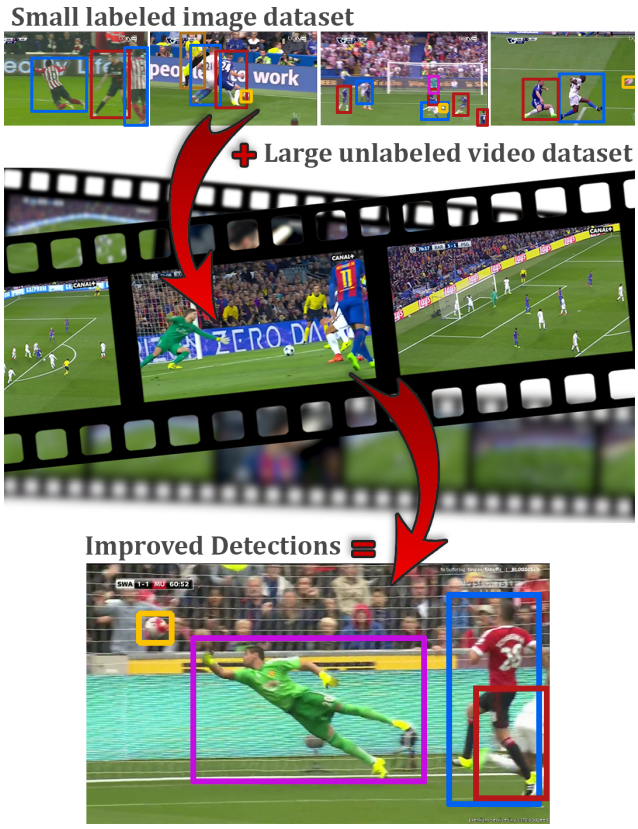


Figure 1. **Overview.** Given a small labeled image dataset for object detection in soccer such as the players, the ball, or the referees, we leverage a large unlabeled dataset of soccer broadcast videos for training an object detector in a semi-supervised fashion. Our training technique allows us to significantly improve the performance of the object detector for the targeted soccer application.

Nowadays, most sports analytics products either rely on manual inspection, which has a heavy cost in terms of manpower, or more recently on automated analysis systems based on artificial intelligence and computer vision techniques. The first step of automated systems often relies on accurately retrieving the players and the ball, which are the key elements to grasp the course of the game. From this information, deeper analyses may be performed such as tracking the players to extract individual speed perfor-

mance, estimating the field coverage by a defending team to unveil potential weaknesses, or analyze critical pass decisions. All these are powerful indicators of an individual's performance, and game strategy analyses may reveal the strengths and weaknesses of opponent or one's own team. Accurately detecting the players and the ball is therefore crucial since analyses rely on these preliminary results.

Over the past few years, artificial intelligence techniques have surpassed their hand-crafted features algorithms counterparts in many areas including player and ball detection in sports. Even though many deep learning detection networks are publicly available for sports companies and researchers, they are often trained on generic data that are not specifically tailored for each sport. The domain gap between the training dataset and the targeted application often results in performances lower than expected, which is why training, or at least fine-tuning, on sport specific data is often required. However, this may require huge amounts of data, which can be costly to annotate and cannot be transferred from one sport to another. Furthermore, some recent works showed that training the network on sport, and even stadium or team, specific data allows to substantially improve the performance of those networks [8].

In this paper, we present a novel generic semi-supervised method for training an object detector with few annotated soccer images, by leveraging a large unlabeled dataset of soccer broadcast videos as illustrated in Figure 1. More specifically, we develop an iterative teacher-student training approach with three different training loss parametrizations for the student, which may doubt the detections performed by the teacher based on their confidence score. We show that including unlabeled data in the training process allows to substantially increase the performances of the detection network on unseen soccer games. Specifically, we provide a complete performance study for different proportions of labeled and unlabeled data, and establish the first benchmark for the detection task on the new SoccerNet-v3 [5] dataset. It is important to note that the presented ideas and achievements do not rely on any data knowledge about soccer, nor on the network architecture. Therefore, our method is applicable to any other sport or domain, characterized by a low amount of annotated data and a large dataset of unlabeled data, and for any detection network.

Contributions. We summarize our contributions as follows. **(i)** We propose a novel semi-supervised method for training a player and ball detection network in soccer games with a teacher-student approach. **(ii)** We introduce three loss parametrizations for training the student with the objective to doubt detections performed by the teacher based on their confidence scores. **(iii)** We establish the first detection benchmark on the new SoccerNet-v3 dataset.

2. Related Work

Object detection in sport analytics. Object detection has been massively studied in the context of sports analytics as it provides a strong basis for further analyses techniques [44]. Even though the first detection algorithms used background subtraction to detect players [2, 35], they have been quickly overthrown by deep learning networks such as convolutional neural networks (CNN). For instance, the authors of [39] use a shallow CNN to detect players on a hockey field with different image representations. Other methods rely on pre-trained networks such as Mask R-CNN [15, 34, 47]. Recently, Cioppa *et al*. [7] proposed a cross-modality online distillation method for player detection and counting on low budget stadium. Liu *et al.* [28] developed a method to detect players and automatically match them with object such as hockey players and their stick.

Some other works use detection as a first step for various downstream tasks such as improving action spotting using camera calibration and player localization [6], player and ball tracking [17, 21, 30], or to model pass feasibility [1].

In order to train deep learning networks, the AI for sports community can count on a large variety of datasets for sports analytics. SoccerNet [11] and SoccerNet-v2 [9] propose 500 complete broadcast soccer games with annotated action events, camera cuts and classes, and replay information. A complementary dataset with spatio-temporal event annotations focusing on player statistical analyses was released by Pappalardo *et al*. [32]. Yu *et al*. [48] and SoccerDB, published by Jiang *et al*. [20], provide annotations for more than 200 soccer games with player bounding boxes and shot transitions. Lately, SoccerNet-v3 [5] was released, providing manual bounding box annotations for player and other objects of interest such as the ball, the lines, and the goal, with extra annotations such as jersey numbers and re-identification of players across multiple views.

Object detection in general. Together with image classification, object detection is among the most studied task in computer vision. Many object detection architectures have been developed in the past few years thanks to the availability of large-scale datasets such as Pascal VOC [10] or MS COCO [26]. Usually, object detectors come into one of two main flavors: two-stage detectors [12, 13, 15, 24, 38], and one-stage detectors [25, 27, 36, 37, 42]. For two-stage detectors, a proposal module is used to propose regions of interest where potential object candidates are likely to be located, for example with a region proposal network such as in Faster R-CNN [38]. The proposals are later refined in a second module, where a class is associated with each predicted bounding box. One-stage detectors operate differently, and directly output the bounding boxes with their classes, leading to faster inference, but often at the price of a lower accuracy compared to their two-stage counterparts.

For these reasons, in this work, we will focus on the two-stage Faster-RCNN [38] architecture, which is widely used in semi-supervised object detection. Note however that our method is applicable regardless of the network architecture.

Semi-supervised object detection. Following the successes of semi-supervised methods achieved for image classification [3, 4, 33, 40, 45, 49], many semi-supervised learning methods for object detection have been developed over the past few years. In 2019, Jeong *et al*. [19] proposed a consistency method for the detections made for an image and its horizontally flipped version. More recently, Sohn *et al*. [41] designed a teacher-student approach [23, 29, 43, 46, 49], where the teacher model is trained with labeled data in a supervised manner, and used to produce pseudo-labels on the unlabeled data. These pseudo-labels, along with the labeled data, are then used to train the student model, leading to better performances. This teacher-student approach relies on a selection mechanism to include or reject pseudo-labels, which is often performed by comparing their confidence score to a threshold. However, determining the appropriate threshold value is an arduous process as it is prone to generate noise, resulting in false positives or false negatives. Therefore, authors have promoted different learning strategies for the student, including Unbiased Teacher [29], which addresses the bias issue regarding the dominant classes with a weighted focal loss [25] for the classification head, and Soft Teacher [46], which uses a confidence score for each pseudo-label to weight the background classification loss. In this paper, we present a weighting strategy on the foreground boxes rather than the background ones, with a doubt mechanism based on the confidence score of the pseudo-labels.

3. Method

Problem statement. We leverage the availability of unlabeled data to improve the detection performance as follows. Given a model tailored for a detection task on images, and trained with a dataset $\mathcal{D}_l$ comprising N_l labeled images, we make use of a dataset $\mathcal{D}_u$ comprising N_u unlabeled images to increase the detection performance of the model; annotations of a labeled image consist in the bounding boxes and classes for all objects contained in it.

This setup is very common in artificial intelligence as datasets are extremely time-consuming and expensive to annotate. Therefore, only a tiny portion of the available data is usually annotated and used for training a model. In this work, we show how to exploit unlabeled images in a semi-supervised fashion for sports analysis. In particular, we propose a method based on a teacher-student approach, where a teacher model $\mathcal{T}$ is trained only with the labeled data, and a student model $\mathcal{S}$ is trained with the labeled and unlabeled, for which pseudo-labels are produced by $\mathcal{T}$.

Iterative semi-supervised training. The first step of our method consists in training the teacher model $\mathcal{T}$ with a standard supervised learning technique on the labeled dataset $\mathcal{D}_l$. Once $\mathcal{T}$ is properly trained, we generate pseudo-labels for images of the unlabeled dataset $\mathcal{D}_u$. More precisely, $\mathcal{T}$ processes each image of $\mathcal{D}_u$ and outputs the box, class and confidence score for each detected object. To avoid multiple predictions of the same object, a classical non-maximum suppression is performed. Let us note that, at this point, the performance of $\mathcal{T}$ corresponds to the typical case of training a model in a supervised fashion on a labeled dataset. Hence, the performance of the first teacher $\mathcal{T}$ is the baseline for comparisons in Section 4.

The next step consists in training a student $\mathcal{S}$, which has the exact same architecture as $\mathcal{T}$, on both $\mathcal{D}_l$ and $\mathcal{D}_u$. The training is performed in a supervised fashion, identical to that of $\mathcal{T}$, but on a larger concatenated dataset (that could be seen as a dataset augmented by $\mathcal{D}_u$). The training loss of $\mathcal{S}$ is taken as the sum of two equal contributions, that is

$$\mathcal{L} = \mathcal{L}_l + \mathcal{L}_u \ , \tag{1}$$

with $\mathcal{L}_l$ and $\mathcal{L}_u$ corresponding to the loss on the labeled dataset and unlabeled dataset, which now contains pseudo-labels, respectively. Once the training is stopped, we fine-tune $\mathcal{S}$ with $\mathcal{D}_l$, to make sure to finalize the training on real ground-truth annotations. While being known in the machine learning community and to the best of our knowledge, the fine-tuning step has only been used once before by Li *et. al* [23] in a self-training method for object detection, despite being highly efficient, as shown in Section 4.

These two steps (generating the pseudo-labels with $\mathcal{T}$ and training $\mathcal{S}$) may be iterated, by considering the last student as the new teacher and re-generating the pseudo-labels on $\mathcal{D}_u$. Hopefully, since the prediction quality of $\mathcal{S}$ is expected to be higher than $\mathcal{T}$, the next pseudo-labels should be better as well and improve the training of the next student.

Since $\mathcal{T}$ is not perfect (otherwise we could stop the training process there), $\mathcal{D}_u$ will contain truly detected objects (true positives), but also some predictions that do not correspond to any real objects (false positives), as well as some missing objects (false negatives). These errors in $\mathcal{D}_u$ affect the training of $\mathcal{S}$, and require to find the best practical trade-off. In the following, we propose three training loss parametrizations for the student based on the confidence score of the proposals in order to reduce the impact of potential errors. The whole pipeline is drawn in Figure 2.

Loss parametrization 1: single threshold. A first way to alleviate false positives in the dataset consists in selecting a subset of the pseudo-labels in $\mathcal{D}_u$ to only retain the true positive predictions and remove the false positive ones. This is usually done by solely keeping predictions with a confidence score higher than a given threshold τ_h. This reduces the number of positive proposals in the $\mathcal{D}_u$ dataset and in-

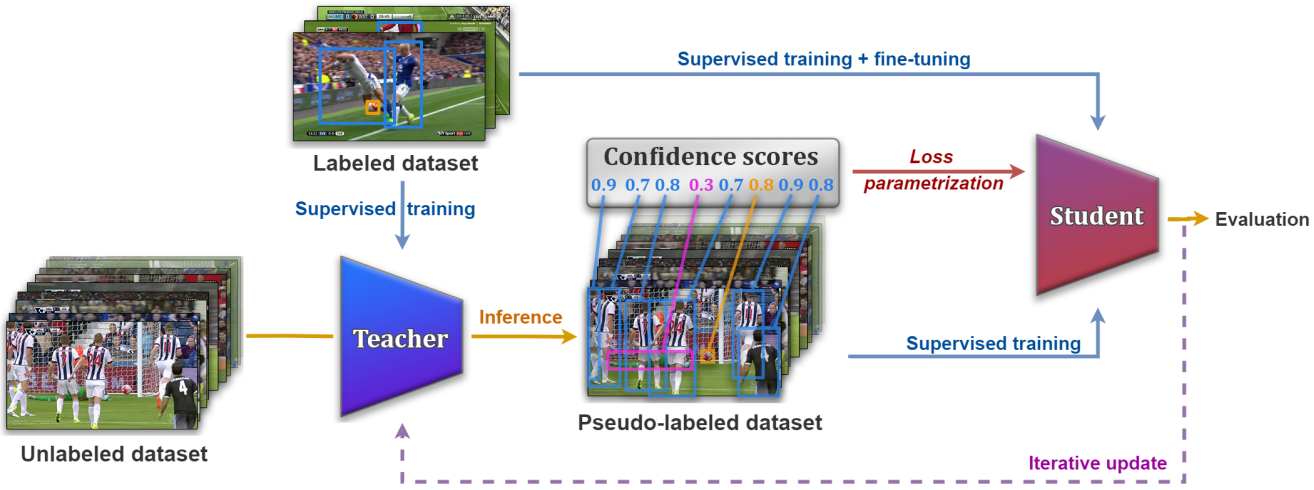


Figure 2. **Overview of our semi-supervised training method for player and ball detection.** We first train a teacher network on a labeled dataset in a fully supervised fashion. Then, we use the trained teacher to produce pseudo-labels on the unlabeled dataset. This creates a first pseudo-labeled dataset, with a confidence score for each prediction. The labeled and pseudo-labeled datasets are then used to train a student network, whose training loss is parameterized based on the confidence score with one of the three parametrization introduced in this paper. This allows the student to doubt unsure proposals by the teacher and achieve good performances on the test dataset. At the end of the training, a final fine-tuning phase is performed with the labeled data, and the student becomes the new teacher for the next iteration.

creases the number of background proposals. The training loss term $\mathcal{L}_l$ of Equation 1, corresponding to the labeled dataset during the training of the student, can be written as:

$$\mathcal{L}_l = \sum_{i=1}^{N_l} \sum_j \mathcal{L}_{cls} + \mathcal{L}_{reg} \ , \tag{2}$$

where $\mathcal{L}_{cls}$ and $\mathcal{L}_{reg}$ denote the classification and box regression loss respectively, and the superscript j stands for the jth proposal for image i. Likewise, the training loss on the unlabeled dataset, $\mathcal{L}_u$, can be written as:

$$\mathcal{L}_u = \sum_{i=1}^{N_u} \sum_j \mathcal{L}_{cls} + \mathcal{L}_{reg} \ . \tag{3}$$

Recent works [29, 41, 43, 46] have shown that using a relatively high threshold value ($\tau \geq 0.7$) ensures pseudo-labels of high quality. This parametrization has two effects: (1) it allows to keep predictions which are supposedly true positives, and (2) predictions boxes with low confidence score are associated to the background and therefore correctly removed. However, the downside is that true positive predictions may also have a confidence score lower than this threshold, leading to the introduction of incorrect false negatives in the dataset. In fact, the threshold value acts as a trade-off between precision and recall, given that lower values tend to increase the recall despite lowering the precision, whereas higher threshold values have the opposite effect. Thus, with the choice of a high threshold value, the trade-off tends towards a higher precision, at the price of introducing false negatives.

Loss parametrization 2: double threshold and doubt. In order to take into account the potential false negatives, we introduce a second threshold value τ_l separating true background predictions with a very low confidence score from the remaining predictions. The goal of this second threshold is to create a range of confidence scores, that is $[\tau_l; \tau_h]$, for which we ignore whether the predictions belong to an actual objects or not. For all predictions with a confidence score in this range, we set the loss to 0 so that the proposals are neither used as positive nor negative examples. This allows to introduce doubt in the training process of the student for unsure predictions of the teacher. The training loss for $\mathcal{D}_l$ is the same as for the first parametrization, but now for $\mathcal{D}_u$, we modify Equation (3) to introduce the new doubt range:

$$\mathcal{L}_u = \sum_{i=1}^{N_u} \sum_j \alpha_j \left(\mathcal{L}_{cls} + \mathcal{L}_{reg} \right) \ , \tag{4}$$

where the term α_j is defined as follows:

$$\alpha_j = \begin{cases} 0 & \text{if } \ \tau_l \leq s_j < \tau_h, \\ 1 & \text{otherwise,} \end{cases} \tag{5}$$

where s_j is the confidence score associated to the jth proposal. Thus, pseudo-labels whose confidence score lies between τ_l and τ_h do not contribute anymore to the loss term $\mathcal{L}_u$. By doing so, we can increase the value of τ_h, ensuring that the positives that we introduce actually correspond to true positives regardless of false negatives introduced in the previous parametrization. This provides more flexibility than for the first parametrization.

Loss parametrization 3: double threshold and progressive doubt. Finally, one could argue that predictions with a confidence score close to τ_h are more reliable than predictions with scores close to τ_l. Therefore, we adapt the second parametrization by introducing a doubt that decreases between the two thresholds. This allows us to tune the uncertainty from high for predictions close to τ_l, to low for predictions as their confidence score approaches τ_h. Equations (2) and (4) stay the same, but Equation (5) becomes:

$$\alpha_j = \begin{cases} \frac{s_j - \tau_l}{\tau_h - \tau_l} & \text{if } \tau_l \leq s_j < \tau_h, \\ 1 & \text{otherwise.} \end{cases} \quad (6)$$

The weighting term of our three parametrizations, for the loss associated with each positive proposal, is illustrated in Figure 3. Note that for the three parametrizations, the weight loss associated with negative proposals is unchanged, regardless of the confidence score, as the background follows a different dynamic than the foreground. Indeed, it is not possible to assign a confidence score to a region without proposals from the teacher. In other terms, this means that we cannot alleviate false negatives already present in the pseudo-labeled dataset. False negative region proposals based on video analysis as in [7], and a loss parametrization for the rejected proposal ($\leq \tau_l$) may be considered. However, they are out of the scope of this paper and could be studied in a further work.

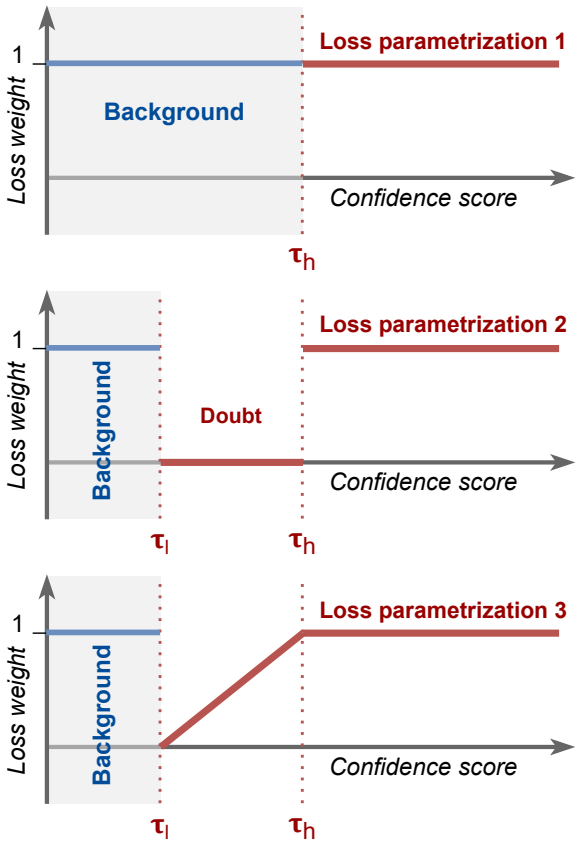


Figure 3. **Our three loss parametrizations for positive candidates.** Comparison of the evolution of the proposal loss weight (corresponding to α_j) with respect to the prediction confidence score for our three parametrizations for positive candidates (in red). (1) Simple threshold value to discriminate between the positive proposals and the background by assigning the same loss weight to all positive samples. (2) Introduction of a second threshold to delimit a doubt zone where the loss is zeroed out. (3) Soft linear approximation for the loss weight in the doubt zone to give more importance to predictions close to τ_h. Note that the loss weight is always 1 for background proposals (in blue), regardless of the parametrization for the positive proposals.

4. Experiments

Dataset. The SoccerNet [11] dataset provides the largest public soccer video collection, including 550 complete broadcast games from the six most influential soccer championships in Europe. Recently, new annotations were released as part of SoccerNet-v3 [5] including 344,660 human bounding boxes of players, referees, and staff, and 26,939 annotations of salient objects such as the ball. These annotations are spread across 33,986 images representing salient moments in soccer with actions such as goals, cards, corners, and their replays.

We choose the training set of SoccerNet-v3 as our labeled dataset, which contains 24,459 frames, its validation set to evaluate performance during training and compare the different loss parametrizations, with 4,797 frames, and its test set for evaluating our final performance, with 4,730 frames. For our unlabeled set, we first retrieve the broadcast videos of the training set games of SoccerNet, which accounts for about 435 hours of video, and extract images at 1 frame per second. This amounts to almost 1,6 million unlabeled frames across 290 different games, which is 64 times more images than the labeled training set!

For the detection task, we focus on the six most important classes for soccer analysis: player, goalkeeper, main referee, side referee, staff, and ball. This amounts to more than 250,000 ground-truth bounding boxes with a highly non-uniform class distribution. This dataset allows us to study our method in many cases ranging from few to many labeled and unlabeled data, with class imbalance and a wide range of object sizes, covering most practical use cases.

Training setup. Both the teacher and student models are based on the same Faster R-CNN [38] architecture with FPN [24] and a ResNet-50 [16] backbone pre-trained on ImageNet. Therefore, these networks are composed of a first-stage region proposal network (RPN) and a second-stage detection network, each having their own classification and regression losses for training. Regarding Equations (1), (2),

(3) and (4), we simply equivalently consider the RPN and detection losses as described in those equations, with the total loss becoming the sum of all four losses.

For the first training phase of the teacher on the labeled dataset, we use the SGD optimizer with an initial learning rate of 0.02, momentum of 0.9, and a weight decay of 10^{-4}. We choose to evaluate our model on the validation set with the mAP ($AP_{50:95}$) metric after every epoch, which is a common metric for object detection. If no improvement is made regarding the mAP for 5 consecutive epochs, we reduce the learning rate by a factor of 10. The models are trained using 4 GPUs with 8 images per batch per GPU, with synchronized batch normalization layers across the different GPUs. For both the RPN and detection modules of Faster R-CNN, we use the standard smooth L1 loss for the regression part $\mathcal{L}_{reg}$ and the cross-entropy loss for the classification part $\mathcal{L}_{cls}$. Note that for the detection module, we also weight the classification loss for each proposal according to the class proportion in $\mathcal{D}_l$, which is a common procedure to counter the class imbalance problem. Specifically, this prevents the networks from focusing too much on the most represented class such as players compared to less represented ones like the balls. Furthermore, we use a simple data augmentation process in which we randomly apply horizontal flipping and color jittering for each training sample. Finally, as an early stopping strategy, we cut off the training of the model if no improvement is made with respect to the mAP on the validation set for 10 consecutive epochs or if the training reaches 200 epochs.

Next, during the inference phase of the teacher, we process all frames of the unlabeled dataset and gather all detection with their confidence scores, localization, and classes, creating the pseudo-labeled dataset. Afterwards, the student network is trained on both the labeled and pseudo-labeled dataset by randomly mixing the samples of both datasets. The exact same training procedure than the one for the first teacher is used except that for each sample of the pseudo-labeled dataset, we parameterize the training loss according to one of the three techniques introduced in Section 3. Once the student finishes training, either by early stopping or by reaching the maximal number of epochs, we fine-tune it on the labeled dataset only.

Finally, the student network is evaluated and becomes the new teacher network for the next iteration. The pseudo-labeled dataset is re-computed with this new teacher and a new student is trained following the above procedure.

Quantitative results. We evaluate our method on increasing labeled dataset sizes to study scenarios ranging from very few to lots of annotated data. In particular, we select the following sizes: $1\%, 5\%, 10\%$, and 100% of $\mathcal{D}_l$, which corresponds to 3, 14, 29 and 290 games (193, 1,196, 2,475, and 24,459 frames respectively). The sampling is operated at the match level rather than at the frame level to stay close to a real-world application in which new data comes from a whole game. For the unlabeled dataset, it is unfortunately too slow to train the model on the whole unlabeled dataset for each setup. Therefore, for most of our experiments, we sample 10 extra matches, not belonging to the labeled matches, which represents around $55{,}000$ frames. Nevertheless, we evaluate our method once on the entire labeled and unlabeled datasets (corresponding to $1{,}596{,}387$ frames) for the best set of parameters found on the restricted unlabeled dataset, which defines the first detection benchmark on the SoccerNet-v3 dataset. Those choices follow the recommendations of Oliver *et al*. [31] regarding the evaluation of semi-supervised learning methods.

Table 1. **Best performances of the teacher and the fine-tuned student after a single iteration.** Performance of our method are given for several labeled dataset sizes, trained with a fixed amount of 10 extra unlabeled games (that is 55,000 frames). According to best practices, hyper parameters such as the threshold values of our parametric losses are optimized on the validation set only. In addition, the performances for the test set are calculated after training with the entire labeled and unlabeled datasets, and the optimal parameters obtained on the validation set. The mAP value of $\mathbf{52.3}\%$ is the first detection benchmark on the new SoccerNet-v3 dataset. (†corresponds to $\tau_h = 0.9$)

Method	τ_l	τ_h	Validation set 1%	5%	10%	100%	Test set 100%
Teacher	-	-	18.1	31.9	39.5	52.7	51.0
Param. 1	-	0.99	25.8†	38.6	**44.3**	53.7	–
Param. 2	0.9	0.99	26.0	38.7	**44.3**	**53.8**	–
Param. 3	0.9	1	**26.2**	**38.9**	43.7	**53.8**	**52.3**

For each labeled dataset size and each loss parametrization, we optimize the threshold values τ_l and τ_h using a grid search strategy on the validation set according to good practice in semi-supervised learning. A complete ablation study of these parameters is presented in the next subsection. The results for the fine-tuned student models after the first iteration may be found in Table 1. As can be seen, the optimal threshold values τ_l and τ_h are quite high for the three loss parametrizations, indicating that we select predictions for which the teacher is extremely confident. Furthermore, for all dataset sizes, each parametrization systematically outperforms the teacher, which is the baseline corresponding to a strictly supervised approach. We can also see that the second and third parametrizations have comparable results, but operate better than the first parametrization with a single threshold. This indicates that doubt introduced by those parametrizations is beneficial for training the student.

Then, we evaluate only once our method trained with the entire labeled and unlabeled datasets on the test set, choosing the best performing loss parametrization and thresholds based on the previous experiments with the restricted unlabeled dataset. As can be seen in Table 1, the best performing method on 100% of the training data with 10 extra games is

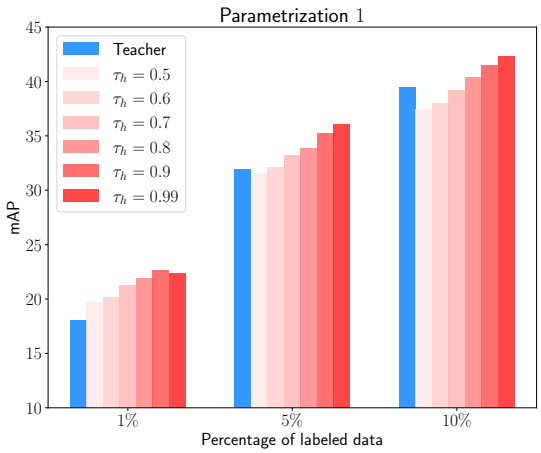


Figure 4. **Optimal threshold value for the first parametrization.** Comparison of the performance of the first parametrization for different threshold values τ_h on various labeled dataset sizes, with 10 extra unlabeled games. The performance of the student increases with the threshold value indicating that only predictions for which the teacher is certain should be considered. Also, the student manages to surpass the teacher for each dataset size.

obtained with the third parametrization and threshold values of $\tau_l = 0.9$ and $\tau_h = 1$. Therefore, we train a student model on the whole labeled and unlabeled dataset with those parameters as well. Since this experiment has a high training time, a single iteration is performed. We achieve an mAP of 52.3% with the fine-tuned student, improving the performance of the teacher by 1.3%, which is slightly better than with 10 extra unlabeled games (52.0% on the test set). This shows that our method improves the detection performance compared with fully supervised methods, especially when considering few annotated data and that more unlabeled data leads to greater improvements.

Ablation study. In this analysis, we start by reviewing the effect of fine-tuning the student, then we propose a thorough study of τ_l and τ_h for our three loss parametrizations, and finally, we explore the further gain one can expect when considering multiple iterations of our method.

First, we discuss the benefit of fine-tuning the student on $\mathcal{D}_l$ at the end of the training process. Table 2 shows the performance of the student before and after fine-tuning for each dataset sizes on the validation set (the results on the right of the arrow are the ones of Table 1). As can be seen, fine-tuning allows to significantly improve the performance no matter the parametrization or the labeled dataset size. For this reason, in this ablation study, we only consider the performance *before fine-tuning* as this step takes consequent computation time and that the important observations can be made on the differences between the performances rather than their absolute values.

Table 2. **Fine-tuning comparison.** Performance improvement when fine-tuning the student network on the labeled dataset at the end of the training for different labeled dataset sizes, with 10 extra unlabeled games. After fine-tuning, the performance increase for all dataset sizes and all parametrizations, showing the importance of this last training step (†corresponds to $\tau_h = 0.9$).

Method	1%	5%	10%	100%
Teacher	18.1	31.9	39.5	52.7
Param. 1	$22.6^\dagger \rightarrow 25.8$	$36.0 \rightarrow 38.6$	$42.3 \rightarrow 44.3$	$52.6 \rightarrow 53.7$
Param. 2	$23.1 \rightarrow 26.1$	$36.6 \rightarrow 38.7$	$43.0 \rightarrow 44.3$	$52.6 \rightarrow 53.8$
Param. 3	$23.0 \rightarrow 26.2$	$36.1 \rightarrow 38.9$	$41.9 \rightarrow 43.7$	$52.7 \rightarrow 53.8$

Table 3. **Optimal threshold values for the second parametrization.** Comparison of the performance of the second parametrization before fine-tuning for different threshold values τ_l and τ_h on 10% of the labeled dataset size with 10 extra games as unlabeled data. The performance of the student increases with both threshold values, indicating that predictions should be considered as background samples for high values of the confidence score as well.

τ_l	0.5	0.5	0.5	0.5	0.6	0.7	0.8	0.9	0.99
τ_h	0.6	0.7	0.8	0.9	0.9	0.9	0.9	0.99	0.999
mAP	38.1	39.0	39.5	40.4	40.9	41.1	41.4	**43.0**	41.0

Second, we investigate the influence of the threshold values on our three loss parametrizations. For the *first loss parametrization*, we study the influence of τ_h which conditions the proportion of false positive and false negative proposals introduced in the pseudo-labeled dataset. The performance of the teacher and student models for the different sizes of labeled dataset and for values of τ_h ranging from 0.5 to 0.99 are shown in Figure 4. For all sizes, increasing the threshold value tends to increase the performance. Furthermore, all student models achieve better performance than the teacher for high threshold values, indicating that even with a simple strategy it is possible to improve on supervised methods using unlabeled data. For the student model trained with 5% and 10% of the labeled dataset, the optimal threshold value corresponds to $\tau_h = 0.99$, showing that it is better to be more selective at the expense of generating false negatives, rather than introducing false positives in the unlabeled dataset.

For the *second parametrization*, we analyze the influence of τ_l and τ_h independently, and provide the results only on 10% of the labeled dataset with 10 extra unlabeled games, since the other labeled dataset sizes lead to similar observations. Our setup is the following: (1) we vary τ_h from 0.6 to 0.9 with a fixed value of $\tau_l = 0.5$, and (2) we vary τ_l from 0.6 to 0.8 with a fixed value of $\tau_h = 0.9$. We also evaluate this parametrization with higher threshold values ($\tau_l = 0.99$ and $\tau_h = 0.999$). All results are presented in Table 3. Similarly to the first parametrization, we see that the performance increases with τ_h. In addition, higher val-

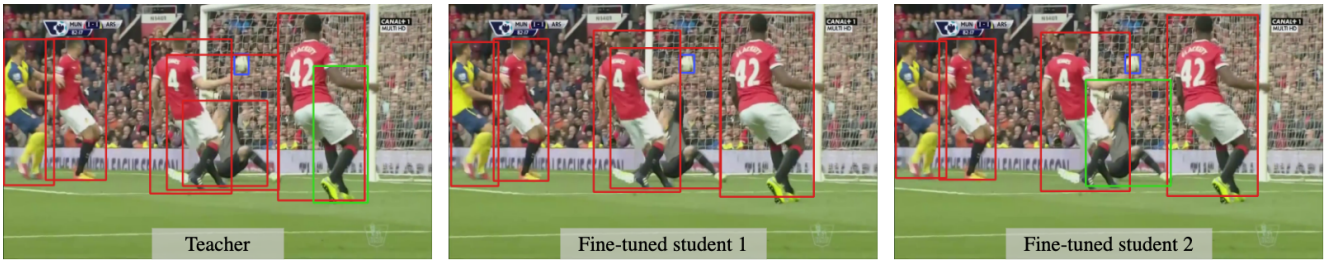


Figure 5. **Qualitative results.** Comparison of the detections on a test set image for the first teacher (left), fine-tuned student model after 1 iteration (middle), and fine-tuned student model after 2 iterations (right). The considered labeled dataset size is 10%, with 10 extra unlabeled games, using the third parametrization for both iterations, with the optimal threshold values presented in Table 1.

Table 4. **Optimal threshold values for the third parametrization.** Comparison of the performance of the third parametrization before fine-tuning for different threshold values τ_l when $\tau_h = 1$, on 10% of the labeled data and 10 extra games as unlabeled data. The performance of the student increases with τ_l showing that only high confidence samples should be considered.

τ_l	0.5	0.6	0.7	0.8	0.9
mAP	39.1	40.1	40.8	41.3	**41.9**

ues for τ_l also lead to better performance. This means that the transition zone between true negatives and positives is around high confidence scores. In other words, detected objects with confidence scores lower than 0.8 should be considered as negative samples rather than being ignored. By construction, this observation is dependent on the considered network architecture and dataset. However, it provides good insights on how we should consider the Faster R-CNN predictions based on their confidence scores. We can also observe that a very high value for τ_l and τ_h reduces the performance of the student.

For the *third parametrization*, we also study the influence of τ_l and τ_h on the performance. From our experiments, we noticed that the best performance is always obtained when choosing $\tau_h = 1$. This means that we should increasingly give credit to the predictions based on their confidence score with no upper limit, independently of the value of τ_l. Therefore, we show the performance when varying τ_l only for this optimal threshold ($\tau_h = 1$). As can be seen in Table 4, the performance increases with the value of τ_l, showing that we should consider predictions with a higher prediction score than before ($\tau_l = 0.9$). In fact, the predictions between the thresholds are not completely ignored compared to the second parametrization, but are simply less considered when approaching τ_l.

Finally, since our method may also be used in an iterative fashion, we provide some insights on to what extend a second iteration of pseudo-labelling using the first student as the new teacher and training a second student further improve the performance. In particular, we study the iterative process with 10% of the labeled dataset and the third parametrization since it gives good performance for one iteration and that its training time is reasonable. As mentioned earlier in Table 1, for this setup, the first teacher and the first fine-tuned student have performances of 39.5% and 43.7%, respectively. After fine-tuning, the second student model reaches an mAP of 45.1%, which further increases the performance compared to the teacher and the first student. In further work, we will study more deeply our iterative process, especially when considering the whole labeled and unlabeled dataset, which is computationally intensive.

Qualitative results. Illustrations of our method's predictions for consecutive iterations are shown in Figure 5 for the first teacher, the first student, and the second student. As can be seen, the first student does not produce false positives, unlike the teacher, but fails at correctly localizing and classifying the goalkeeper. However, the second student manages to correctly detect the goalkeeper. This perfectly illustrates the detection improvements at each iteration.

5. Conclusion

In this work, we propose a new generic semi-supervised method based on a teacher-student approach for object detection. In particular, we show how unlabeled data improves the detection performance of a model trained solely on labeled data. Our method consists in using a teacher trained on labeled data to produce surrogate ground-truth annotations on the unlabeled dataset, later added to the labeled data to train a student model. To adapt the training process to our scenario, we propose three loss parametrizations based on the confidence score of the teacher's predictions to introduce doubt. By doing so, our method substantially improves the performance compared to supervised training. A side result is that we set the first detection benchmark on the new SoccerNet-v3 dataset. Since our method is data and network agnostic, we presume that it is always possible to use available unlabeled data, a common situation in sports analysis, to further improve a detection network.

References

[1] Adrià Arbués Sangüesa, Adrián Martín, Javier Fernández, Coloma Ballester, and Gloria Haro. Using player's body-orientation to model pass feasibility in soccer. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. Workshops (CVPRW)*, pages 3875–3884, Seattle, WA, USA, June 2020. 2

[2] M. Archana and M. Geetha. An efficient ball and player detection in broadcast tennis video. In *Intelligent Systems Technologies and Applications*, volume 384 of *Adv. in Intell. Syst. and Comput.*, pages 427–436. Springer, 2015. 2

[3] David Berthelot, Nicholas Carlini, Ekin Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *Int. Conf. on Learn. Rep. (ICLR)*, Addis Abada, Ethiopia, Apr.-May 2020. 3

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. MixMatch: A holistic approach to semi-supervised learning. In *Adv. in Neural Inform. Process. Syst. (NeurIPS)*, volume 32, Vancouver, Canada, Dec. 2019. Curran Associates, Inc. 3

[5] Anthony Cioppa, Adrien Deliège, Silvio Giancola, Bernard Ghanem, and Marc Van Droogenbroeck. SoccerNet-v3: Scaling up soccernet with multi-view spatial localization and re-identification. *Submitted to Scientific Data*, 2022. 2, 5

[6] Anthony Cioppa, Adrien Deliège, Silvio Giancola, Floriane Magera, Olivier Barnich, Bernard Ghanem, and Marc Van Droogenbroeck. Camera calibration and player localization in SoccerNet-v2 and investigation of their representations for action spotting. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. Workshops (CVPRW), CVsports*, pages 4537–4546, Nashville, TN, USA, June 2021. 2

[7] Anthony Cioppa, Adrien Deliège, Noor Ul Huda, Rikke Gade, Marc Van Droogenbroeck, and Thomas B. Moeslund. Multimodal and multiview distillation for real-time player detection on a football field. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. Workshops (CVPRW), CVsports*, pages 3846–3855, Seattle, WA, USA, June 2020. 2, 5

[8] Anthony Cioppa, Adrien Deliège, Maxime Istasse, Christophe De Vleeschouwer, and Marc Van Droogenbroeck. ARTHuS: Adaptive real-time human segmentation in sports through online distillation. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. Workshops (CVPRW), CVsports*, pages 2505–2514, Long Beach, CA, USA, June 2019. 2

[9] Adrien Deliège, Anthony Cioppa, Silvio Giancola, Meisam J. Seikavandi, Jacob V. Dueholm, Kamal Nasrollahi, Bernard Ghanem, Thomas B. Moeslund, and Marc Van Droogenbroeck. SoccerNet-v2: A dataset and benchmarks for holistic understanding of broadcast soccer videos. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. Workshops (CVPRW), CVsports*, pages 4508–4519, Nashville, TN, USA, June 2021. Best CVSports paper award. 2

[10] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *Int. J. Comp. Vis.*, 88(2):303–338, June 2010. 2

[11] Silvio Giancola, Mohieddine Amine, Tarek Dghaily, and Bernard Ghanem. SoccerNet: A scalable dataset for action spotting in soccer videos. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. Workshops (CVPRW)*, pages 1711–1721, Salt Lake City, UT, USA, June 2018. 2, 5

[12] Ross Girshick. Fast R-CNN. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 1440–1448, Santiago, Chile, Dec. 2015. 2

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. (CVPR)*, pages 580–587, Columbus, OH, USA, June 2014. 2

[14] Christina Gough. Market size of the sports analytics industry worldwide in 2020 and 2028, 2021. https://www.statista.com/statistics/1185536/sports-analytics-market-size/. 1

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2980–2988, Venice, Italy, Oct. 2017. 2

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. (CVPR)*, pages 770–778, Las Vegas, NV, USA, June 2016. 5

[17] Samuel Hurault, Coloma Ballester, and Gloria Haro. Self-supervised small soccer player detection and tracking. In *Int. ACM Workshop Multimedia Content Anal. in Sports (MMSports)*, pages 9–18, Seattle, WA, USA, Oct. 2020. 2

[18] Mordor Intelligence. Sports analytics market – Growth, trends, COVID-19 impact, and forecasts (2022 - 2027), 2022. https://www.mordorintelligence.com/industry-reports/sports-analytics-market. 1

[19] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Adv. in Neural Inform. Process. Syst. (NeurIPS)*, volume 32, Vancouver, Canada, Dec. 2019. Curran Associates, Inc. 3

[20] Yudong Jiang, Kaixu Cui, Leilei Chen, Canjin Wang, and Changliang Xu. SoccerDB: A large-scale database for comprehensive video understanding. In *Int. ACM Workshop Multimedia Content Anal. in Sports (MMSports)*, pages 1–8, 2020. 2

[21] Paresh R. Kamble, Avinash G. Keskar, and Kishor M. Bhurchandi. A deep learning ball tracking system in soccer videos. *Opto-Electronics Review*, 27(1):58–69, Mar. 2019. 2

[22] DTAI Sports Analytics Lab. Why sports analytics, 2019. https://dtai.cs.kuleuven.be/sports/. 1

[23] Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, and Boqing Gong. Improving object detection with selective self-supervised self-training. In *Eur. Conf. Comput. Vis. (ECCV)*, volume 12374 of *Lect. Notes Comp. Sci.*, pages 589–607. Springer, Oct. 2020. 3

[24] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. (CVPR)*, pages 2117–2125, Honolulu, HI, USA, July 2017. 2, 5

[25] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017. 2, 3

[26] Tsung-Yi. Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Eur. Conf. Comput. Vis. (ECCV)*, volume 8693 of *Lect. Notes Comp. Sci.*, pages 740–755. Springer, Sept. 2014. 2

[27] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander Berg. SSD: Single shot multibox detector. *CoRR*, abs/1512.02325, 2016. 2

[28] Yang Liu, Luiz Hafemann, Michael Jamieson, and Mehrsan Javan. Detecting and matching related objects with one proposal multiple predictions. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. Workshops (CVPRW)*, pages 4515–4522, Nashville, TN, USA, June 2021. 2

[29] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Pzizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. In *Int. Conf. on Learn. Rep. (ICLR)*, May 2021. 3, 4

[30] Mehrtash Manafifard, Hamid Ebadi, and Hamid Abrishami Moghaddam. A survey on player tracking in soccer videos. *Comp. Vis. and Image Underst.*, 159:19–46, June 2017. 2

[31] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Adv. in Neural Inform. Process. Syst. (NeurIPS)*, volume 31, Montréal, Canada, Dec. 2018. Curran Associates, Inc. 6

[32] Luca Pappalardo, Paolo Cintia, Alessio Rossi, Emanuele Massucco, Paolo Ferragina, Dino Pedreschi, and Fosca Giannotti. A public data set of spatio-temporal match events in soccer competitions. *Scientific Data*, 6:1–15, Oct. 2019. 2

[33] Hieu Pham, Zihang Dai, QIzhe Xie, Minh-Thang Luong, and Quoc Le. Meta pseudo labels. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. (CVPR)*, pages 11557–11568, Nashville, TN, USA, June 2021. 3

[34] Miran Pobar and Marina Ivasic-Kos. Mask R-CNN and optical flow based method for detection and marking of handball actions. In *Int. Congress on Image and Signal Process., BioMedical Eng. and Inform. (CISP-BMEI)*, pages 1–6, Beijing, China, Oct. 2018. 2

[35] Upendra M. Rao and Umesh C. Pati. A novel algorithm for detection of soccer ball and player. In *Int. Conf. Commun. and Signal Process. (ICCSP)*, pages 344–348, Melmaruvathur, India, Apr. 2015. 2

[36] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, June 2016. 2

[37] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *CoRR*, abs/1804.02767, Apr. 2018. 2

[38] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, June 2017. 2, 3, 5

[39] Melike Sah and Cem Direkoglu. Evaluation of image representations for player detection in field sports using convolutional neural networks. In *International Conference on Theory and Application of Fuzzy Systems and Soft Computing (ICAFS)*, volume 896 of *Adv. in Intell. Syst. and Comput.*, pages 107–115. Springer, 2018. 2

[40] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Han Kurakin, Alexand Zhang, and Colin Raffel. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Adv. in Neural Inform. Process. Syst. (NeurIPS)*, volume 33, pages 596–608. Curran Associates, Inc., Dec. 2020. 3

[41] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *CoRR*, abs/2005.04757, 2020. 3, 4

[42] Mingxing Tan, Ruoming Pang, and Quoc V. Le. EfficientDet: Scalable and efficient object detection. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. (CVPR)*, pages 10778–10787, Seattle, WA, USA, June 2020. 2

[43] Peng Tang, Chetan Ramaiah, Yan Wang, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. In *IEEE Winter Conf. Applicat. Comp. Vis. (WACV)*, pages 2291–2301, Waikoloa, HI, USA, Jan. 2021. 3, 4

[44] Graham Thomas, Rikke Gade, Thomas B. Moeslund, Peter Carr, and Adrian Hilton. Computer vision for sports: current applications and research topics. *Comp. Vis. and Image Underst.*, 159:3–18, June 2017. 2

[45] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves ImageNet classification. In *IEEE Int. Conf. Comput. Vis. and Pattern Recogn. (CVPR)*, pages 10684–10695, Seattle, WA, USA, June 2020. 3

[46] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 3060–3069, Montréal, Canada, Oct. 2021. 3, 4

[47] Yukun Yang, Min Xu, Wanneng Wu, Ruiheng Zhang, and Yu Peng. 3D multiview basketball players detection and localization based on probabilistic occupancy. In *Digit. Image Comp.: Tech. and Applicat.*, pages 1–8, Canberra, ACT, Australia, Dec. 2018. 2

[48] Junqing Yu, Aiping Lei, Zikai Song, Tingting Wang, Hengyou Cai, and Na Feng. Comprehensive dataset of broadcast soccer videos. In *IEEE Conf. on Multimedia Inform. Process. and Retrieval (MIPR)*, pages 418–423, Miami, FL, USA, June 2018. 2

[49] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le. Rethinking pre-training and self-training. In *Adv. in Neural Inform. Process. Syst. (NeurIPS)*, volume 33, pages 3833–3845. Curran Associates, Inc., Dec. 2020. 3

3.2 EPILOGUE

The main contribution of this paper is an uncertainty-aware formulation for pseudo-labeled losses in SSOD. Relative to Chapter 2, the work replaces a purely binary view of pseudo-label reliability with confidence-weighted supervision. Beyond the method itself, it established a practical baseline for SoccerNet-based detection (Deliège et al. [2021]; Giancola et al. [2018]), and the released student model provides a usable off-the-shelf detector for soccer analytics.

This domain also highlights an important challenge: class imbalance. In soccer broadcasts, player instances are much more frequent than referees or goalkeepers, which can produce uneven pseudo-label quality across classes. A natural extension is to enforce better class balance during pseudo-label generation or sampling.

A key limitation remains the dependence on threshold hyperparameter search. In other words, this chapter improves what happens *after* pseudo-label selection, but not *how* the threshold is chosen. This unresolved point directly motivates Chapter 4, which addresses threshold selection explicitly and removes the need for manual sweeps.

4

AUTOMATIC THRESHOLD SELECTION FOR PSEUDO-LABELING

Outline

This chapter presents the following publication: ***Vandeghen, R.**, Louppe, G., and Van Droogenbroeck, M. Adaptive Self-Training for Object Detection. In IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), 2023.*
Following the pseudo-labeling discussion in Chapter 2 and the uncertainty-aware weighting strategy in Chapter 3, this paper addresses the remaining bottleneck in SSOD: manual threshold tuning for pseudo-label selection. We propose a simple adaptive heuristic based on pseudo-label confidence histograms to select thresholds automatically.

4.1 PROLOGUE

Chapter 2 identified thresholding as a central trade-off in pseudo-labeling, and Chapter 3 showed that confidence-aware loss weighting improves robustness once pseudo-labels are selected. The remaining limitation is how thresholds are chosen. Most SSOD methods, including STAC (Sohn et al. [2020b]), Unbiased Teacher (Liu et al. [2021b, 2022a]), and Soft Teacher (Xu et al. [2021]), still rely on manual threshold search for pseudo-label filtering. Reported optimal thresholds vary substantially across setups, and one global value can affect classes unevenly.

To address this issue, we propose an adaptive thresholding heuristic derived from the score distribution of candidate pseudo-labels. The method has three objectives:

1. Remove the need for costly threshold grid search.
2. Enable per-class thresholding at negligible additional cost.
3. Improve transferability across datasets and visual domains.

The proposed pipeline builds directly on Chapter 3. In short, Chapter 3 improves the quality of pseudo-label supervision after selection, while this chapter improves the selection step itself.

Author contribution

As lead author, I designed the method, implemented the public codebase (https://github.com/rvandeghen/ASTOD), conducted the experiments, and wrote the manuscript. Gilles Louppe and Marc Van Droogenbroeck supervised the work and contributed to scientific discussion and writing.

Adaptive Self-Training for Object Detection

Renaud Vandeghen
University of Liège
r.vandeghen@uliege.be

Gilles Louppe
University of Liège
g.louppe@uliege.be

Marc Van Droogenbroeck
University of Liège
m.vandroogenbroeck@uliege.be

Abstract

Deep learning has emerged as an effective solution for solving the task of object detection in images but at the cost of requiring large labeled datasets. To mitigate this cost, semi-supervised object detection methods, which consist in leveraging abundant unlabeled data, have been proposed and have already shown impressive results. These methods however often rely on a thresholding mechanism to allocate pseudo-labels. This threshold value is usually determined empirically for a dataset, which is time consuming and requires a new and costly parameter search when the domain changes. In this work, we introduce a new teacher-student method, named Adaptive Self-Training for Object Detection (ASTOD), which is simple and effective. ASTOD selects pseudo-labels adaptively by examining the score histogram. In addition, we also introduce the idea to systematically refine the student, after training, with the labeled data only to improve its performance. While the teacher and the student of ASTOD are trained separately, in the end, the refined student replaces the teacher in an iterative fashion.

Our experiments show that, on the MS-COCO dataset, our method consistently outperforms other adaptive state-of-the-art methods, and performs equally with respect to methods that require a manual parameter sweep search, and are therefore of limited use in practice. Additional experiments with respect to a supervised baseline on the DIOR dataset containing satellite images lead to similar conclusions, and prove that it is possible to adapt the score threshold automatically in self-training, regardless of the data distribution. The code is available at https://github.com/rvandeghen/ASTOD.

1. Introduction

On the path to consolidate on the successes of supervised deep learning on large labeled datasets, semi-supervised learning is growing in interest to leverage unlabeled data and improve the performance in many computer vision areas, when the amount of labeled data is scarce. Particularly, semi-supervised learning has led to many improvements for the task of image classification [2, 3, 18, 28, 29, 33], and is currently growing in interest for object detection. According to current state-of-the-art research [7, 10, 17, 25, 27, 34], semi-supervised learning methods for object detection (SSOD) are usually based on the principle of self-training, wherein a teacher model is first trained with the labeled data in order to generate pseudo-labels for unlabeled data. Then a second model, called the student, is trained with the pseudo-labeled data. Most of the time, the teacher and the student are trained at the same time in a mutual way.

How can we effectively endorse candidate labels generated by methods in the context of SSOD? This question becomes particularly important when considering state-of-the-art classification methods applied to object detection tasks. More precisely, one has to answer the question of how far endogenous (candidate) labels created by a teacher are to be trusted so that, when added to the labeled dataset, the detection performance of a student network twinned with the teacher network can be improved. Keeping only trustable labels by thresholding the predictions provided by the teacher based on their confidence scores is a simple yet effective method. But beyond this simplicity, determining the adequate threshold value remains tricky. Current works in SSOD often require a costly parameter sweep across different values to determine a suitable threshold. While it is easy to understand the behavior of such a threshold regarding the generation of false positives or false negatives, it is not clear which threshold to choose, as evidenced by previous works where the reported optimal threshold value ranges between 0.5 and 0.9 depending on the datasets and network architectures. Also, most works only cover the case of natural scenes, such as MS-COCO [15] and PASCAL VOC [4], preventing drawing conclusions for other datasets.

In this paper, we introduce our Adaptive Self-Training for Object Detection method (ASTOD) to perform the task of object detection. The main idea of our method is to determine the threshold value applied to the confidence score to select pseudo-labels adaptively which is based on the score histogram of the pseudo-labels. In addition, this strategy has the benefit of determining a threshold value for each class without additional cost, which would be very costly

with a parameter sweep for most practical semi-supervised setups. On top of this adaptive threshold, we use different views of the unlabeled images during the generation of pseudo-labels to improve the predictions of the teacher by reducing the number of missed objects, and to improve the predictions of the bounding box coordinates. It is also important to account for the uncertainty in the pseudo-labeled data when we use them. To do so, we downscale the contribution of pseudo-labels in the loss based on their confidence scores. Lastly, we refine our student with the labeled dataset before using it as our new teacher in an iterative way.

In Section Section 3, we delve into the details of our ASTOD method, whose pipeline is illustrated in Figure 1, after a formal definition of our problem statement. Later, in Section 4, we validate the principle of adaptive self-training with ASTOD for two experimental setups, namely *COCO-standard* and *DIOR*.

Our contributions can be summarized as follows.

- We present a novel end-to-end SSOD method, called ASTOD, based on an iterative teacher-student framework. This method includes a computational-free heuristic based on the score histogram to determine the threshold value for the selection of pseudo-labels.
- We show that using multiple views to generate candidate labels is a simple yet effective technique to improve the labeling process.
- We show that the systematic use of a refinement step is crucial to improve the performance of the student.
- We demonstrate its effectiveness against state-of-the-art methods for two setups.

2. Related Work

Semi-supervised learning. Semi-supervised learning has already been thoroughly studied for image classification. Among the achievements, some methods are based on the principles of consistency training [1, 2, 3, 8, 28, 33], which forces the invariance of a model with respect to input noise by introducing a regularization loss for the unlabeled data. For example, Zhai *et al*. [32] used consistency training to improve the robustness of the model under adversarial attacks. Xie *et al*. [28] have tried another approach in which they minimize the divergence between the output prediction of an unlabeled image and its augmented counterpart.

Another principle for semi-supervised learning is self training [9, 12, 19, 21, 23, 29]. It consists of three parts. First, a teacher model is trained with the labeled data. Then, the trained teacher model is used to generate pseudo-labels on the unlabeled data. Finally, a student model is trained with a dataset comprising the original labels and the pseudo-labels. In particular, Xie *et al*. [29] showed that adding noise during the training of the student model and increasing the network capacity lead to state-of-the-art results.

Semi-supervised object detection. Driven by the successes obtained for image classification, different semi-supervised learning methods have been tailored for the specific task of the object detection [6, 7, 12, 16, 17, 10, 23, 25, 27, 34, 36], even though pioneering work began in 2005 [21]. Among them, most methods [7, 10, 12, 16, 17, 23, 27, 34] use a threshold value determined empirically to select or reject a pseudo-label. Only few of them are designed without threshold [6, 25]. Particularly, one of the first work was done by Jeong *et al*. [6], who proposed a consistency-based semi-supervised learning method by applying consistency between a horizontally flipped image and the original one for the classification part, with the Jensen-Shannon divergence, as well as for the localization part, with a weighted sum of the squared errors of the four different components of the localization loss. They applied this consistency loss for labeled and unlabeled data. Given that this loss can be dominated by the background class, they performed a background elimination, which removes predictions likely to belong to the background. Another approach is to use soft pseudo-labels [25], which means that the whole distribution of class probabilities is used rather than the hard pseudo-label. Those methods give more flexibility as they do not need any threshold value. Among the threshold-based methods, the field of SSOD has grown in interest after that Sohn *et al*. [23] introduced a semi-supervised learning method based on self-training and augmentation-driven consistency regularization. They start by training a teacher in a supervised manner. Afterwards, they use the teacher to generate candidate labels, which are selected when their confidence scores are above a threshold of 0.9. The method then uses a second model, which is trained on both the labeled and pseudo-labeled data by jointly minimizing a conventional supervised loss and a weighted unsupervised loss based on consistency regularization with strong data augmentations. In [12], Li *et al*. presented a selective self-supervised self-training for object detection method. They started with a teacher-student self-training method with a threshold value of 0.7 during pseudo-labeling, and improved the pseudo-labeling step with a so-called selective network. This network splits the set of pseudo-labels into three categories (positive, negative, and ambiguity), but only the positive pseudo-labels are considered in the loss term. They also implemented a consistency term in their loss based on the work presented in [6], which is also only used for the positive pseudo-labels. Liu *et al*. [16] proposed an unbiased teacher, which is a teacher-student method trained in a mutual setting. The teacher generates pseudo-labels for the training of the student and, then, the teacher is updated with exponential moving average (EMA), leading to continually improving models. The

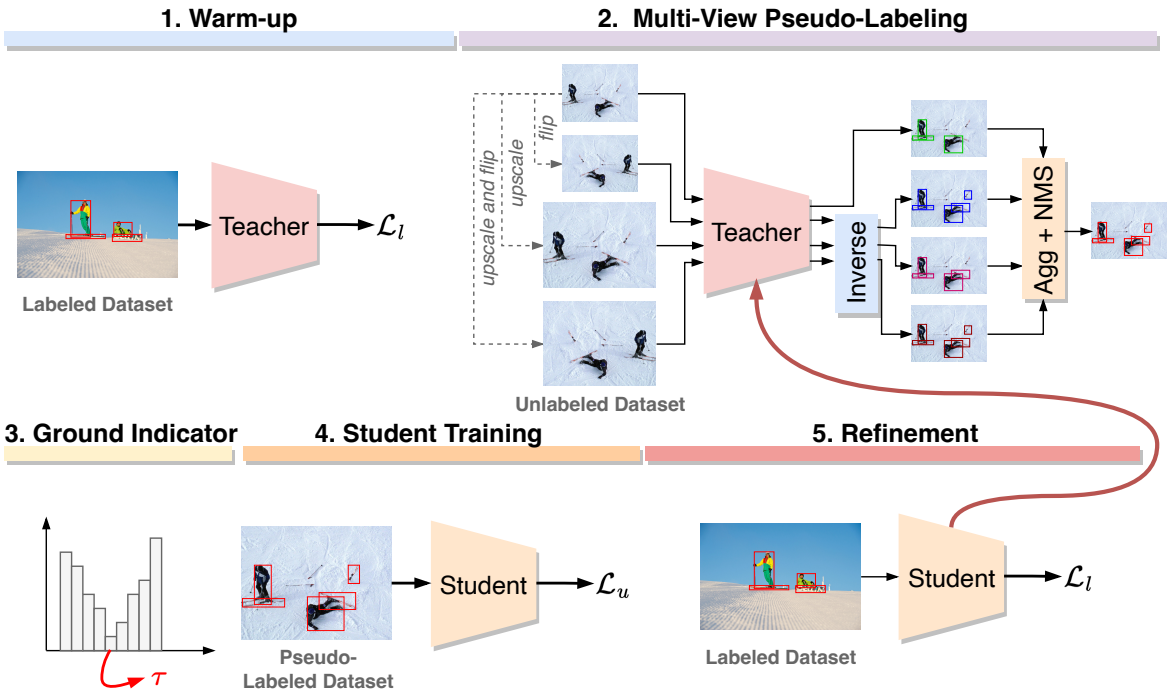


Figure 1: Pipeline of our self-training ASTOD method. (1) A teacher is trained with the labeled dataset. (2) We use the teacher to generate candidate labels on the unlabeled data using multiple views. We apply the inverse view transformation to gather the different predictions in the same dimensional space. The predictions are then merged with NMS. (3) Based on the confidence score histogram, we determine the threshold value τ to filter the candidate boxes, leading to a pseudo-labeled dataset. (4) Next, we train the student with the labeled and pseudo-labeled datasets. (5) Finally, we refine the student with the labeled dataset and use it to replace the teacher. ASTOD can then be used in an iterative fashion by replacing the teacher (2) with the refined student.

authors also used a threshold value of 0.7 to remove boxes with low confidence scores and they addressed the problem of class imbalance by replacing the standard cross-entropy loss with the focal loss [14]. They also published a second version of their unbiased teacher [17] for anchor-free detectors. Zhang *et al*. [34] also addressed the problem of class imbalance with two modules. The first module addresses the problem of foreground-background imbalance by pasting synthetic objects from the training/pseudo dataset in the training images. The second module addresses the problem of foreground-foreground imbalance, which changes the sampling probability with respect to the class occurrence. Kim *et al*. [7] presented a data augmentation technique for the unlabeled dataset that mixes image tiles and feature tiles together and then unmixes the features for the student. Those unmixed feature maps are then processed by the RPN and ROI heads with the pseudo-labels being generated by the teacher with the same weakly augmented images. Li *et al*. [10] also adopted the teacher-student dual learning but took into account the noisy nature of pseudo-boxes regression. Their method is based on a learning scheme that uses multiple views for both the images and the feature maps to enforce consistency. Tanaka *et al*. [24] recently proposed to optimize the threshold based on the β-score and without iterating on the student.

The current literature on the topic of semi-supervised learning for object detection exhibits a wide variety of nuances around a teacher-student scheme and the calculation of pseudo-labels. Alongside, this often results in the heuristic determination of parameters among which the determination of a threshold on the confidence score during the generation of pseudo-labels. As opposed to most approaches of current literature, we intentionally skip this process thanks to an adaptive calculation of such threshold embedded into a new iterative, multiple-view, and refined teacher-student scheme. This forms the basis of our concept of adaptive self-training. In the case of ASTOD, this calculation occurs by analyzing the histogram of scores associated to the generation of pseudo-labels in a fully automatic fashion.

3. Method

Problem statement. We consider a set, $\mathcal{D}$, of images, $\boldsymbol{x}_i$, containing several classes of objects to be detected. Among $\mathcal{D}$, only a subset of images, $\boldsymbol{x}_i^l$, are annotated with the class and localization of all objects of interest, $\boldsymbol{y}_i^l$ (called "labels" or ground-truths), and compose the subset of labeled images $\mathcal{D}_l = \{\boldsymbol{x}_i^l, \boldsymbol{y}_i^l\}_{i=1}^{N_l}$, where N_l is the number of labeled images in $\mathcal{D}$. Each ground-truth $\boldsymbol{y}_i^l$ is composed by a set of classes $\boldsymbol{c}$ and bounding box coordinates $\boldsymbol{b}$. The remaining images of $\mathcal{D}$ with no labels, $\boldsymbol{x}_i^u$, compose the subset of unlabeled images $\mathcal{D}_u = \{\boldsymbol{x}_i^u\}_{i=1}^{N_u}$, where N_u is the number of unlabeled images in $\mathcal{D}$. These subsets are complementary sets (that is $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$), and we assume that they come from the same data distribution. In semi-supervised learning setups, we often have $N_l \ll N_u$.

Teacher warm up. Our method relies on a teacher-student scheme, where the student learns from the pseudo-labels generated by the teacher. Thus, the first step is to learn a teacher that is able to generate high-quality candidate labels, which are all the predictions made by a model without restrictions. The first step of our method then consists in training the teacher model with the labeled data only. We use the conventional training loss for object detection, which is the sum of the classification and regression losses:

$$\mathcal{L}_l = \sum_{i=1} \left[\sum_{j=i} \left(\mathcal{L}_{cls} \left(p(c_j|\boldsymbol{x}_i), c_j \right) + \mathcal{L}_{reg} \left(p(\boldsymbol{b}_j|\boldsymbol{x}_i), \boldsymbol{b}_j \right) \right) \right], \tag{1}$$

where the index j corresponds to the index of an anchor, $p(c_j|\boldsymbol{x}_i)$ is the predicted class probability of anchor j in the image $\boldsymbol{x}_i$, and $p(\boldsymbol{b}_j|\boldsymbol{x}_i)$ are the 4 bounding box coordinates of a predicted bounding box.

Multi-view pseudo-labeling and ground threshold. After we warm up the teacher model, we use it to generate candidate labels for the unlabeled data. Since we want to mitigate the false negatives due to missed predictions and we want the most accurate predictions, the inference of each unlabeled image is processed under multiple views: original image, horizontally flipped image, rescaled image, and both flipped and rescaled image. Afterwards, we apply the inverse transformations to the predictions so that they can be aggregated in the same dimensional space. To reduce redundancy, we apply non-maximum suppression (NMS) on each prediction before and after aggregation. This leads to a subset of candidate pseudo-labels $\mathcal{D}_{\hat{u}} = \{\boldsymbol{x}_i^u, \hat{\boldsymbol{y}}_i^u\}_{i=1}^{N_c}$, with N_c being the number of unlabeled images which have at least one prediction $\hat{\boldsymbol{y}}$, given that we automatically discard images without prediction. Note that each prediction $\hat{\boldsymbol{y}}_i^u$ is composed by the set of predicted classes, its corresponding bounding box coordinates, and the confidence scores $\boldsymbol{s}$ associated to each box. Now that we have access to high-quality candidate labels, we need to select among them those that can be considered as true positives. In contrast to classification tasks, where we can select the class with the highest probability, this is a challenging step for SSOD. Indeed, there can be multiple objects in the same image, meaning that an independent decision must be taken for each anchor. The most straightforward and, by far, most common solution is to threshold the candidates based on their score predictions.

Previous works in the field use a threshold value, denoted by τ, which suits at best their method and the dataset on which they evaluate it. Typical values for this threshold range between 0.5 and 0.9. However, this threshold value is often determined with a costly parameter sweep, unique for all classes, and is optimized for only one image distribution (natural scenes with MS-COCO). To account for those shortcomings, we propose a new heuristic, called *ground thresholding*, based on the score histogram to determine the threshold value: ground thresholding selects the bin with the lowest density. From our experience, taking the bin with the lowest density is a heuristic that constitutes a well-suited compromise solution for eliminating false positives (bins on the left) while preserving a high enough recall (bins on the right). The final pseudo-labeled dataset is then $\mathcal{D}_p = \{\boldsymbol{x}_i^p, \hat{\boldsymbol{y}}_i^p\}_{i=1}^{N_p}$, with N_p being the number of candidate images which have at least one prediction that satisfies $\hat{\boldsymbol{y}}^c \geq \tau$. Since this heuristic does not require the burden of a parameter sweep to find the threshold value, it can be applied independently for each class, which does not bias the threshold value with respect to the class occurrence. It also means that our method can easily generalize to any dataset without any additional computational cost.

Iterative student training. We train the student model in the same manner as for the teacher, but with the labeled and pseudo-labeled data ($\mathcal{D}_{l'} = \mathcal{D}_l \cup \mathcal{D}_p$). During the training of the student, we do not distinguish images coming from $\mathcal{D}_l$ or $\mathcal{D}_p$. However, to account for the uncertainty in the pseudo-labeled data, we generalize the weighting term

$$\alpha_j = \begin{cases} \frac{s_j - \tau_l}{\tau_h - \tau_l} & \text{if } \tau_l \leq s_j < \tau_h, \\ 1 & \text{otherwise.} \end{cases} \tag{2}$$

used in [26] for the loss by fixing $\tau_h = 1$, where τ_h and τ_l represent a high and a low threshold value and s_j the score prediction. This leads to the weighted loss function:

$$\mathcal{L}_u = \sum_{i=1} \left[\sum_{j=i} \alpha_j \left(\mathcal{L}_{cls} + \mathcal{L}_{reg} \right) \right], \tag{3}$$

where $\mathcal{L}_{cls}$ and $\mathcal{L}_{cls}$ are the same classification and regression losses as in Equation (1). The weighting factor α_j used to reduce the contribution of each prediction is then defined as

$$\alpha_j = \begin{cases} \frac{s_j - \tau_j}{1 - \tau_j} & \text{if } \tau_j \leq s_j \leq 1, \\ 1 & \text{otherwise,} \end{cases} \tag{4}$$

with s_j and τ_j being the score and the class-wise threshold value associated to the prediction. Since the score value of labeled data are implicitly set to 1, only pseudo-labeled data contribute to the weighting factor of the loss.

The final step consists in the refinement of the trained student model with the labeled data only. Our method can then be used in an iterative pipeline, where the refined student model will become the new teacher. Since we expect that the student model achieves better results compared to the teacher, its predictions for the candidate labels should be of higher quality which thus leading to an even better new student.

4. Experiments

4.1. Experimental setup

Datasets. Our experimental setup follows the methodology introduced in STAC [23]. In particular, we evaluate our method on two setups: (*setup* 1) natural images on MS-COCO [15] and (*setup* 2) satellite images from the DIOR [11] dataset. For the first setup (called *COCO-standard* hereafter), we randomly sample $1, 2, 5$ and 10% labeled training data out of the 118k images available in the `train2017` split and use the remaining ones as unlabeled training data. For the second setup (*DIOR*), we first shuffle all the labeled images in two parts: the training part with 80% and the validation part with the remaining 20%. Then, we sample 10% of the training dataset as labeled data and the remaining 90% as unlabeled data. Unlike most of the other works in semi-supervised for object detection, which use PASCAL VOC [4] as second dataset, we evaluate our method on satellite images with the DIOR dataset to analyze our method for a totally different image distribution. For both *COCO-standard* and *DIOR*, we report the mean and standard deviation of the $AP_{50:95}$ (mAP) over 5 folds.

Implementation details. For a fair comparison with previous works, we use Faster-RCNN [20] with FPN [13] and a ResNet-50 [5] backbone pretrained on ImageNet [22] as object detector. For the teacher warm-up, we train the model for 20k steps of gradient descent with a starting learning rate of 0.08 that decays after 13k and 18k steps by a factor 10. For the generation of pseudo-labels, we use 4 different views of the unlabeled image: (1) normal view, (2) upscale of the original image by a factor of 2, (3) horizontal flip of the original image, and (4) both upscaling and flipping of the original image. From the score histogram, we set the threshold by selecting the bin with the lowest density between 0.5 and 1 with 21 bins —the choice of the $[0.5, 1]$ range was motivated by the need to select only pseudo-labels with enough confidence, while we choose 21 bins because we wanted an odd number of bins and, by experience, the impact of more bins on the threshold value was insignificant. The student models are trained for 180k steps, with the same learning rate as the teacher, which follows the same decay strategy after 120k and 160k steps. Finally, the student models are refined on the labeled dataset for 10k steps with a learning rate starting at 0.0008 which decays after 6k steps. All the models for *COCO-standard* are trained on 4 GPUs, with a batch size of 16 per GPU. We apply random color and scale jitter as data augmentation. When we train the student models, the batches are formed with 2 labeled and 14 unlabeled images. For *DIOR*, we use 3 scale levels per anchor to better match the different bounding box shapes. The batch size is reduced to 8 per GPU and the student is trained for $90k$ steps.

4.2. Results

COCO-standard. We compare our model with the state-of-the-art semi-supervised object detection methods on *COCO-standard*, as it is the main benchmark adopted by the SSOD community. We group the different methods according to how their threshold value is set, if any. In particular, we group methods that perform a parameter search to find the optimal threshold value. This kind of methods represents most of previous works in the field, such as STAC [23], Soft Teacher [30] or Unbiased Teacher [16, 17]. The second group is for methods that do not need an empirical search for their threshold, such as CSD [6], Humble Teacher [25] and our method. One could say that the former group are dataset dependent while the latter ignore the dataset distribution. Even though our ASTOD method has a threshold parameter, it is adaptive to the dataset, thus closer to the second group than the first. The results are shown in Table 1, where our results are obtained after 3 iterations of student training plus refinement. While being competitive w.r.t. to the state-of-the-art methods with empirical threshold, like Unbiased Teacher v2 [17] and PseCo [10], our method consistently outperforms methods that do not take into account the dataset distribution [6, 25]. It is important to note that if Unbiased Teacher v2 [17] and PseCo [10] have better performances than ASTOD, they would be more challenging to use in practice on a new dataset, simply because there is no data to fine-tune their thresholds.

DIOR. It is important to design SSOD methods that are usable and effective in many different setups. While previous works in the field have mainly focused on natural scene images with MS-COCO and PASCAL-VOC, we decided to evaluate our method with a different and challenging setup. We targeted the field of satellite images because of the growing interest around it, with the DIOR dataset.

Since there is no baseline for SSOD methods for that dataset, we will compare with the supervised baseline achieved by our teacher model. We generated candidate labels with the Flip+Scale strategy and used a class-wise ground threshold. We also refine the student model to further improve its performance. The results of the different

Table 1: Experimental results on *COCO-standard* for the mAP: we report the mean and standard deviation over 5 randomly sampled dataset. We group the different methods w.r.t. to their thresholding strategy. The methods in the middle of the table use a manual empirical search for the threshold value (these methods are thus intractable when applied on a new unknown domain), while methods in the lower part are fully automatic. The results of Supervised† represents the performance of our teacher, which sets our supervised baseline. The results of our method are obtained after 3 iterations of student with refined models, where we used our ground threshold and the multi-views strategies during the pseudo-labeling step.

	1%	2%	5%	10%
Supervised	9.05 ± 0.16	12.70 ± 0.15	18.47 ± 0.22	23.86 ± 0.81
Supervised†	12.14 ± 0.21	16.67 ± 0.30	23.59 ± 0.20	29.34 ± 0.20
STAC [23]	13.97 ± 0.35	18.25 ± 0.25	24.38 ± 0.12	28.64 ± 0.21
Instant Teaching [35]	18.05 ± 0.15	22.45 ± 0.15	26.75 ± 0.05	30.40 ± 0.05
ISMT [31]	18.88 ± 0.74	22.43 ± 0.56	26.37 ± 0.24	30.53 ± 0.52
Unbiased Teacher [16]	20.75 ± 0.12	24.30 ± 0.07	28.27 ± 0.11	31.50 ± 0.10
Soft Teacher [30]	20.46 ± 0.39	-	30.74 ± 0.08	34.04 ± 0.14
Omni-DETR [27]	18.6	23.2	30.2	34.1
Unbiased Teacher v2 [17]	$\mathbf{25.40 \pm 0.36}$	$\mathbf{28.37 \pm 0.03}$	31.85 ± 0.09	35.05 ± 0.02
PseCo [10]	22.43 ± 0.36	27.77 ± 0.18	$\mathbf{32.50 \pm 0.08}$	$\mathbf{36.06 \pm 0.24}$
CSD [6]	10.51 ± 0.06	13.93 ± 0.12	18.63 ± 0.07	22.46 ± 0.08
Humble Teacher [25]	16.96 ± 0.38	21.72 ± 0.24	27.70 ± 0.15	31.61 ± 0.28
Ours (ASTOD)	$\mathbf{19.47 \pm 0.39}$	$\mathbf{24.85 \pm 0.21}$	$\mathbf{30.43 \pm 0.50}$	$\mathbf{34.58 \pm 0.22}$

Table 2: Comparison between models trained on *DIOR*. The refined student models are trained with candidates labels generated with the Scale+Flip technique and a class-wise threshold. We report the mean and standard deviation over 5 randomly sampled dataset.

	Supervised	Student	Refined
mAP	47.59 ± 0.36	51.23 ± 0.35	$\mathbf{52.89 \pm 0.33}$

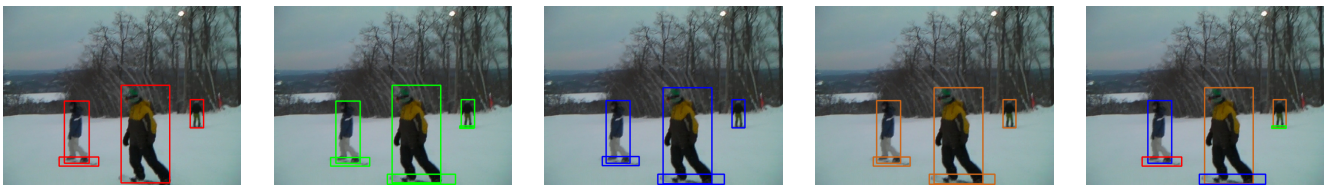
models are presented in Table 2. The gain obtained after one iteration shows the effectiveness and robustness of our method towards completely different data distribution, meaning that it can be further used in other applications.

4.3. Ablation study

We study our method w.r.t. to its different components on *COCO-standard* with a labeled dataset size of 10%.

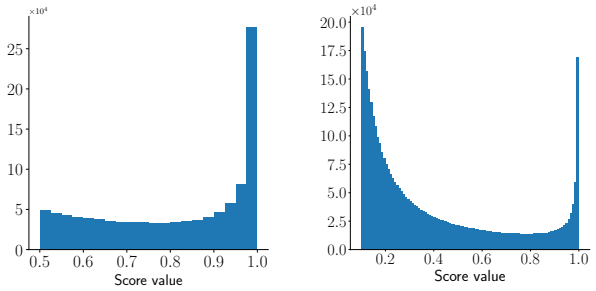
Pseudo-labeling. It is important to rely on high-quality pseudo-labels. To obtain those high-quality pseudo-labels, it is possible to use a high threshold value but at the cost of rejecting potential true positives with lower confidence scores. However, it is not possible to avoid false negatives due to missed predictions. Our multi-view pseudo-labeling strategy can help to reduce their numbers. Figure 2 shows the predicted candidate labels for the different views we consider. We can effectively see that only using the normal view fails to predict some objects in the image, such as the right snowboard, and adding the predictions of the other views solves the problem. Since our aggregation of boxes is performed with NMS, which is a score-based method, the final boxes are a mix of the different views, leading to the best possible candidates.

Ground threshold. The key component of our method is its ability to determine a suited threshold value without empirical search. The proposed strategy is to compute the score histogram and set the threshold value to the bin with the lowest number of instances. While the number of bins and the score range are parameters of the proposed method, Figure 3 shows that the shapes of the histograms are the same, that is U-shaped with high density regions for very low and very high scores, and that they do not influence the position of the lowest density bin. Since this heuristic is independent on the data distribution, it can be applied for each class separately, which gives the possibility to have a set of thresholds rather than a single one. However, training with a uniform threshold seems to achieve better results on *COCO-standard*, as can be seen in Table 4. Looking at Figure 4, which shows the score histogram for a single class and for all the classes jointly, we can see that both of them define a U-shape. This is the shape that we expect since a threshold value lower than the ground threshold would lead to more pseudo-labels, with many of them having a high probability to be false positives. Also, if the score threshold was higher than the ground threshold, we would probably create false negatives. But the problem arises for classes that are hard to learn. For those classes, the chosen heuristic can fail. Since most of the predictions for those classes may have a low confidence score, the histogram can be monotonically decreasing, leading to a ground threshold equal



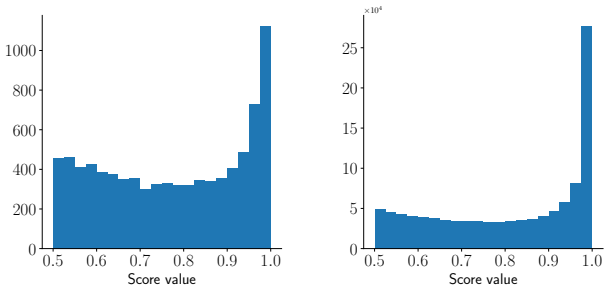
(a) Normal view (b) Scaled view (c) Flipped view (d) Scaled+Flipped view (e) Aggregated view

Figure 2: Comparison between the candidate labels for the different views. The normal view (a) misses two snowboards. Both flipped and scaled+flipped views (c) and (d) miss the small snowboard. Only the scaled view (b) has detected all the snowboards. The aggregated view (e) combines the information of all images (with NMS) to produce the final candidate labels. Note that images (b), (c) and (d) are transformed back to the original space.



(a) Histogram ranging from 0.5 to 1 with 21 bins. (b) Histogram ranging from 0.1 to 1 with 101 bins.

Figure 3: Histograms for different parameters.



(a) Score histogram for a single class. (b) Score histogram for all classes.

Figure 4: Score histograms for a single class ($\tau = 0.7$) (a), and for all the classes ($\tau = 0.75$) (b).

to the last bin. There are multiple consequences to that scenario. First, we generate a lot of false negatives in the pseudo-labeled dataset. Then, the student trains on noisy labels, which will again emphasize the problem of classes hard to learn. Taking a uniform threshold sets the threshold value at a lower score than their ground threshold, leading to fewer false negatives. However, in the *DIOR* setup, we observe that this problem is not present, which we explain by the fact that the teacher is able to better learn the different classes. As shown in Table 4, the student model performs better with a class-wise threshold.

The key advantage of our method is that it eliminates the need for a parameter search to find the threshold value. However, it is interesting to see how it behaves against this parameter search. Table 3 shows the performance of the student model trained with different threshold values and trained with our method. The results depict two interesting behaviors: (1) the optimal threshold from the parameter search does not always gives a better performance, as can be seen on *DIOR*, (2) the optimal threshold value with parameter search for two distinct image distributions does not give the same threshold (0.7 for *COCO-standard* and 0.8 for *DIOR*). This emphasizes that a manual sweep is unsuitable for generalization purposes. It is also important to note that we can process our candidate labels using 6 bins ranging from 0.5 to 1. In this setup, the width of a bin is 0.1, meaning that the bin with the lowest density will match a value that could have been selected with this classical grid search. On *COCO-standard*, we observed that the ground threshold value in this particular setup is 0.7, which appears to be the optimal value in Table 4. This result further consolidate that our ground threshold strategy gives a good threshold value at no cost. Although this particular setup would give the best result for the *COCO-standard* setup, we argue that restraining our method to this particular example does not fulfill our idea to be adaptive to any dataset, which is confirmed with the result obtained on *DIOR*.

Table 3: Performances obtained with a parameter search on the threshold compared to our method. We report the mean over 5 randomly sampled dataset for both setups.

τ	0.5	0.6	0.7	0.8	0.9	Ours
COCO 10%	32.81	32.83	**33.02**	32.82	32.57	32.91
DIOR	52.14	52.29	52.55	52.61	52.49	**52.89**

Iterative students + Refinement. Since our method is not

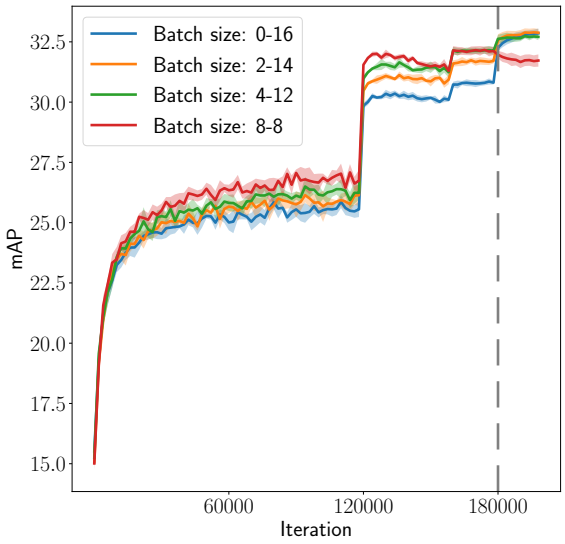


Figure 5: Comparison between the different learning curves of student and refined models w.r.t. the batch size configuration. The vertical dashed line indicate when the refinement step begins.

Table 4: Comparison between refined student models trained with different thresholding strategies. We report the mean and standard deviation over 5 folds for both setups. Surprisingly, using a uniform value performs better than using one determined class-wise on *COCO-standard*.

Setup	Class-wise τ	Uniform τ
COCO-standard	32.56 ± 0.11	$\mathbf{32.91 \pm 0.16}$
DIOR	$\mathbf{52.89 \pm 0.33}$	52.52 ± 0.25

based on a mutual training between the teacher and student models, the pseudo-labels used for the student are fixed. However, if we can obtain better results by the end of its training, we can expect to have better candidate labels by using the student as our new teacher. Before we replace the student as our new teacher, we refine the student on the labeled dataset only for a few gradient descent steps. This idea has already been used by Vandeghen *et al*. [26] to improve the ROI Heads with only trustable ground-truth labels. As it is shown in Table 5 for *COCO-standard* and in Table 2 for *DIOR*, this final trick is highly effective. The results obtained for 3 iterations, before and after refinement, are shown in Table 5.

During our experiments, we performed an analysis on the batch size distribution between the labeled and pseudo labeled images. The different configurations were $0|16$, $2|14$, $4|12$ and $8|8$, for the labeled and unlabeled size respectively. The averaged learning curves of the first student models are shown in Figure 5, where the training of the student models stops at 180,000 iterations. From the student results, it could be obvious to discard the first two configurations. However, those refined models tend to perform better than the last two configurations. This analysis shows that (1) refining the student models is a crucial step to improve their performance, and (2) drawing some conclusions with only the student performance may not be sufficient.

Table 5: Comparison between the different iterations of student and refined models for the mAP on *COCO-standard*. There is a twofold message from those results: (1) Consecutive iterations of student training consistently improve the performance compared to the previous iteration. (2) Refining the student is a simple yet effective way to boost the performance. We report the mean and standard deviation over 5 randomly sampled dataset.

	1%	2%	5%	10%
Supervised	9.05 ± 0.16	12.70 ± 0.15	18.47 ± 0.22	23.86 ± 0.81
Supervised†	12.14 ± 0.21	16.67 ± 0.30	23.59 ± 0.20	29.34 ± 0.20
Student 1	16.57 ± 0.46	21.53 ± 0.34	27.64 ± 0.17	31.77 ± 0.14
Refined 1	16.67 ± 0.36	21.93 ± 0.36	28.47 ± 0.43	32.91 ± 0.16
Student 2	17.75 ± 0.31	23.23 ± 0.29	29.17 ± 0.45	32.88 ± 0.18
Refined 2	17.95 ± 0.37	23.62 ± 0.33	29.54 ± 0.45	33.86 ± 0.18
Student 3	18.71 ± 0.30	24.23 ± 0.34	29.65 ± 0.41	33.40 ± 0.23
Refined 3	$\mathbf{19.47 \pm 0.39}$	$\mathbf{24.85 \pm 0.21}$	$\mathbf{30.43 \pm 0.50}$	$\mathbf{34.58 \pm 0.22}$

5. Conclusion

In this paper, we present ASTOD, an iterative end-to-end self-training method for object detection. Our method solves the problem of parameter sweep for the threshold value in SSOD with a heuristic threshold value which adapts easily to different setups. We also present the systematic use of a refinement step of the student models to improve their performance. Our experiments show that our method largely outperforms state-of-the-art methods in SSOD, that are threshold-free methods.

Limitations and further work. While our method shows an excellent capacity to adapt to diverse data distributions, there is still potential to adapt it to methods which approach the teacher-student scheme with mutual learning. We believe that more work should address the problem of thresholding methods based on parameter search. Finally, a deeper investigation regarding the refining step may be useful, as we have shown that this step consistently improves the performance.

Acknowledgments. The present research benefited from computational resources made available on Lucia, the Tier-1 supercomputer of the Walloon Region, infrastructure funded by the Walloon Region under the grant agreement n°1910247.

References

[1] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, volume 27, Montréal, Can., Dec. 2014. Curran Assoc. Inc. 2

[2] David Berthelot, Nicholas Carlini, Ekin Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *Int. Conf. Learn. Represent. (ICLR)*, Addis Abada, Ethiopia, Apr.-May 2020. 1, 2

[3] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. MixMatch: A holistic approach to semi-supervised learning. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, volume 32, Vancouver, Canada, Dec. 2019. Curran Assoc. Inc. 1, 2

[4] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *Int. J. Comput. Vis.*, 88(2):303–338, Jun. 2010. 1, 5

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 770–778, Las Vegas, NV, USA, Jun. 2016. 5, 11

[6] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, volume 32, Vancouver, Canada, Dec. 2019. Curran Assoc. Inc. 2, 5, 6

[7] JongMok Kim, JooYoung Jang, Seunghyeon Seo, Jisoo Jeong, Jongkeun Na, and Nojun Kwak. MUM: Mix image tiles and UnMix feature tiles for semi-supervised object detection. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14492–14501, New Orleans, LA, USA, Jun. 2022. Inst. Electr. Electron. Eng. (IEEE). 1, 2, 3

[8] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *ArXiv*, abs/1610.02242, 2016. 2

[9] Dong-hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on Challenges in Representation Learning*, volume 3, pages 1–6, Atlanta, Georgia, USA, Jun. 2013. 2

[10] Gang Li, Xiang Li, Yujie Wang, Yichao Wu, Ding Liang, and Shanshan Zhang. PseCo: Pseudo labeling and consistency training for semi-supervised object detection. *ArXiv*, abs/2203.16317, 2022. 1, 2, 3, 5, 6

[11] Ke Li, Gang Wan, Gong Cheng, Liqiu Meng, and Junwei Han. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS J. Photogramm. Remote Sens.*, 159:296–307, Jan. 2020. 5

[12] Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, and Boqing Gong. Improving object detection with selective self-supervised self-training. In *Eur. Conf. Comput. Vis. (ECCV)*, volume 12374 of *Lect. Notes Comput. Sci.*, pages 589–607. Springer, Oct. 2020. 2

[13] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2117–2125, Honolulu, HI, USA, Jul. 2017. 5, 11

[14] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. *ArXiv*, abs/1708.02002, 2017. 3

[15] Tsung-Yi. Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Eur. Conf. Comput. Vis. (ECCV)*, volume 8693 of *Lect. Notes Comput. Sci.*, pages 740–755. Springer, Sept. 2014. 1, 5

[16] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Pzizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. In *Int. Conf. Learn. Represent. (ICLR)*, May 2021. 2, 5, 6

[17] Yen-Cheng Liu, Chih-Yao Ma, and Zsolt Kira. Unbiased teacher v2: Semi-supervised object detection for anchor-free and anchor-based detectors. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9809–9818, New Orleans, LA, USA, Jun. 2022. Inst. Electr. Electron. Eng. (IEEE). 1, 2, 3, 5, 6

[18] Hieu Pham, Zihang Dai, QIzhe Xie, Minh-Thang Luong, and Quoc Le. Meta pseudo labels. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 11557–11568, Nashville, TN, USA, Jun. 2021. 1

[19] Hieu Pham, Zihang Dai, Qizhe Xie, Minh-Thang Luong, and Quoc V. Le. Meta pseudo labels. *ArXiv*, abs/2003.10580, 2020. 2

[20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, Jun. 2017. 5, 11

[21] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. In *IEEE Workshops on Applications of Computer Vision (WACV/MOTION)*, volume 1, pages 29–36, Breckenridge, Colorado, USA, Jan. 2005. Inst. Electr. Electron. Eng. (IEEE). 2

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, Apr. 2015. 5

[23] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *ArXiv*, abs/2005.04757, 2020. 2, 5, 6

[24] Yuki Tanaka, Shuhei M. Yoshida, and Makoto Terao. Non-iterative optimization of pseudo-labeling thresholds for training object detection models from multiple datasets. In *IEEE Int. Conf. Image Process. (ICIP)*, pages 1676–1680, Bordeaux, France, Oct. 2022. Inst. Electr. Electron. Eng. (IEEE). 3

[25] Yihe Tang, Weifeng Chen, Yijun Luo, and Yuting Zhang. Humble teachers teach better students for semi-supervised object detection. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3131–3140, Nashville, TN, USA, Jun. 2021. Inst. Electr. Electron. Eng. (IEEE). 1, 2, 5, 6

[26] Renaud Vandeghen, Anthony Cioppa, and Marc Van Droogenbroeck. Semi-supervised training to improve player and ball detection in soccer. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW), CVsports*, New Orleans, LA, USA, Jun. 2022. 4, 8

[27] Pei Wang, Zhaowei Cai, Hao Yang, Gurumurthy Swaminathan, Nuno Vasconcelos, Bernt Schiele, and Stefano Soatto. Omni-DETR: Omni-supervised object detection with transformers. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9357–9366, New Orleans, LA, USA, Jun. 2022. Inst. Electr. Electron. Eng. (IEEE). 1, 2, 6

[28] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. Unsupervised data augmentation for consistency training. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, volume 33, pages 6256–6268. Curran Assoc. Inc., Dec. 2020. 1, 2

[29] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves ImageNet classification. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10684–10695, Seattle, WA, USA, Jun. 2020. 1, 2

[30] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 3060–3069, Montréal, Can., Oct. 2021. 5, 6

[31] Qize Yang, Xihan Wei, Biao Wang, Xian-Sheng Hua, and Lei Zhang. Interactive self-training with mean teachers for semi-supervised object detection. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 5941–5950, Nashville, TN, USA, Jun. 2021. 6

[32] Runtian Zhai, Tianle Cai, Di He, Chen Dan, Kun He, John Hopcroft, and Liwei Wang. Adversarially robust generalization just requires more unlabeled data. *ArXiv*, abs/1906.00555, 2019. 2

[33] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4L: Self-supervised semi-supervised learning. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 1476–1485, Seoul, South Korea, Oct. 2019. Inst. Electr. Electron. Eng. (IEEE). 1, 2

[34] Fangyuan Zhang, Tianxiang Pan, and Bin Wang. Semi-supervised object detection with adaptive class-rebalancing self-training. *ArXiv*, abs/2107.05031, 2021. 1, 2, 3

[35] Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, and Hao Li. Instant-teaching: An end-to-end semi-supervised object detection framework. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 4079–4088, Nashville, TN, USA, Jun. 2021. Inst. Electr. Electron. Eng. (IEEE). 6

[36] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le. Rethinking pre-training and self-training. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, volume 33, pages 3833–3845. Curran Assoc. Inc., Dec. 2020. 2

6. Supplementary Material

6.1. Implementation details.

Networks. We use a pre-trained ResNet-50 [5] as backbone for Faster-RCNN [20] with FPN [13] as object detector.

Training parameters. For the *COCO-standard* setup, the teacher models are warmed-up for 20k steps with a learning rate decay after 13k and 18k steps. Then, our student models are trained for 180k steps, using a global batch size of 64. We apply the same learning rate decay after 120k and 160k steps. We use SGD as optimizer, with an initial learning rate of 0.08 and with default other parameters. The refined student models are trained for 10k steps, using a initial learning rate of 0.0008, which is reduced after 6k steps. For the *DIOR* setup, the student models are trained for 90k steps with a learning rate decay after 60k and 80k steps. The different values are gathered in Table 6.

Table 6: Hyper-parameters used during the training of the different models

Parameters	*COCO-standard*			*DIOR*		
	Teacher	Student	Refined	Teacher	Student	Refined
Training steps	20k	180k	10k	20k	90k	10k
Learning rate	0.08	0.08	0.08	0.08	0.08	0.08
Learning rate decay	13k-18k	120k-160k	6k	13k-18k	60k-80k	6k
Batch Size (labeled — pseudo labeled)	64|0	8|56	64|0	32|0	8|24	32|0

Data augmentations. For the data augmentations during training, we use some large scale color jittering, such as random changes in brightness, contrast, hue and saturation. We also apply some scale jittering and random horizontal flips.

6.2. Student training.

In Table 7, we show the results of refined student models trained with pseudo-labels generated with different view strategies. The idea of using the four different views (normal, flip, scale and flip+scale). We can see that only scaling up the view gives worse results, but scaling and flipping gives a tiny improvement compared to only flipping the image.

Table 7: Comparison between refined student models trained with different view techniques during the generation of candidate labels for the mAP on *COCO-standard*. For the Scale+Flip technique, we use the information of the normal view, the scaled/flipped only view and the scale+flip view. Adding multiple views is a simple yet effective way to improve the quality of candidate labels. We report the mean and standard deviation over 5 randomly sampled dataset.

Transformation	None	Scale	Flip	Scale+Flip
mAP	32.87 ± 0.23	32.76 ± 0.17	32.90 ± 0.19	$\mathbf{32.91 \pm 0.16}$

We also study the effect of weighting the loss in Equation (3) during the training of the student. We trained a student by fixing the α term in Equation (3) to 1. On average, the gain of mAP is 0.13 for the model trained with the weighted loss.

4.2 EPILOGUE

The main contribution of this work is a practical adaptive mechanism for pseudo-label threshold selection in SSOD. The method matches or improves manually tuned baselines while avoiding repeated parameter sweeps.

We validate the approach on both COCO (Lin et al. [2014]) and DIOR (Li et al. [2020a]), which propose different data distributions. This cross-domain evaluation confirms a central point from Chapter 2: fixed thresholds are sensitive, and a value tuned on one dataset does not necessarily transfer to another. In addition to adaptive thresholding, the method benefits from two complementary components: confidence-aware loss weighting for pseudo-labels (inherited from Chapter 3) and a final refinement stage that fine-tunes the student on labeled data only.

Two limitations remain. First, the method is designed for offline pseudo-labeling because histogram estimation requires full pseudo-label sets. Extending it to online methods likely requires a running score memory. Second, the heuristic assumes informative score distributions; for rare or difficult classes, the histogram can be unstable, which weakens threshold estimation.

Together, Chapters 3 and 4 provide a coherent response to the semi-supervised part of the thesis question in Chapter 1: they improve both pseudo-label reliability and pseudo-label selection under limited supervision. The next part of the manuscript shifts from semi-supervised detection to self-supervised pretraining, starting with masked modeling in Chapter 5.

"Education is not the learning of facts, but the training of the mind to think."

Albert Einstein

Part II

SELF-SUPERVISED LEARNING

5 MASKED MODELING

Outline

This chapter introduces masked modeling as a core paradigm for self-supervised pretraining in vision. We start from the central idea of Masked Autoencoders (MAE) for images, then extend the discussion to masked video modeling. The chapter emphasizes masking and reconstruction choices that matter for motion-sensitive pretraining.
We assume that the reader is familiar with the basics of machine learning, deep learning, and computer vision.

5.1 MASKED IMAGE MODELING

The key idea of MAE is simple: hide most of an image, then train a model to recover the missing content from the remaining visible patches (He et al. [2022]). To succeed, the model must infer global structure, object layout, and local texture from partial evidence, which turns reconstruction into a representation learning problem rather than a pure pixel interpolation problem.

The MAE pipeline follows a mask-and-predict strategy described hereafter and shown in Fig. 6:

1. The image is split into non-overlapping patches (*patchify*) (Dosovitskiy et al. [2021]).
2. A large subset of patches is randomly masked.
3. The encoder processes only visible patches.
4. A lightweight decoder receives encoded visible tokens and mask tokens.
5. The model reconstructs masked patches.
6. After pretraining, only the encoder is retained for downstream transfer.

Formally, let $\boldsymbol{x} \in \mathbb{R}^{H \times W \times C}$ denote an input image. This image is first patchified (Dosovitskiy et al. [2021]) yielding N patches $\mathcal{P} = \{\boldsymbol{p}_i\}_{i=1}^{N}$. A random mask set $\mathcal{M} \subset \{1, \ldots, N\}$ is sampled at a high ratio ($\rho \approx 75\%$ in standard MAE), and only visible patches

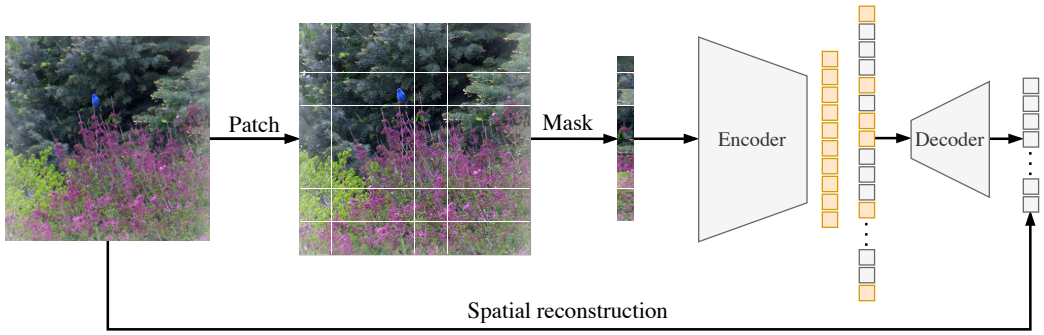


Figure 6: **Masked Autoencoder (MAE) pretraining pipeline** (He et al. [2022]). An input image is patchified, a large subset of patches is masked, and only visible tokens are processed by the encoder. A lightweight decoder reconstructs masked content from encoded visible tokens and learnable mask tokens using a pixel reconstruction loss. The pretrained encoder is then transferred to downstream tasks.

$\mathcal{P}^{visible} = \{\boldsymbol{p}_i \,|\, i \notin \mathcal{M}\}$ are processed by the encoder model. A lightweight decoder model then predicts the missing patches $\hat{\boldsymbol{p}}_i$ for $i \in \mathcal{M}$. The reconstruction objective is

$$\mathcal{L}_{\text{MAE}} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \|\boldsymbol{p}_i - \hat{\boldsymbol{p}}_i\|_2^2 \quad . \tag{1}$$

Three design choices explain why MAE is effective in practice (He et al. [2022]). First, high-ratio masking creates a strong information bottleneck and forces non-trivial inference. Second, the encoder processes only visible tokens, which makes pretraining computationally efficient. Third, the asymmetric encoder-decoder design keeps most modeling capacity in the encoder, which is the component transferred to downstream tasks. This MAE formulation serves as the conceptual foundation for the object-centric image pretraining strategy in Chapter 6.

5.2 MASKED VIDEO MODELING

Masked video modeling (MVM) extends MAE from images to clips, often formed by 16 temporally subsampled frames, using spatiotemporal tokens (tubelets) (Feichtenhofer et al. [2022]; Tong et al. [2022]; Wang et al. [2023a]). Let $\boldsymbol{x} \in \mathbb{R}^{H \times W \times T \times C}$ denote an input clip, consistent with Chapter 2. The high-level pipeline remains analogous to MIM: tokenize, mask, encode visible tokens, decode with mask tokens, and then reconstruct masked regions.

The main adaptation from image MAE is the masking policy. Two common variants are used in practice. In random tube masking, masked spatial locations are aligned across time. In frame-wise random masking, each frame receives an independent random mask.

In both cases, very high masking ratios are common because adjacent frames are strongly redundant (Tong et al. [2022]).

Two design axes are central in modern MVM:

1. **What to mask.** Uniform random masking is simple and effective (Tong et al. [2022]), while motion-aware or temporally structured masking can allocate visible tokens more deliberately across static and dynamic regions (Fan et al. [2023]; Huang et al. [2023]).

2. **What to reconstruct.** Pixel-space reconstruction is direct and stable (Feichtenhofer et al. [2022]; Tong et al. [2022]), whereas feature-level or motion-oriented targets can encourage stronger semantic and temporal abstraction (Girdhar et al. [2023]; Wang et al. [2023a]; Yang et al. [2024]).

These choices directly affect transfer behavior to downstream tasks. Appearance-dominated pretraining often performs well when static scene structure is sufficient, but it can underperform in fine-grained temporal reasoning. This limitation motivates explicit motion learning during pretraining, which is particularly important for motion-dominated benchmarks where temporal dynamics are the primary signal.

This distinction motivates the second self-supervised contribution of this thesis: motion-aware masked video pretraining with explicit temporal supervision, introduced in Chapter 7.

6

OBJECT-CENTRIC REPRESENTATION LEARNING WITH MASKED IMAGE MODELING

Outline

This chapter presents the following publication: *Eymaël, A.*[*], ***Vandeghen, R.***[*], *Cioppa, A., Giancola, S., Ghanem, B., and Van Droogenbroeck, M. Efficient Image Pre-training with Siamese Cropped Masked Autoencoders. In European Conference on Computer Vision (ECCV), 2024.*
The paper investigates object-centric representation learning with a Siamese masked modeling framework that uses paired crops from images rather than paired frames from videos.

6.1 PROLOGUE

Chapter 5 framed masked modeling as a representation-learning problem driven by two core decisions: what information is hidden and what target is reconstructed. This chapter instantiates that framework in the image setting, with an explicit focus on object-centric representation learning.

The project started from a broader objective on temporal self-supervision. During development, the central question became whether object-level invariances could be learned efficiently without video, using spatial view variation alone. Compared with classical MAE (He et al. [2022]), which primarily captures global image structure, our approach biases learning toward object-centric cues through Siamese masked pretraining on paired crops of the same image.

This design choice creates a direct connection between the conceptual discussion in Chapter 5 and the method studied here. We keep the MAE-style mask-and-predict principle, but we replace temporal pairing with image-level crop pairing, reducing data and compute requirements while maintaining strong transfer behavior on object-centric tasks.

Author contribution

I co-authored this work with Alexandre Eymaël during his master's thesis. I co-designed the methodology, co-ran experiments, and co-wrote the manuscript. I also supervised Alexandre's day-to-day research progress throughout the project and submission phase. Anthony Cioppa, Silvio Giancola, Bernard Ghanem, and Marc Van Droogenbroeck supervised the work and contributed to scientific direction and writing.

Efficient Image Pre-Training with Siamese Cropped Masked Autoencoders

Alexandre Eymaël*[1], Renaud Vandeghen*[1], Anthony Cioppa[1,2], Silvio Giancola[2], Bernard Ghanem[2], and Marc Van Droogenbroeck[1]

[1] University of Liège, Belgium
[2] KAUST, Saudi Arabia
r.vandeghen@uliege.be

Abstract. Self-supervised pre-training of image encoders is omnipresent in the literature, particularly following the introduction of Masked autoencoders (MAE). Current efforts attempt to learn object-centric representations from motion in videos. In particular, SiamMAE recently introduced a Siamese network, training a shared-weight encoder from two frames of a video with a high asymmetric masking ratio (95%). In this work, we propose CropMAE, an alternative approach to the Siamese pre-training introduced by SiamMAE. Our method specifically differs by exclusively considering pairs of cropped images sourced from the same image but cropped differently, deviating from the conventional pairs of frames extracted from a video. CropMAE therefore alleviates the need for video datasets, while maintaining competitive performances and drastically reducing pre-training and learning time. Furthermore, we demonstrate that CropMAE learns similar object-centric representations without explicit motion, showing that current self-supervised learning methods do not learn such representations from explicit object motion, but rather thanks to the implicit image transformations that occur between the two views. Finally, CropMAE achieves the highest masking ratio to date (98.5%), enabling the reconstruction of images using only two visible patches. Our code is available at `https://github.com/alexandre-eymael/CropMAE`.

Keywords: Self-supervised learning, Masked autoencoders, Siamese networks, Video segmentation, Label propagation.

1 Introduction

Self-supervised learning (SSL) has become increasingly popular in the last few years thanks to its capacity to learn meaningful and robust representation without the need for labels, sometimes even leading to performances on downstream tasks surpassing its supervised counterpart. This is especially interesting in domains in which data labelling is costly, such as image segmentation or object detection, or when the exact task to solve is not known beforehand [1]. Among popular self-supervised paradigms, visual contrastive learning [7, 20, 24] and masked image modeling (MIM) [23, 30, 48] have received significant interest due to their

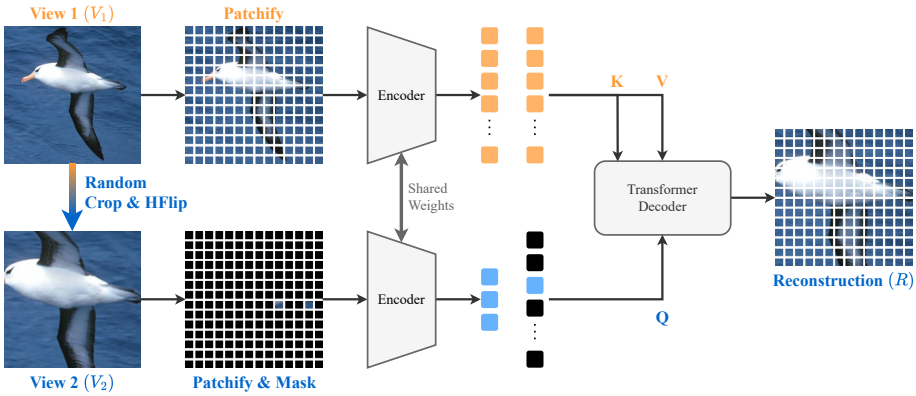


Fig. 1: CropMAE self-supervised pre-training. Given an input image (V_1), a second image is generated by performing a random crop and, optionally, a horizontal flip on the original image (V_2). We then patchify [13] both views and mask [23, 30] an extremely high portion of the second image (above 98.5%). Both views are encoded by a Siamese [5] ViT encoder, with added positional embedding [13]. A transformer [19] decoder reconstructs the masked image R using self-attention layers on the tokens of the masked image and cross-attention layers between the tokens of the masked and unmasked images.

impressive performance. While highly effective, MIM methods often require a large amount of data and/or extensive training time to achieve satisfactory performance [16, 21, 41]. This necessity largely stems from their objective to develop a conceptual understanding of the data distribution they are trained on, enabling them to reconstruct images at the pixel level. This challenge is particularly pronounced with Vision Transformers (ViTs) [13] as encoders, as they perform suboptimally with limited data due to the lack of visual inductive biases that they exhibit [13]. A major weakness of contrastive learning techniques is that they rely on carefully chosen transformations to achieve good performances [7, 20, 47].

Recently, Siamese Masked autoencoders (SiamMAE) [21] achieved state-of-the-art performance in numerous propagation tasks [27,35,52] by learning object-centric representations from videos. This method leverages a Siamese encoder [5] to process pairs of frames that are asymmetrically masked. Despite its impressive performance, SiamMAE faces two main limitations. Firstly, it is designed to only process video frames, not standalone images. Yet, image datasets are typically orders of magnitude larger than video datasets, and less computationally expensive to decode, making image-based pre-training more effective and scalable than video-based pre-training. Secondly, while SiamMAE reduces the need for the intense data augmentation found in contrastive learning methods, it still requires learning a conceptual understanding of the visual world, similar to most MIM techniques, thus requiring extensive training (2,000 epochs) on large datasets such as K400 [29] to reach state-of-the-art performances.

In this work, we propose a novel self-supervised learning method, called *CropMAE*, that reframes the siamese-based paradigm introduced in SiamMAE in order to alleviate the need for video dataset, while keeping competitive performances on downstream tasks. Specifically, we use random views of the same image to simulate viewpoint changes, object transformations, motion, and occlusions. Our method can therefore leverage both image and video datasets, and train at a significantly faster pace than SiamMAE. Moreover, we demonstrate that CropMAE learns meaningful object-centric representations for downstream video tasks without explicit motion. Finally, unlike most MIM techniques, the pretext task of CropMAE is directly tractable based on the visible frame without the need to learn conceptual information about the world, which we believe is the reason for its faster training. An overview of our method is presented in Figure 1.

Contributions. We summarize our contributions as follows. **(i)** We introduce a novel pre-training method, CropMAE, based on sole images, which alleviates the need for video decoding and significantly accelerates training. The novel pretext task we introduce learns faster while quickly reaching good performances. **(ii)** We empirically demonstrate the feasibility of learning meaningful representations for downstream video tasks from still images or data distributions traditionally not associated with videos. Notably, this approach yields better results than training directly on video frames. **(iii)** We show, for the first time, that employing an extremely high masking ratio (98.5%, *i.e.*, using only two visible patches for a ViT/16), surpassing those explored in existing studies, can be optimal and generate a meaningful and challenging self-supervised task.

2 Related Work

Visual representation learning. Visual self-supervised learning focuses on learning rich and generalizable representations of images or videos. This is typically achieved through pretext tasks [7, 32, 33, 39], enabling the learned representations to be applicable to a broad set of downstream tasks [11, 14, 52], either by fine-tuning the learned models for specific tasks, or by freezing the weights and training a linear classifier or an MLP on top of it. Key downstream tasks in the visual domain include image classification [3, 6, 7, 9, 18, 20, 23, 34, 48, 51], video classification [15,16,18,34,41,44], object detection [9,20,23], and video segmentation [4,6,9,21,28]. Our method, CropMAE, is a new visual self-supervised representation learning method for propagation tasks [27, 35, 52].

Contrastive Self-Supervised Learning. Contrastive self-supervised learning [22] has been recognized as an effective method for feature extraction, applicable both to images [9, 20] and videos [10, 38]. This approach encourages the encoder to learn robust representations of the input data by minimizing the distance between representations of different augmented versions of the same image. Initially, it was common to enforce distinct images to have different representations in order to avoid representation collapse [7, 12, 46]. However, subsequent

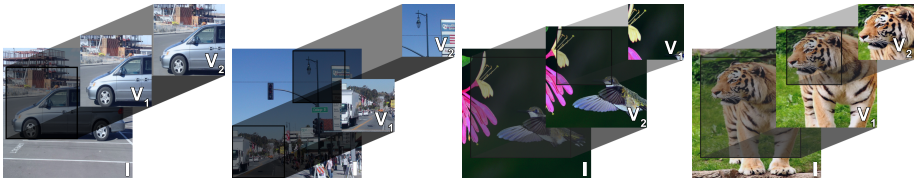
discoveries [20, 24] have shown that robust learning can be achieved even without imposing this constraint. Contrastive self-supervised learning has also been widely used for correspondence learning [26, 45], as it inherently learns to build representations that are invariant and robust to perturbations. Contrary to contrastive learning, CropMAE does not rely as extensively on data augmentations and is not subject to representation collapse issues.

Masked Image Modeling. Drawing inspiration from the field of natural language processing [30], masked image modeling (MIM) techniques have emerged as highly effective learners in the vision domain [3, 23, 49]. This approach involves dividing images into small patches [13], with a high proportion of them being masked, and subsequently reconstructing them using a denoising autoencoder [43]. Notably, after the training phase, the decoder is discarded, leaving the encoder to serve as a feature extractor. MIM has been applied with success across a broad range of fields, and has had numerous extensions and improvements [2, 8, 15–18, 21, 28, 34, 36, 41, 44].

Siamese Masked Autoencoders. Building upon the work of masked autoencoders [23], Siamese Masked Autoencoders (SiamMAE) [21] have emerged as a new state-of-the-art in video propagation tasks such as video object segmentation [35], pose keypoint propagation [27], and semantic part propagation [52]. SiameseMAE uses a Siamese encoder [5] to process either pairs [21] or groups [28] of frames, randomly selected from a video. A key feature of SiameseMAE is its asymmetric masking technique: the initial frame undergoes no masking, thereby serving as a complete reference, while a substantial portion (up to 95%) of the second frame is masked. This setup encourages the network to accurately reconstruct the masked subsequent frames using the fully visible initial frame as a reference. The efficacy of SiameseMAE is believed to stem from its ability to effectively model object motion from videos and visual correspondence, learning the "propagation" and boundaries of objects from their observed positions in the past to their future locations, based on the few visible patches [21]. In this work, we show that explicit motion derived from videos is not mandatory for Siamese masked autoencoders to learn object-centric representations. Particularly, we demonstrate that the ability to recognize object boundaries and acquire propagation skills can be effectively learned from still images.

3 Method

We propose a novel self-supervised method, namely *CropMAE*, capable of learning valuable representations both from images and video frames. First, we create two augmented views (V_1 and V_2) of an input image (I) by randomly cropping, resizing and horizontally flipping the original image (Sec. 3.1). Second, we patchify [13] both views V_1 and V_2 (Sec. 3.2) and mask [23, 30] an extremely high portion of the second view (V_2) (Sec. 3.3). Both views are encoded in a Siamese [5] ViT encoder, with an additional positional embedding [13]. Third, a



(a) Same. **(b)** Random. **(c)** Local-to-Global. **(d)** Global-to-Local.

Fig. 2: Illustration of our four cropping strategies. For a given input image I, we generate an unmasked view V_1 and a masked view V_2 following one of four different cropping strategies: (a) Same Views, where $V_1 = V_2$; (b) Random Views, where V_1 and V_2 are two independent random crops; (c) Local-to-Global, where V_1 is a random crop within V_2, and (d) Global-to-Local, where V_2 is a random crop within V_1.

transformer [19] decoder reconstructs a target image R (Sec. 3.4). The Siamese network and the decoder are trained by minimizing the L2 norm between the target V_2 and the reconstructed image R. After such pre-training, the decoder is discarded, and we use the encoder as a feature extractor on downstream tasks. With this setup, we demonstrate that meaningful data augmentations, particularly random crops, can generate rich and useful object-centric representations for propagation tasks *without* explicit motion. Figure 1 illustrates the main components of our method.

3.1 Cropping

Random crops have been widely used in visual self-supervised learning, especially in contrastive learning, where they are essential to reach excellent performances and develop robust representations [7,9,20]. Specifically, we examine four strategies inspired by the contrastive learning literature [7].

- **Same Views.** This setup corresponds to a direct adaptation of SiamMAE to images, in which the input image I is cropped once and serves both as V_1 and V_2. An illustration is given in Figure 2a.
- **Random Views.** For a given input image I, two independent random cropped views are generated for V_1 and V_2. This setup poses a challenge, particularly when the views are adjacent, *i.e.*, that there is minimal to no overlap between the two crops as illustrated in Figure 2b.
- **Local-to-Global Views.** In this setup, the masked view V_2 is a random crop of the original image I, and the unmasked view V_1 is another random crop of the masked view V_2. An illustration is provided in Figure 2c.
- **Global-to-Local Views.** Inversely, the unmasked view V_1 is a random crop of the original image I, and the masked view V_2 is another random crop of the unmasked view V_1. An illustration is provided in Figure 2d.

Note that our experiments indicate that the Global-to-Local view strategy leads to the best performance.

3.2 Patching

The two views V_1 and V_2 are patched following the original ViT [13]. Specifically, each view is converted into $N \times N$ patches that are fed into the encoder. Similar to SiamMAE, we augment the linear projections of these patches with positional embeddings [42], and append a `[CLS]` token.

3.3 Masking

Since both views are highly spatially redundant, a high masking ratio (above 75%) is usually necessary to create a challenging pretext task and to achieve optimized performances with masked autoencoders [23]. This is even more important in videos where both the spatial and temporal dimensions are highly redundant, requiring even higher masking ratios (90%) [16, 41, 44]. SiamMAE [21] employs a highly asymmetrical masking strategy, where the first frame is left completely visible while the second one is masked at 95%, which corresponds to 9 visible patches out of the 196 available when using a ViT/16 [13]. Using such a high masking ratio encourages the model to propagate the visible patches from the first frame to the second one and to learn temporal correspondences through motion [21]. However, employing a high masking ratio can make some examples ambiguous or may require additional knowledge beyond merely "propagating" patches from the unmasked view. For instance, if an object is only partially visible in the first view, while it is completely present (but masked) in the second one, the task becomes intractable if the model relies solely on the first view to reconstruct it. This prompts the model to learn a conceptual representation of the objects it encounters [23], enabling it to "hallucinate" what it partially sees when propagating past patches is either impossible or insufficiently informative.

Unlike previously introduced MAE methods, CropMAE does not need to learn any conceptual information about objects. Indeed, since our pretext task reconstructs a local view from a global one, there is no ambiguity as the local view is always present within the global view. Provided that the model **(i)** successfully identifies the location of the local view within the global view based on the visible patches and **(ii)** accurately determines the transformations required to reconstruct the local view from the global view, the task is directly tractable based on the inputs that the model receives without any prior conceptual knowledge. This naturally makes the pretext task significantly easier than in other MAE approaches such as MAE [23], VideoMAE [41], or SiamMAE [21], where rich conceptual representations should be used to solve the task. For that reason, we employ an even higher masking ratio. More specifically, our method performs best with only a few visible patches, typically 1 or 2 out of 196, which corresponds to a masking ratio between 98% and 99%. Note that increasing the masking ratio from 95% to 98.5% decreases the number of visible patches by a factor of 4.5, reducing them from 9 to just two visible patches.

3.4 Encoder and Decoder Architectures

Following [21], we use a Siamese ViT [13] encoder to process our two views and a vanilla Transformer [42] composed of cross-attention and self-attention layers as our decoder. Specifically, our decoder alternates between self-attention, where tokens of the masked image attend to each other, and cross-attention layers, where the tokens of the masked image attend to tokens of the visible image. We train the Siamese architecture by minimizing the L2 loss between the normalized [23] pixel values of the view V_2 and the reconstruction R.

4 Experiments

4.1 Experimental setup

Implementation details. Following previous methods [6, 21, 41], we use the ViT-S/16 as encoder architecture [13] for most of our experiments and fair comparisons with respect to other methods in the field. For the decoder, we employ a 4-layer Transformer [42] with a dimension $d_{\text{model}} = 256$, where each block comprises a cross-attention layer, a feed-forward layer (of dimension $d_{\text{ff}} = 2048$), and a self-attention layer. GELU activation functions [25] are utilized alongside a dropout rate of 10% [40]. We use the AdamW [31] optimizer and a base learning rate of $1.5e^{-4}$. The exhaustive list of hyper-parameters that we use can be found in the Appendix.

Baselines. We compare our method with several state-of-the-art methods including MAE-ST [16], MAE [23], VideoMAE [41], and SiamMAE [21]. To the best of our knowledge, no official open-source code is available for SiamMAE, so we reimplemented it to compare the evolution of our performance during training, using the exact same hyperparameters described in the SiamMAE paper (refer to the supplementary material). Our results are consistent with the ones reported in their paper [21]. However, we train for 400 epochs instead of 2000 to save computational resources. Results for longer training can be found in the Appendix.

Datasets. We pre-train our models on Kinetics-400 [29] (K400), on ImageNet [37] (IN), or on a subset of ImageNet (IN Subset). IN Subset contains 239,787 randomly selected images, which corresponds to the number of videos in K400, for fair comparison between methods trained on K400 and ImageNet. During pre-training, we randomly sample an image (or a frame on K400), which is then processed following our methodology described in Section 3.

Downstream tasks. We evaluate our method on three propagation downstream tasks: video object segmentation (DAVIS-2017 [35]), human pose propagation (JHMDB [27]) and semantic part propagation (VIP [52]). These propagation tasks are framed as a semi-supervised problem, where the first annotated frame is provided, and the model is expected to propagate the segmentation mask to subsequent frames.

Table 1: Comparison with prior work. We evaluate our method on three downstream tasks: video object segmentation (DAVIS-2017 [35]), human pose propagation (JHMDB [27]) and semantic part propagation (VIP [52]). Specifically, we compare our method with other methods trained on 400 epochs, on K400 [29] or on our ImageNet [11] Subset (IN Sub) for fair comparison. † refers to results reported in [21]. ‡ refers to our implementation.

				DAVIS			VIP	JHMDB	
Method	Backbone	Dataset	Epochs	$\mathcal{J}\&\mathcal{F}_m$	$\mathcal{J}_m$	$\mathcal{F}_m$	mIoU	PCK@0.1	PCK@0.2
MAE-ST [16] †	ViT-L/16	K400	800	54.6	55.5	53.6	33.2	44.4	72.5
MAE [23] †	VIT-B/16	IN	1600	53.5	52.1	55.0	28.1	44.6	73.4
VideoMAE [41] †	ViT-S/16	K400	800	39.3	39.7	38.9	23.3	41.0	67.9
SiamMAE [21] †	ViT-S/16	K400	2000	**62.0**	**60.3**	**63.7**	**37.3**	**47.0**	**76.1**
SiamMAE [21] ‡	ViT-S/16	K400	400	57.9	56.0	60.0	33.2	**46.1**	**74.0**
CropMAE (ours)	ViT-S/16	K400	400	58.6	55.8	61.4	**33.7**	42.9	71.1
CropMAE (ours)	ViT-S/16	IN Sub	400	**60.4**	**57.6**	**63.3**	33.3	43.6	72.0
CropMAE (ours)	ViT-B/16	IN Sub	400	60.9	57.9	63.8	32.8	44.3	72.3

4.2 Results

We compare our method to previous works and present quantitative results in Table 1. We then provide some qualitative results of the reconstructed image and the downstream tasks respectively in Figures 3 and 4. The first part of Table 1 displays results as reported in their original papers, under optimal training conditions in terms of both training duration and data volume. In the second part, we report the results achieved by our reproduced implementation of SiamMAE and CropMAE under our constrained training: either on K400 or on our ImageNet Subset, for a fixed duration of 400 epochs, and for both ViT-S/16 and ViT-B/16.

When trained for 2,000 epochs on K400, SiamMAE achieves state-of-the-art performances on the three downstream tasks, and outperforms previous MAE methods such as MAE-ST [16], MAE [23] and VideoMAE [41]. However, considering a fixed budget of 400 epochs, CropMAE achieves significantly better results than SiamMAE on DAVIS-2017 [35], both when trained on K400 and on our ImageNet Subset (+0.7% and +2.5% respectively). We believe that by explicitly transforming images through cropping, our pre-training method more quickly understands features useful for segmentation, such as object boundaries. On VIP [52], CropMAE still performs better than SiamMAE, although by a smaller margin (+0.1 when trained on ImageNet, and +0.5 when trained on K400). On JHMDB [27], CropMAE only outperforms VideoMAE. We explain these inferior performances by noting that SiamMAE uses two different frames, resulting in complex human pose modifications, which likely helps the network understand human motion and perform better on JHMDB. Conversely, our random crops do not mimic these transformations. Yet, they help the network learn object boundaries more explicitly, making it more suited for segmentation tasks such as DAVIS.

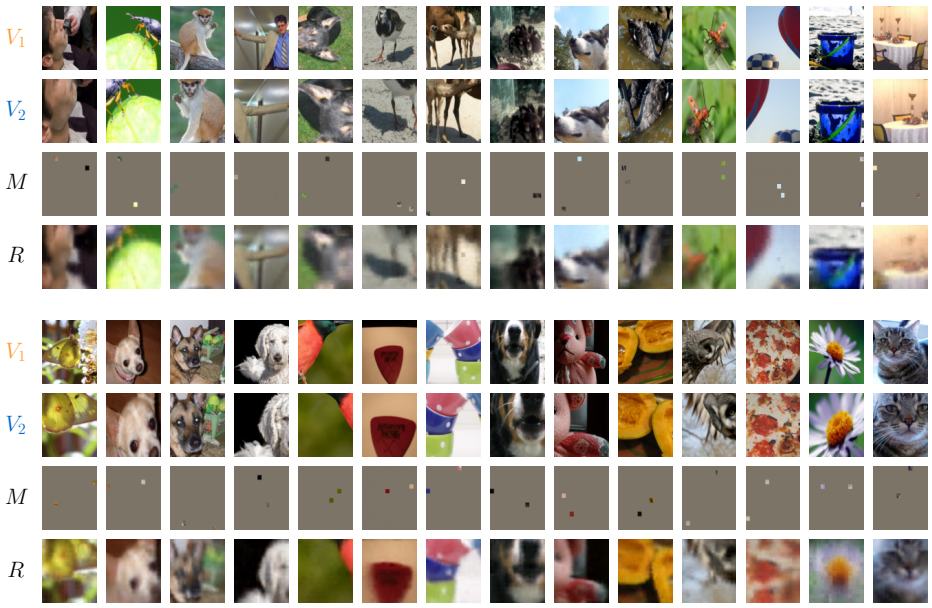


Fig. 3: Reconstructions of CropMAE. We train CropMAE with a ViT-S/16 without normalizing pixel values and a masking ratio of 98.5%. We visualize the reconstructions of some images from ImageNet. The images are displayed in the following order from top to bottom: Input Image (V_1), Random Resized Crop (V_2), Masked Image (M), and Reconstruction (R).

4.3 Attention Maps

In SiamMAE, Gupta *et al.* [21] argue that their model learns the concept of object boundaries through object motion in videos. To support this claim, they present attention maps extracted at some layers of their model, demonstrating that attention predominantly focuses on object boundaries. In a similar way, we train a ViT-S/8 with CropMAE on our ImageNet Subset and visualize the self-attention maps of the [CLS] token from a specific head of the last encoder layer. We show the results in Figure 5. Our findings indicate that our model learns to identify object boundaries as well as SiamMAE without explicit motion (*i.e.*, without relying on video frames). This implies that learning object boundaries is not solely attributable to the motion observed in videos; instead, it can also stem from the transformations and deformations operated on a single image. Hence, this phenomenon is present in both SiamMAE, where it happens naturally between two frames, and in CropMAE, where motion is artificially induced through random cropping. The main difference remains that CropMAE is trained on images instead of videos.

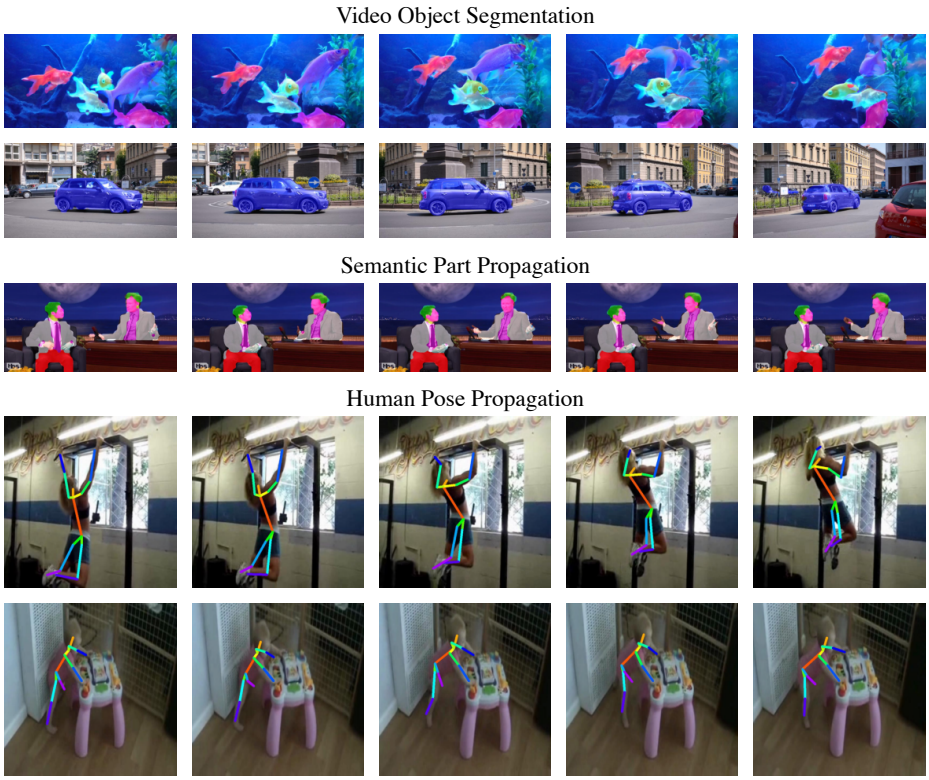


Fig. 4: Qualitative results. We train CropMAE with a ViT-S/16 and qualitatively validate our results on three propagation downstream tasks: video object segmentation (DAVIS-2017 [35]), semantic part propagation [52], and human pose propagation (JHMDB [27]).

4.4 Learning Speed

We evaluate the evolution of the performances of CropMAE and SiamMAE. In particular, we compare SiamMAE trained on K400, CropMAE trained on K400, and CropMAE trained on ImageNet Subset, all for 400 epochs. The performance on the DAVIS-2017 object propagation task [35] is reported every 50 epochs in Figure 6. Remarkably, our approach demonstrates superior performance when trained on the ImageNet Subset compared to training using K400 video frames. This improvement can be attributed to two main factors: **(i)** the greater diversity of the ImageNet dataset, containing a broader spectrum of objects, and **(ii)** its focus on currated object-centric images, which likely results in more relevant crops and reconstruction tasks. In contrast, random cropping in K400 frequently yields images without any objects, diminishing the effectiveness of the learning process.

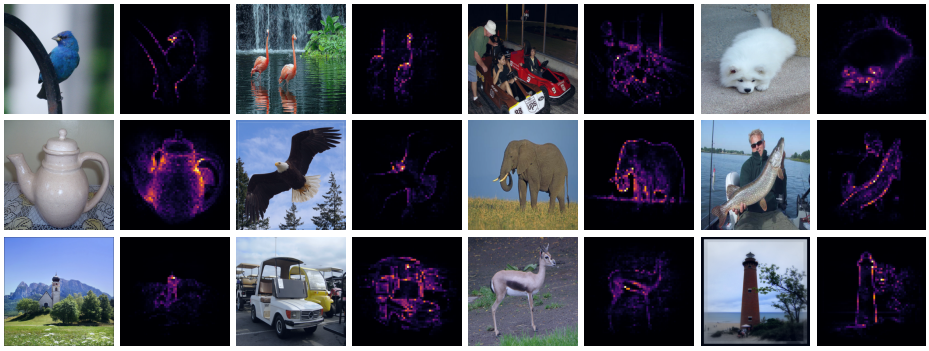


Fig. 5: Self-attention maps from CropMAE with a ViT-S/8 trained on our ImageNet subset. We visualize the self-attention of the `[CLS]` token from a selected head in the last encoder layer of a ViT-S/8, which was trained on our ImageNet subset without using any supervision to learn this specific token. These self-attention maps reveal that our model can learn object boundaries without the need for prior motion information during pre-training.

Our approach demonstrates significantly faster learning than SiamMAE. In particular, our method achieves a $\mathcal{J}\&\mathcal{F}_m$ value of 58.0 after only 150 epochs on our ImageNet Subset and 250 epochs on K400. In contrast, SiamMAE reaches the same performance level after 350 epochs. We attribute this trend to our pretext task, which does not require any conceptual knowledge to be completely tractable and uses object transformations much more explicitly than SiamMAE, leading to faster propagation comprehension. In contrast, SiamMAE must learn the concept of motion and understand object transformations more implicitly between two frames through more complex perturbations such as occlusions and viewpoint changes.

4.5 Training time

We compare the training times of CropMAE and SiamMAE. On the one hand, CropMAE uses an extremely high masking ratio, and only needs a single frame of a video clip to train, or even a standalone image. On the other hand, SiamMAE uses a lower masking ratio and needs two different frames to work. Both these factors significantly impact the training time, as seeking distant frames may require decoding a larger portion of the video, and the number of operations performed by the attention layers increases quadratically with the number of visible patches [23]. We measure the total time taken by both approaches to train and report our results in Table 2. As it can be seen, CropMAE trains almost 30% faster than SiamMAE on K400 for a fixed computational budget, thanks to its use of fewer patches and frames. When pre-training on images (*i.e.*, on the IN Subset), which are significantly faster to decode, CropMAE achieves a tremendous speed-up of 2380% on our hardware while also reaching better performances.

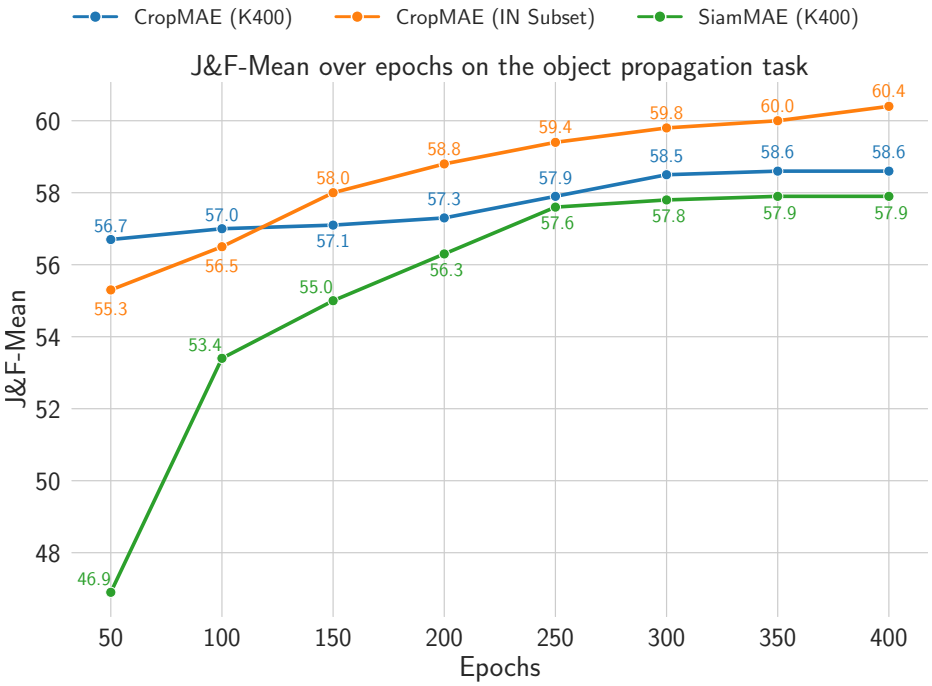


Fig. 6: Performances of CropMAE and SiamMAE on DAVIS during pre-training. For a fixed number of 400 epochs, CropMAE trains faster and consistently yields better results than SiamMAE [21], when trained on K400 frames or ImageNet Subset images.

Table 2: Speedup of CropMAE compared to SiamMAE. We train both methods for 400 epochs on K400, and on ImageNet Sub for CropMAE, and report the speedups observed on the whole training process.

Method	Dataset	Number of images	Mask Ratio	GFLOPS	Speedup
SiamMAE	K400	2	95%	5.8	×1.0
CropMAE	K400	1	98.5%	5.6	×1.29
CropMAE	IN Subset	1	98.5%	5.6	×23.8

4.6 Ablation Studies

We perform several ablation studies on the different components of CropMAE and report the results in Table 3. Unless stated otherwise, we use the default parameters presented in the Appendix. Specifically, we train CropMAE on our ImageNet subset for 400 epochs and report the results obtained on the DAVIS-2017 [35] object propagation task.

Table 3: Ablation Study. We analyze the different components of our method to understand their impact on the downstream performance. We use a ViT-S/16 [13] with the default configuration, as presented in Section 4.1, and report the results obtained on the DAVIS-2017 [35] validation set.

Crop Strategy	$\mathcal{J}\&\mathcal{F}_m$	$\mathcal{J}_m$	$\mathcal{F}_m$
Same Views	36.6	35.8	37.5
Random Views	60.0	57.2	62.8
Local-to-Global	55.9	53.8	58.0
Global-to-Local	**60.4**	**57.6**	**63.3**

(a) Crop Strategy. A simple extension of SiamMAE to images does not work. Reconstructing the local view from the global view works best for CropMAE.

Mask Ratio	$\mathcal{J}\&\mathcal{F}_m$	$\mathcal{J}_m$	$\mathcal{F}_m$
0.75 (49)	45.3	44.3	46.3
0.90 (19)	47.1	46.1	48.0
0.95 (9)	51.2	49.9	52.4
0.985 (2)	**60.4**	**57.6**	**63.3**
0.99 (1)	58.6	55.9	61.5

(b) Mask Ratio and number of visible patches. Our method works best when an extremely large portion of the patches is masked.

Decoder Depth	$\mathcal{J}\&\mathcal{F}_m$	$\mathcal{J}_m$	$\mathcal{F}_m$
2	59.1	56.7	61.6
4	**60.4**	**57.6**	**63.3**
8	57.0	54.5	59.4

(c) Decoder Depth. Our method works best with a small depth.

Decoder Embed Dim	$\mathcal{J}\&\mathcal{F}_m$	$\mathcal{J}_m$	$\mathcal{F}_m$
128	58.5	56.0	61.0
256	**60.4**	**57.6**	**63.3**
384	59.0	56.3	61.7

(d) Decoder Embedding Dimension. Our method works best with a small decoder embedding dimension.

Augmentation	$\mathcal{J}\&\mathcal{F}_m$	$\mathcal{J}_m$	$\mathcal{F}_m$
Color Jitter	56.2	53.1	59.2
Gaussian Blur	59.6	56.7	62.4
None	60.3	57.4	63.2
Horizontal flip	**60.4**	**57.6**	**63.3**

(e) Data Augmentations. Our method works best with horizontal flips randomly applied on both random crops.

Cropping Strategy. We study the effect on performance of different cropping strategies in Table 4a. We can see that reconstructing the same views (Figure 2a) yields very poor performances (36.6), suggesting that the model failed to learn any propagation capabilities. Reconstructing the Local-to-Global view (Figure 2c) results in significantly improved performance (55.9). The Random Views (Figure 2b) and Global-to-Local (Figure 2d) approaches achieve the highest scores (60.0 and 60.4, respectively). Interestingly, these setups are the only ones enabling a completely tractable task without any prior knowledge, meaning the reconstruction can solely rely on the unmasked image. In fact, tractability is *sometimes* guaranteed in the random setting, while it is *always* true for the Global-to-Local approach, which likely explains its superior performance.

Masking Ratio. We examine the importance of the masking ratio in Table 4b. Our method exhibits suboptimal performance at a 75% masking ratio, despite this being the preferred choice for the traditional image MAE framework [23]. Similarly, it underperforms at the 90% ratio used in video frameworks [16, 41, 44]. We can see an improvement with a masking ratio of 95%, as adopted in SiamMAE [21], but the optimal results are reached with a visibility reduced to merely a few patches, *i.e.*, two (60.4) or one (58.6), equivalent of masking ratios of 98.5% and 99%, respectively. We attribute this trend to the fact that our pretext task is simpler than those used in other frameworks as it does not require any conceptual knowledge and can be fully achieved with the help of the visible image, thus requiring an extremely high masking ratio to be challenging.

Decoder Architecture. Next, we study different decoder architectures, specifically their depth and embedding dimension. We report our results in Tables 4c and 4d. Similarly to other MAE works [23,50], we found that the optimal decoder (256-d, 4 blocks) is smaller than the encoder (384-d, 12 blocks).

Data Augmentations. We evaluate our method with additional data augmentations commonly used in contrastive learning [7, 20] and present our results in Table 4e. Similar to SiamMAE [21], we observe that using color jitter significantly reduces performance. The use of Gaussian blur also leads to a decline in performance but to a lesser extent. When we do not apply the random horizontal flip, we observe a minimal drop in performance.

5 Conclusion

In this work, we introduce CropMAE, a self-supervised method for quickly learning rich features for video propagation tasks by reconstructing a crop of an image that has been masked at an extremely high proportion (over 98.5%). We empirically demonstrate that our method can learn useful features for video downstream tasks without requiring explicit video motion. These features can be learned from still images, resulting in even richer information. Thanks to our tractable pretext task, our method trains faster than existing methods and is applicable to both video frames and still images. Finally, we show on-par performances with state-of-the-art methods for three video propagation downstream tasks.

Limitations and future work. Despite being designed to work with small quantities of data and facilitate fast training, we believe the scalability of our method warrants further investigation. This includes both model scalability (*i.e.*, patch size and ViT size) and data scalability (*i.e.*, the amount of data available and the differences between images and video frames). More effort should be directed towards understanding the unique contributions of video frames instead of still images, especially concerning scalability, and determining their necessity to develop rich and robust representations.

Acknowledgements

A. Cioppa is funded by the F.R.S.-FNRS. The research reported in this publication was supported by funding from KAUST Center of Excellence on GenAI, under award number 5940, and the SDAIA-KAUST Center of Excellence in Data Science and Artificial Intelligence. The present research benefited from computational resources made available on Lucia, the Tier-1 supercomputer of the Walloon Region, infrastructure funded by the Walloon Region under the grant agreement n°1910247. We acknowledge EuroCC Belgium for awarding this project access to the LUMI supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CSC (Finland) and the LUMI consortium.

References

1. Balestriero, R., Ibrahim, M., Sobal, V., Morcos, A., Shekhar, S., Goldstein, T., Bordes, F., Bardes, A., Mialon, G., Tian, Y., et al.: A cookbook of self-supervised learning. CoRR **abs/2304.12210** (2023). https://doi.org/10.48550/arXiv.2304.12210
2. Bandara, W.G.C., Patel, N., Gholami, A., Nikkhah, M., Agrawal, M., Patel, V.M.: AdaMAE: Adaptive masking for efficient spatiotemporal learning with masked autoencoders. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 14507–14517. Inst. Electr. Electron. Eng. (IEEE), Vancouver, Can. (Jun 2023). https://doi.org/10.1109/cvpr52729.2023.01394
3. Bao, H., Dong, L., Piao, S., Wei, F.: BEiT: BERT pre-training of image transformers. In: Int. Conf. Learn. Represent. (ICLR) (May 2022), https://openreview.net/forum?id=p-BhZSz59o4
4. Bao, Z., Tokmakov, P., Jabri, A., Wang, Y.X., Gaidon, A., Hebert, M.: Discovering objects that can move. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 11779–11788. Inst. Electr. Electron. Eng. (IEEE), New Orleans, LA, USA (june 2022). https://doi.org/10.1109/cvpr52688.2022.01149
5. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a "siamese" time delay neural network. In: Cowan, J., Tesauro, G., Alspector, J. (eds.) Advances in Neural Information Processing Systems. vol. 6. Morgan-Kaufmann (1993), https://proceedings.neurips.cc/paper_files/paper/1993/file/288cc0ff022877bd3df94bc9360b9c5d-Paper.pdf
6. Caron, M., Touvron, H., Misra, I., Jegou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: IEEE/CVF Int. Conf. Comput. Vis. (ICCV). pp. 9630–9640. Inst. Electr. Electron. Eng. (IEEE), Montreal, QC, Canada (october 2021). https://doi.org/10.1109/iccv48922.2021.00951
7. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: Int. Conf. Mach. Learn. (ICML). Proc. Mach. Learn. Res., vol. 119, pp. 1597–1607 (Jul 2020)
8. Chen, X., Ding, M., Wang, X., Xin, Y., Mo, S., Wang, Y., Han, S., Luo, P., Zeng, G., Wang, J.: Context autoencoder for self-supervised representation learning. Int. J. Comput. Vis. **132**(1), 208–223 (Aug 2023). https://doi.org/10.1007/s11263-023-01852-4

9. Chen, X., He, K.: Exploring simple siamese representation learning. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 15745–15753. Inst. Electr. Electron. Eng. (IEEE), Nashville, TN, USA (Jun 2021). https://doi.org/10.1109/cvpr46437.2021.01549
10. Dave, I., Gupta, R., Rizve, M.N., Shah, M.: TCLR: Temporal contrastive learning for video representation. Comput. Vis. Image Underst. **219**, 1–9 (Jun 2022). https://doi.org/10.1016/j.cviu.2022.103406
11. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 248–255. Inst. Electr. Electron. Eng. (IEEE), Miami, FL, USA (Jun 2009). https://doi.org/10.1109/CVPR.2009.5206848, https://doi.org/10.1109/CVPR.2009.5206848
12. Doersch, C., Zisserman, A.: Multi-task self-supervised visual learning. In: IEEE Int. Conf. Comput. Vis. (ICCV). pp. 2070–2079. Inst. Electr. Electron. Eng. (IEEE), Venice, Italy (Oct 2017). https://doi.org/10.1109/iccv.2017.226
13. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: Int. Conf. Learn. Represent. (ICLR). Austria (May 2021)
14. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (Jun 2010). https://doi.org/10.1007/s11263-009-0275-4
15. Fan, D., Wang, J., Liao, S., Zhu, Y., Bhat, V., Santos-Villalobos, H., M. V., R., Li, X.: Motion-guided masking for spatiotemporal representation learning. In: IEEE/CVF Int. Conf. Comput. Vis. (ICCV). pp. 5596–5606. Inst. Electr. Electron. Eng. (IEEE), Paris, Fr. (october 2023). https://doi.org/10.1109/iccv51070.2023.00517
16. Feichtenhofer, C., fan, h., Li, Y., He, K.: Masked autoencoders as spatiotemporal learners. In: Adv. Neural Inf. Process. Syst. (NeurIPS). vol. 35, pp. 35946–35958. Curran Assoc. Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/e97d1081481a4017df96b51be31001d3-Paper-Conference.pdf
17. Feng, Z., Zhang, S.: Evolved part masking for self-supervised learning. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 10386–10395. Inst. Electr. Electron. Eng. (IEEE), Vancouver, Can. (Jun 2023). https://doi.org/10.1109/cvpr52729.2023.01001
18. Girdhar, R., El-Nouby, A., Singh, M., Alwala, K.V., Joulin, A., Misra, I.: OmniMAE: Single model masked pretraining on images and videos. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 10406–10417. Inst. Electr. Electron. Eng. (IEEE), Vancouver, Can. (Jun 2023). https://doi.org/10.1109/cvpr52729.2023.01003
19. Girdhar, R., Ramanan, D.: Attentional pooling for action recognition. In: Adv. Neural Inf. Process. Syst. (NeurIPS). vol. 30, pp. 1–12. Curran Assoc. Inc., Long Beach, CA, USA (Nov 2017)
20. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., Piot, B., Kavukcuoglu, K., Munos, R., Valko, M.: Bootstrap your own latent – a new approach to self-supervised learning. In: Adv. Neural Inf. Process. Syst. (NeurIPS). vol. 33, pp. 21271–21284. Curran Assoc. Inc. (2020)
21. Gupta, A., Wu, J., Deng, J., Fei-Fei, L.: Siamese masked autoencoders. In: Adv. Neural Inf. Process. Syst. (NeurIPS). vol. 37. Curran Assoc. Inc., New Orleans, LA, USA (2023), https://openreview.net/forum?id=yC3q7vInux

22. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR). vol. 2, pp. 1735–1742. Inst. Electr. Electron. Eng. (IEEE), New York, NY, USA (Jun 2019). https://doi.org/10.1109/cvpr.2006.100
23. He, K., Chen, X., Xie, S., Li, Y., Dollar, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 15979–15988. Inst. Electr. Electron. Eng. (IEEE), New Orleans, LA, USA (Jun 2022). https://doi.org/10.1109/cvpr52688.2022.01553
24. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 9726–9735. Inst. Electr. Electron. Eng. (IEEE), Seattle, WA, USA (Jun 2020). https://doi.org/10.1109/cvpr42600.2020.00975
25. Hendrycks, D., Gimpel, K.: Gaussian error linear units (GELUs). CoRR **abs/1606.08415** (2016). https://doi.org/10.48550/arXiv.1606.08415
26. Jabri, A., Owens, A., Efros, A.A.: Space-time correspondence as a contrastive random walk. In: Adv. Neural Inf. Process. Syst. (NeurIPS). vol. 34. Curran Assoc. Inc. (2020)
27. Jhuang, H., Gall, J., Zuffi, S., Schmid, C., Black, M.J.: Towards understanding action recognition. In: IEEE Int. Conf. Comput. Vis. (ICCV). pp. 3192–3199. Inst. Electr. Electron. Eng. (IEEE), Sydney, NSW, Aust. (Dec 2013). https://doi.org/10.1109/iccv.2013.396
28. Jiang, Z., Wang, B., Xiang, T., Niu, Z., Tang, H., Li, G., Li, L.: Concatenated masked autoencoders as spatial-temporal learner. CoRR **abs/2311.00961** (2023). https://doi.org/10.48550/arXiv.2311.00961
29. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., Zisserman, A.: The kinetics human action video dataset. CoRR **abs/1705.06950** (2017). https://doi.org/10.48550/arXiv.1705.06950
30. Kenton, L., Devlin, J., Chang, M.W., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT. vol. 1, pp. 4171–4186. Minneapolis, Minnesota (Jun 2019)
31. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: Int. Conf. Learn. Represent. (ICLR). New Orleans, LA, USA (May 2019)
32. Misra, I., Zitnick, C.L., Hebert, M.: Shuffle and learn: Unsupervised learning using temporal order verification. In: Eur. Conf. Comput. Vis. (ECCV). Lect. Notes Comput. Sci., vol. 9905, pp. 527–544. Springer Int. Publ. (2016). https://doi.org/10.1007/978-3-319-46448-0_32
33. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: Eur. Conf. Comput. Vis. (ECCV). Lect. Notes Comput. Sci., vol. 9910, pp. 69–84. Springer Int. Publ. (2016). https://doi.org/10.1007/978-3-319-46466-4_5
34. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning robust visual features without supervision. Trans. Mach. Learn. Res. (2024), https://openreview.net/forum?id=a68SUt6zFt
35. Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., Van Gool, L.: The 2017 DAVIS challenge on video object segmentation. CoRR **abs/1704.00675** (2017). https://doi.org/10.48550/arXiv.1704.00675

36. Qing, Z., Zhang, S., Huang, Z., Wang, X., Wang, Y., Lv, Y., Gao, C., Sang, N.: MAR: Masked autoencoders for efficient action recognition. IEEE Trans. Multimedia **26**, 218–233 (2024). https://doi.org/10.1109/tmm.2023.3263288
37. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (Apr 2015). https://doi.org/10.1007/s11263-015-0816-y
38. Sermanet, P., Lynch, C., Chebotar, Y., Hsu, J., Jang, E., Schaal, S., Levine, S., Brain, G.: Time-contrastive networks: Self-supervised learning from video. In: IEEE Int. Conf. Robot. Autom. (ICRA). pp. 1134–1141. Inst. Electr. Electron. Eng. (IEEE), Brisbane, QLD, Australia (May 2018). https://doi.org/10.1109/icra.2018.8462891
39. Spyros, G., Praveer, S., Nikos, K.: Unsupervised representation learning by predicting image rotations. In: Int. Conf. Learn. Represent. (ICLR). Vancouver, Can. (May 2018), https://openreview.net/forum?id=S1v4N2l0-
40. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (Jan 2014)
41. Tong, Z., Song, Y., Wang, J., Wang, L.: VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In: Adv. Neural Inf. Process. Syst. (NeurIPS). vol. 35, pp. 10078–10093. Curran Assoc. Inc. (2022)
42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. CoRR **abs/1706.03762** (2017). https://doi.org/10.48550/arXiv.1706.03762
43. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning - ICML '08. p. 1096–1103. ACM Press, Helsinki, Finland (Jul 2008). https://doi.org/10.1145/1390156.1390294
44. Wang, L., Huang, B., Zhao, Z., Tong, Z., He, Y., Wang, Y., Wang, Y., Qiao, Y.: VideoMAE V2: Scaling video masked autoencoders with dual masking. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 14549–14560. Inst. Electr. Electron. Eng. (IEEE), Vancouver, Can. (Jun 2023). https://doi.org/10.1109/cvpr52729.2023.01398
45. Wang, X., Jabri, A., Efros, A.A.: Learning correspondence from the cycle-consistency of time. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 2561–2571. Inst. Electr. Electron. Eng. (IEEE), Long Beach, CA, USA (Jun 2019). https://doi.org/10.1109/cvpr.2019.00267
46. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR). pp. 3733–3742. Inst. Electr. Electron. Eng. (IEEE), Salt Lake City, UT, USA (Jun 2018). https://doi.org/10.1109/cvpr.2018.00393
47. Xiao, T., Wang, X., Efros, A.A., Darrell, T.: What should not be contrastive in contrastive learning. In: Int. Conf. Learn. Represent. (ICLR). Vienna, Austria (May 2021)
48. Xie, R., Wang, C., Zeng, W., Wang, Y.: An empirical study of the collapsing problem in semi-supervised 2D human pose estimation. In: IEEE/CVF Int. Conf. Comput. Vis. (ICCV). pp. 11220–11229. Inst. Electr. Electron. Eng. (IEEE), Montreal, QC, Canada (Oct 2021). https://doi.org/10.1109/iccv48922.2021.01105
49. Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., Hu, H.: SimMIM: a simple framework for masked image modeling. In: IEEE/CVF Conf. Comput. Vis.

Pattern Recognit. (CVPR). pp. 9643–9653. Inst. Electr. Electron. Eng. (IEEE), New Orleans, LA, USA (Jun 2022). https://doi.org/10.1109/cvpr52688.2022.00943

50. Yao, R., Lin, G., Xia, S., Zhao, J., Zhou, Y.: Video object segmentation and tracking: A survey. ACM Transactions on Intelligent Systems and Technology **11**(4), 36:1–47 (May 2020). https://doi.org/10.1145/3391743
51. Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: iBOT: Image bert pre-training with online tokenizer. In: Int. Conf. Learn. Represent. (ICLR). Vienna, Austria (May 2022), https://openreview.net/forum?id=ydopy-e6Dg
52. Zhou, Q., Liang, X., Gong, K., Lin, L.: Adaptive temporal encoding network for video instance-level human parsing. In: Proceedings of the 26th ACM international conference on Multimedia. p. 1527–1535. ACM (october 2018). https://doi.org/10.1145/3240508.3240660

6.2 EPILOGUE

This chapter demonstrates that a practical alternative to video-based Siamese pretraining is possible: paired crops from still images are sufficient to learn strong object-centric representations. From the perspective of Chapter 5, the method preserves the masked modeling backbone while changing the source of supervisory variation from temporal co-occurrence to geometric view transformations.

Methodologically, the results support three points. First, image-only Siamese pretraining remains competitive on object-centric downstream tasks. Second, removing dependence on video reduces pretraining complexity and wall-clock cost. Third, very high masking ratios remain effective in this regime, reinforcing the information bottleneck emphasized in MAE.

The main boundary is unchanged: motion is not supervised explicitly. This is acceptable for appearance-dominated transfer, but it is insufficient for tasks where temporal dynamics are the primary signal. Chapter 7 therefore extends the same mask-and-predict logic to video and makes motion explicit in both masking strategy and learning target.

7

MOTION-CENTRIC REPRESENTATION LEARNING WITH MASKED VIDEO MODELING

Outline

This chapter presents the following publication: ***Vandeghen, R.***, Thoker, F. M.*, Ghanem, B., and Van Droogenbroeck, M. TrackMAE: Video Representation Learning via Track Mask and Predict. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2026.*
The paper introduces a masked video pretraining strategy that uses explicit motion targets and motion-aware masking to improve motion-centric downstream performance.

7.1 PROLOGUE

Chapter 5 highlighted that masked video modeling is governed by two linked choices: which tokens are hidden and which targets are reconstructed. Chapter 6 showed that, in the image setting, this framework can already yield strong object-centric features with efficient pretraining. The present chapter addresses the complementary regime where temporal dynamics are not optional but central to downstream performance.

Standard masked video formulations mainly reconstruct appearance and capture motion only indirectly, which is a limitation for motion-centric tasks requiring fine-grained temporal reasoning. The main idea of this work is therefore to use motion as a direct supervision signal during pretraining. We introduce TrackMAE, which augments spatial reconstruction with trajectory prediction derived from a point tracker. In parallel, masking is made motion-aware so that visible tokens are allocated across both high-motion and low-motion regions rather than sampled uniformly.

This formulation keeps the simplicity and scalability of mask-and-predict pretraining while increasing temporal supervision density. The goal is not to replace spatial reconstruction but to complement it with explicit motion targets that better match the structure of temporal downstream tasks.

Author contribution

I co-led this work with Fida M. Thoker. I co-designed the method, implemented major components of the training pipeline, ran experiments, and co-wrote the manuscript.

Bernard Ghanem and Marc Van Droogenbroeck supervised the work and contributed to scientific discussion and writing.

TrackMAE: Video Representation Learning via Track Mask and Predict

Renaud Vandeghen* [1,2] Fida Mohammad Thoker* [2] Bernard Ghanem[2] Marc Van Droogenbroeck[1]
[1]University of Liège [2]KAUST
∗ Equal contribution

Abstract

Masked video modeling (MVM) has emerged as a simple and scalable self-supervised pretraining paradigm, but only models motion information implicitly, limiting the encoding of temporal dynamics in the learned representations. As a result, such models struggle on motion-centric tasks that require fine-grained motion awareness. To address this, we propose TrackMAE, a simple masked video modeling paradigm that explicitly uses motion information as a reconstruction signal. In TrackMAE, we use an off-the-shelf point tracker to sparsely track points in the input videos generating motion trajectories. Furthermore, we exploit the extracted trajectories to improve the random tube masking with a motion-aware masking strategy. We enhance video representations learned in both pixel and feature semantic reconstruction space by providing a complimentary supervision signal in the form of motion targets. We evaluate on six datasets across diverse downstream settings and find that TrackMAE consistently outperforms the state-of-the-art video SSL baselines, therefore learning more discriminative and generalizable representations.

1. Introduction

Self-supervised learning (SSL) has become the default pretraining recipe, allowing models to learn diverse yet powerful representations for different modalities, including text [5, 12] image [3, 6–9, 21, 22, 36, 58] and video [23, 49, 51] replacing manual labels with pretext objectives that exploit the structure of the raw data. Among video methods, masked video modeling (MVM) stands out for its simplicity and scalability: a high fraction (often 80–95%) of spatiotemporal tokens is hidden, and a vision transformer is trained to reconstruct them from the visible context [14, 23, 45, 49]. In practice, videos are decomposed into tubelets (*e.g.*, 2×16×16) and only visible tokens are encoded, which makes computation efficient and enables training on large corpora. However, the most common instantiation of MVM is pixel reconstruction with random tube masking, which tends to emphasize low-level appearance statistics (color/texture continuity, local smoothness) and under-utilize temporal structure. Real-world videos also exhibit strong temporal redundancy and sparse foreground motion, so masked pixel reconstruction can often be solved via spatial correlations or short-horizon consistency without learning fine-grained dynamics. This mismatch is observed in the downstream performance of such models, as shown in [46, 47]. They perform well on appearance-dominated datasets like Kinetics-400 [27] or UCF101 [44] but lag on motion-centric tasks such as Something-Something V2 [19] and FineGym[42], where accurate temporal modeling is crucial.

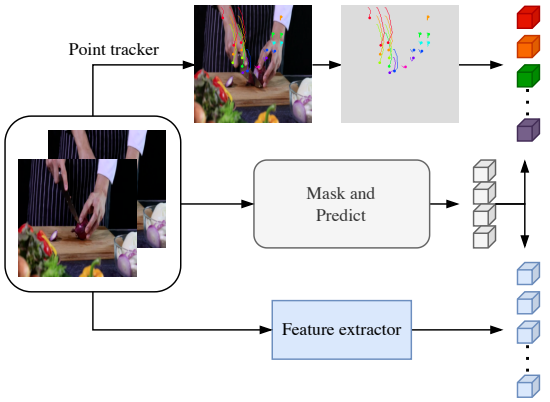


Figure 1. **TrackMAE** improves masked video modeling by jointly predicting spatial features and motion trajectories in a mask-and-predict fashion.

Recent efforts inject more structure into MVM along two orthogonal axes. One line alters *where* to learn by biasing masks toward informative regions, *e.g.*, selecting high-motion tokens via flow [23] or motion vectors [14], or adaptively sampling tokens with learned heuristics [2]. Another line changes *what* to learn by replacing pixels with feature-space targets (*e.g.*, HOG, DINO, CLIP), which mitigates pixel-level shortcuts and encourages learning of high-level semantics like object/part structure, attributes, and ob-

ject–scene relations [45, 48]. While both directions help, they supervise motion only *implicitly*. The model is never asked to predict *how* things move or to maintain identity over time. As a result, improvements in motion sensitivity are limited, especially under high masking ratios where temporal cues must be inferred from few visible tokens. We suggest that temporal correspondence should be a first-hand signal during pretraining, complementing pixel/feature targets rather than competing with them.

In this work, we propose to directly use motion as a training signal by complementing the spatial reconstruction with a motion prediction target. By leveraging the motion prediction produced by a point tracker, our method **TrackMAE** (Fig. 1) aims to jointly reconstruct both the spatial and motion targets. In particular, we use CoTracker3 [26], a robust and high-quality point tracker, to extract motion trajectories. TrackMAE adds two components to masked video modeling: (i) a lightweight *trajectory decoder* that predicts point-track displacements, and (ii) *motion-aware masking* that preferentially samples visible tokens from both high- and low-motion regions. Finally, we show that our motion target complements both pixel- and feature-based spatial targets, leading to on par with or state-of-the-art results on several benchmarks. We summarize our contributions as follows:

1. We propose TrackMAE, a new masked video pretraining scheme with explicit motion awareness based on prediction of tracked point trajectories.
2. We improve the random tube masking strategy with a motion-aware masking that equally samples visible tokens from high- and low-motion regions.
3. Extensive evaluation on multiple datasets demonstrating state-of-the-art performance on motion-centric benchmarks as well as strong generalization capabilities.

2. Related work

Masked video modeling. Masked modeling has emerged as a powerful learning paradigm for representation learning for both visual and non-visual modalities like text [5, 12], audio [10, 18, 24], images [1, 3, 13, 20, 22, 55, 58], and video [14, 16, 23, 45, 49, 50]. The goal is to hide a portion of the input from the model and aim to predict the hidden part from the visible context only. In vanilla masked video modeling [49], a considerable portion of the input pixels are typically kept hidden at random, and the model is trained to predict those hidden pixels from the visible portion, thereby encoding useful video representations. Many follow-up works improve this mask-and-predict task by either leveraging *what to mask* [2, 14, 23, 38] or *what to predict* [31, 40, 45, 48, 54, 56].

In the masking paradigm, MAR [38] designed a new masking strategy tailored for action recognition, based on the running cell masking. MGM [14] leverages the motion information contained in motion vectors of the raw videos to create a motion-aware masking strategy. MGMAE [23] follows the same idea of improving the masking based on motion information extracted from an optical flow instead. AdaMAE [2] learns a sampling network that selects regions with high information, which continually improves the masking strategy over training time.

Beyond the pixel reconstruction paradigm, MaskFeat [54] aims to reconstruct HOG features [11]. In the same manner, MME [45] reconstructs both position changes and the HOG-based shape features. MME builds motion targets based on pre-computed optical flow forming trajectory-like signals, and extracts hand-crafted HOG features around these trajectories for prediction. Such a pipeline requires a heavy pre-processing, camera-motion–sensitive pipeline and produces noisier long-range motion, while we directly extract the trajectories from RGB on the fly. Furthermore, MotionMAE [56] aims to reconstruct temporal frame differences, while recent works like SIGMA [40] rely on clustered DINO [8] features as the reconstruction target. Closely related, SMILE [48] uses CLIP [39] features for reconstruction and improves the motion awareness by injecting synthetic motions by copy-pasting segmented objects onto the videos using randomly generated paths. Different from SMILE, our motion information comes from real trajectories from a tracking module representing actual pixel motion.

Overall, our approach extracts motion trajectories from a tracking module and leverages them as additional motion reconstruction target and reuses them to enhance the motion awareness in the masking, thereby improving the spatial and motion semantics of the learned video representations.

Learning from motion. Motion in video data has long served as free supervision in vision, and prior work exploits it for several tasks. For image representation learning, Wang and Gupta [52] use visual tracking to create positive pairs, encouraging patches that are linked by a track to have similar features. For dense visual correspondence, Wang et al. [53] learn features by tracking backward and then forward in time (cycle consistency), enabling self-supervised correspondence. Li et al. [32] learn dense correspondence by optimizing region tracking and pixel-level matching.

With the emergence of reliable point trackers such as CoTracker3 [26], point tracks have become strong supervisory signals. They can guide attention routing [29] or supervise time-consistent dense features via clustering [41]. Tracktention [29] injects point-track correspondences into the attention layer of image models, yielding temporally consistent features that handle large motion and turning them into strong video models for depth estimation and colorization. Similarly, MoSIC [41] first clusters long-range tracked trajectories via optimal-transport clustering and then propa-

gates the cluster assignments along tracks to enforce temporal coherence under occlusion and viewpoint change, improving dense representations.

Our approach is inspired by these works to use trajectories for motion injection, albeit with a different objective. Instead of temporal consistency or label propagation, we aim to improve the motion semantics in masked video modeling representations.

3. Methodology

In this section, we start by revisiting the masked video modeling frameworks in Sec. 3.1. In Sec. 3.2, we develop our new training scheme, based on motion prediction.

3.1. Masked Video Modeling

Input. Masked video modeling, *e.g*. VideoMAE [49], is an extension of standard image masked modeling MAE [22]. The input is a short clip $\mathbf{V} \in \mathbb{R}^{T\times H\times W\times 3}$, which is partitioned into non-overlapping space-time tubelets $\mathcal{C} = \{\mathbf{c}_i\}_{i=1}^{N}$ of size $t\times p\times p$, where t is the temporal size of the tubelet and p is the spatial patch size (*e.g*., $t=2, p=16$). This yields $N = \frac{T}{t}\cdot\frac{H}{p}\cdot\frac{W}{p}$ tokens of size 3, each corresponding to a local cubic video volume. A tubelet embedding layer (a 3D conv) maps each tubelet in $\mathcal{C}$ into a set of tokens $\mathcal{T} = \{\tau_i\}_{i=1}^{N}$, $\tau \in \mathbb{R}^D$, and a fixed positional embedding is added to preserve spatial and temporal order.

Masking. Masked video modeling randomly hides a large subset of tokens and asks the model to recover them from the remaining context. A high-ratio masking (typically 90%) is applied at the token level following a Bernoulli distribution. This produces a visible set $\mathcal{T}^{visible}$ and a masked set of tokens $\mathcal{T}^{masked}$. Such aggressive masking makes reconstruction non-trivial, forcing the model to capture meaningful spatio-temporal dependencies in the video data.

Architecture. Masked video modeling relies on a standard ViT-based encoder–decoder architecture. The encoder Φ takes the visible set of tokens as the input to produce latent representations $\mathbf{Z} = \Phi(\mathcal{T}^{visible})$. The decoder Ψ then maps the encoder's output to produce reconstruction predictions. To reconstruct the input clip, a complete sequence is formed by inserting learnable `[MASK]` tokens at the masked positions and adding the same positional embeddings used at input. The decoder Ψ takes this complete sequence and predicts the missing content at every position. The goal of the decoder is to predict the space-time tubelets $\hat{\mathcal{C}} = \Psi([\mathbf{Z}, [\texttt{MASK}]])$, of the same size as $\mathcal{C}$.

Reconstruction objectives. Since the main goal is to employ a mask-and-predict task, the reconstruction is purely performed on masked tokens to prevent any information leakage or shortcut solutions by also predicting the visible tokens. In most masked video modeling methods [14, 23, 49], the reconstruction is done in the pixel space. That is, the model is directly optimized to predict the pixel values of masked space-time tubelets from the set of visible tubelets with the following L2 loss

$$\mathcal{L}_{pixel} = \frac{1}{|\mathcal{T}^{masked}|}\sum_{i\in\mathcal{T}^{masked}} \|\mathbf{c}_i - \hat{\mathbf{c}}_i\|_2^2, \qquad (1)$$

where $\hat{\mathbf{c}}_i$ is the i_{th} token in the decoder. To solve this reconstruction task under a strong information dropout, the model has to encode the spatio-temporal dynamics of the input videos, thereby learning useful video representations.

Beyond the pixel space reconstruction, several works reconstruct in more semantic feature spaces *e.g*., HOG [11], DINO [8, 36], or CLIP [39]. Concretely, video frames are processed to extract per-frame descriptors, which are then aligned to the model's space–time tokens. In other words, each space-time pixel tubelet $\mathbf{c}_i$ is projected onto a feature space $\mathcal{F}$, extracting feature tokens $\mathbf{f}_i$ for each tubelet. The model is then optimized to predict the semantic feature values of masked space-time tubelets from the set of visible pixel tubelets, according to the following loss

$$\mathcal{L}_{feature} = \frac{1}{|\mathcal{T}^{masked}|}\sum_{i\in\mathcal{T}^{masked}} \left\|\mathbf{f}_i - \hat{\mathbf{f}}_i\right\|_2^2, \qquad (2)$$

where $\hat{\mathbf{f}}_i$ is again the i_{th} token in the decoder output $\hat{\mathcal{F}} = \Psi(\Phi(\mathcal{T}^{visible}), [\texttt{MASK}])$. Such reconstruction adds an abstraction to the mask-and-predict task, reducing the chances of any shortcuts in pixel reconstruction, and encourages directly modeling high-level video semantics instead of low-level semantics in the pixel space.

3.2. TrackMAE: Learning from Motion

In this section, we introduce TrackMAE, a masked video pretraining framework that leverages tracked trajectories from CoTracker3 [26] in the form of motion prediction and motion-aware masking to enhance temporal awareness in learned video representations. In particular, we make two additions to the vanilla masked modeling paradigm. First, we integrate motion prediction, *i.e*. predicting a sparse set of trajectories as an additional self-supervision task along with the spatial reconstruction. Second, we replace the random uniform masking with a motion-aware masking that keeps visible tokens from high- and low-motion regions using displacement magnitudes from the extracted trajectories.

Extracting motion targets. As mentioned in Sec. 3.1, the input clip $\mathbf{V} \in \mathbb{R}^{T\times H\times W\times 3}$ is patchified into space-time tubelets $\mathcal{C} = \{\mathbf{c}_i\}_{i=1}^{N}$ of size $t\times p\times p$, where p is the spatial patch size. Our goal is to approximate the motion in the given input clip $\mathbf{V}$ by tracking the center pixels of patches in the first frame in the subsequent frames with an off-the-shelf tracking module like CoTracker3 [26]. To that end, we sample query points from a uniform grid of size $G\times G$ for the first frame, where $G = \frac{H}{p}$, and predict their

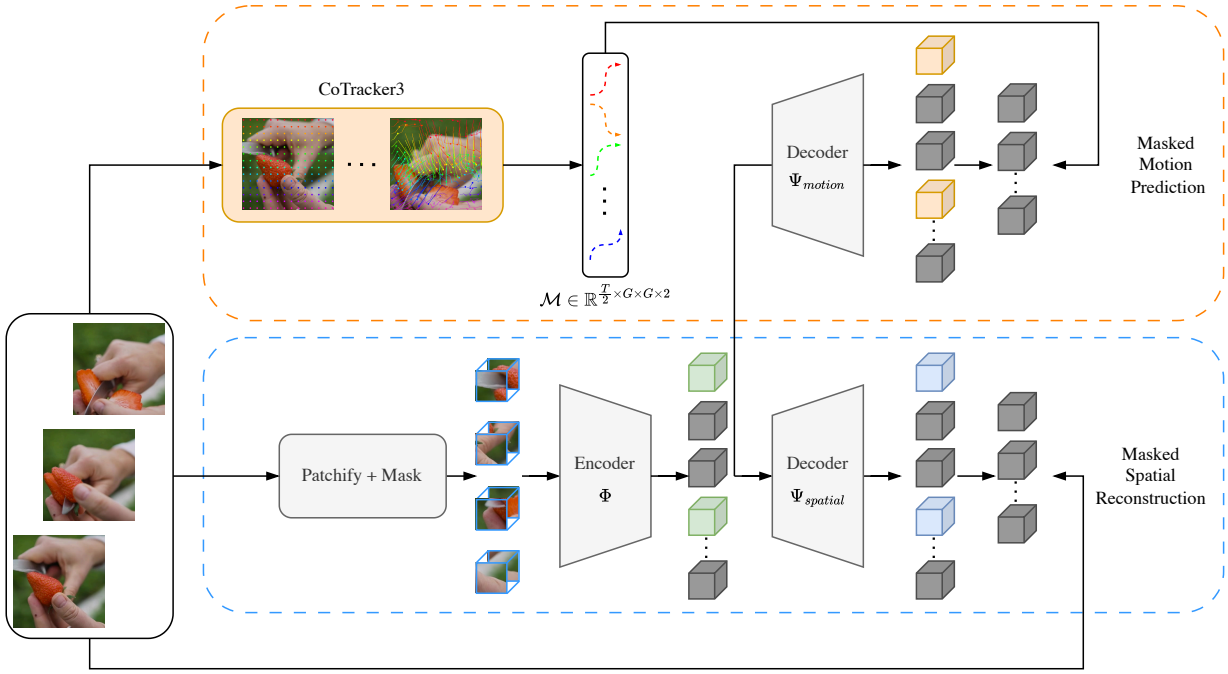


Figure 2. **Overview of TrackMAE.** In the lower branch, a video clip $\mathbf{V}$ is first patchified and masked. The visible tokens are fed to a ViT encoder Φ. Then the decoder $\Psi_{spatial}$ aims to reconstruct spatial features based on the encoder output. In the upper branch, the input video clip is processed by a CoTracker3 module, extracting sparse point trajectories. The encoder output is then passed to a second decoder Ψ_{Motion}, which aims to predict the extracted trajectories. The training objective combines both motion and spatial reconstruction.

2D positions (x, y) in subsequent frames, extracting a set of motion tracks of shape $T \times \frac{H}{p} \times \frac{W}{p} \times 2$. For efficiency and to match the shape of input video tokens, we feed the tracking module every other frame, producing motion tokens $\mathcal{M} = \{\mathbf{m}_i\}_{i=1}^{N}$ of size 2, matching the size of $\mathcal{C}$. Finally, this set of motion tokens $\mathcal{M}$ is used as the reconstruction target. In practice, we predict the displacement of the point tracks instead of absolute trajectory values.

By sparsely tracking only one point per 16 by 16 patch, we may not accurately capture fine motion displacement in the extracted motion tokens. Ideally, we would like to track as many points per patch as possible to generate denser motion targets. However, dense tracking is expensive, with computational cost proportional to the query grid size G. To overcome this issue, we introduce a simple yet effective upsampling trick. Assuming that nearby pixels in a patch behave similarly in terms of motion, we can spatially interpolate extracted sparse motion tokens to simulate denser trajectories per patch. In other words, we can spatially upsample the sparse motion tokens $\mathcal{M}$ into a dense set of motion tokens $\mathcal{U}$ of size $\frac{T}{2} \cdot \frac{\upsilon H}{p} \cdot \frac{\upsilon W}{p}$, where υ is the upsampling factor. This is equivalent to tracking υ^2 points per patch, generating more dense motion targets for reconstruction. We show, in Sec. 4.3, that upsampled targets demonstrate better downstream performance without any added cost.

Architecture. In TrackMAE, our goal is to jointly solve two prediction tasks in a mask-and-predict manner; one is spatial reconstruction, and the other is motion prediction. As in Sec. 3.1, we employ an encoder-decoder framework with a common encoder and two separate decoders. As before, encoder Φ takes visible tokens $\mathcal{T}^{visible}$ as the input to produce latent features $\mathbf{Z}$. The two decoders $\Psi_{spatial}$ and Ψ_{motion} take the encoded features $\mathbf{Z}$ with learnable `[MASK]` tokens and positional embeddings to predict the spatial and motion tokens, respectively, as $\hat{\mathcal{C}} = \Psi_{spatial}([\mathbf{Z}, [\texttt{MASK}]])$ and $\hat{\mathcal{M}} = \Psi_{motion}([\mathbf{Z}, [\texttt{MASK}]])$.

Objectives. In addition to the spatial reconstruction objective, we additionally optimize for the motion prediction. We follow the masked reconstruction strategy to only predict motion tokens of the hidden portion on the input as

$$\mathcal{L}_{motion} = \frac{1}{|\mathcal{T}^{masked}|} \sum_{i \in \mathcal{T}^{masked}} \|\mathbf{m}_i - \hat{\mathbf{m}}_i\|_2^2, \quad (3)$$

which supervises the motion decoder on masked positions only. The final objective is a weighted sum of the two objectives as

$$\mathcal{L} = \mathcal{L}_{spatial} + \lambda * \mathcal{L}_{motion}, \quad (4)$$

where $\mathcal{L}_{spatial}$ is either $\mathcal{L}_{pixel}$ (Eq. (1)) or $\mathcal{L}_{feature}$ (Eq. (2)). As shown in the experiments, our proposed

motion prediction loss $\mathcal{L}_{motion}$ is complementary to both $\mathcal{L}_{pixel}$ and feature reconstruction $\mathcal{L}_{feature}$. The overview of the method is shown in Fig. 2.

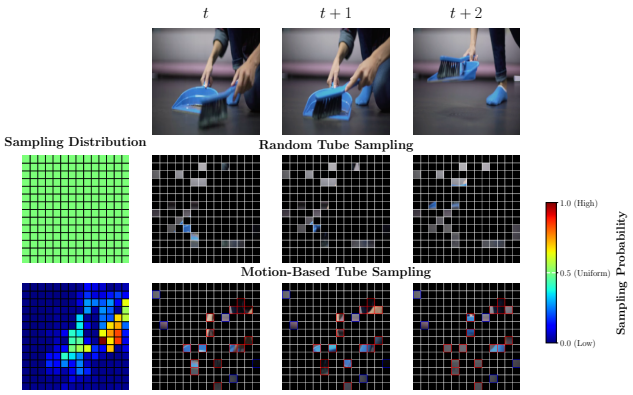


Figure 3. **Masking comparison.** We show how our motion-based tube masking compares to random tube masking. By explicitly sampling visible tokens, our motion-based sampling distribution ensures that visible tokens cover both motion and static regions. In the motion-based tube sampling, **red** squares are sampled from the high-motion and **blue** squares from the low-motion bin.

Motion aware masking. The vanilla tube masking used in VideoMAE does not assume any information of motion, leading to random masking maps. Since the tracker module already gives us motion prediction for trajectory reconstruction, we can also leverage that information to create a motion-guided masking map. In particular, based on the full trajectory predictions, we compute the average displacement $\bar{\mathbf{M}}$, of size $p \times p$, for each query point over the temporal dimension. We then use $\bar{\mathbf{M}}$ as a sampling distribution for the visible tokens. Such distribution can be seen in the first column of the last row in Fig. 3. To sample visible tokens from that distribution, we first create 2 uniform bins, containing high- and low-motion samples. We uniformly sample visible tokens from each bin, using a motion ratio ρ_{motion} to control the number drawn per bin. This masking formulation allows to control where the masking should be done and in what proportion. Such a masking example is depicted in Fig. 3, as well as the random tube masking, which can be expressed as a special case of our sampling distribution, using a uniform distribution instead of $\bar{\mathbf{M}}$.

4. Experiments

In Sec. 4.1, we describe the pretraining setup that we use in our experiments. In Sec. 4.2, we compare our method against prior methods in different linear probing and full finetuning setups. We run some ablation studies in Sec. 4.3 and assess the generalization performance in Sec. 4.4.

4.1. Implementation Details

Following previous methods, we pretrain a video-based ViT-B model on the Kinetics-400 (K400) [27]. We replace the original tube masking with our motion-guided tube masking, using a motion ratio of $\rho_{motion} = 50\%$, and equally balance the losses with λ=1. For pixel reconstruction, we use the default offline CoTracker3 module with a grid size of 14×14, and use upsampling with υ=2. With feature reconstruction, we use a CLIP ViT-B model to extract features. In that setup, we use a grid size of 28×28, without upsampling and without our motion-guided masking. Unless stated otherwise, we follow the same hyperparameters as in [49] and pretrain our model for 800 epochs. More details can be found in the supplementary materials.

Table 1. **Linear probing comparison.** We evaluate TrackMAE on different spatial- and motion-centric benchmarks. For both pixel and feature reconstruction, our method is on par with previous methods for spatial-centric tasks. For motion-centric tasks, our method largely improves previous methods. We report the Top-1% accuracies of ViT-B models pretrained on K400, and highlight in **bold** best and underline second best results.

Method	Target	Spatial-centric		Motion-centric	
		K400	HMDB	SSv2	GYM
Pixel Reconstruction					
VideoMAE [49]	Pixel	20.7	37.7	17.5	23.9
MVD [51]	Pixel	18.7	28.6	12.2	22.7
MGMAE [23]	Pixel	24.9	41.3	16.8	26.1
EVEREST [25]	Pixel	14.1	30.3	14.5	23.3
MGM [14]	Pixel	19.8	40.3	21.7	25.8
TrackMAE (ours)	Pixel	25.7	40.6	23.6	29.0
Feature Reconstruction					
MME [45]	HOG	19.1	37.1	16.6	29.0
SIGMA [40]	DINO	47.5	52.3	20.8	30.1
SMILE [48]	CLIP	**56.2**	**53.4**	23.7	30.2
TrackMAE (ours)	CLIP	55.2	53.1	**27.3**	**31.8**

4.2. Comparison with State-of-the-Art

Linear Probing. To evaluate the quality of learned video representations, we conduct linear probing experiments across four standard action recognition benchmarks. In this setup, the pretrained encoder is frozen, and a linear classifier is trained on top of it. This isolates the effect of pretraining and removes the influence of fine-tuning dynamics, providing a proper measure of representation quality. We compare TrackMAE with prior methods based on both pixel and feature reconstruction. For a fair comparison with prior methods, we evaluate using their publicly released ViT-B checkpoints pretrained on K400. All downstream results are obtained under a unified evaluation protocol.

Table 1 presents the linear probing results across multiple benchmarks. In the pixel reconstruction setting, Track-

Table 2. **Full finetuning comparison.** We evaluate TrackMAE finetuned on SSv2 and K400 after pretraining the models on K400, outperforming all prior works. We highlight in **bold** best and underline second best results.

Method	Targets	SSv2 Top-1	K400 Top-1
VideoMAE [49]	Pixel	68.5	80.0
OmniMAE [17]	Pixel	69.0	80.8
MGM [14]	Pixel	71.1	80.8
MGMAE [23]	Pixel	68.9	81.2
TrackMAE (ours)	Pixel	70.1	80.8
MME [45]	HOG	70.5	81.5
SIGMA [40]	DINO	71.1	81.5
SMILE [48]	CLIP	72.1	83.1
TrackMAE (ours)	CLIP	**72.8**	**83.6**

MAE consistently outperforms the VideoMAE baseline by a margin of ∼5% across all datasets. This affirms our hypothesis that predicting motion provides a strong self-supervisory signal for learning discriminative video representations. Moreover, TrackMAE surpasses other motion-aware pixel reconstruction methods such as MGM [14] and MGMAE [23], particularly on motion-centric datasets like SSv2 and FineGym. This performance gap underscores TrackMAE 's superior ability to encode fine-grained temporal dynamics. In the feature reconstruction setting, TrackMAE again yields consistent gains and outperforms all prior methods, including the state-of-the-art SMILE for the motion-centric tasks. These results demonstrate that motion prediction is not only effective in isolation but also complements semantically rich targets (*e.g.*, CLIP features), enhancing representation learning beyond the pixel space. In summary, these findings confirm that our approach leads to more temporal awareness in the learned representations.

Full Finetuning. To fully leverage the learned spatiotemporal representations, we perform end-to-end finetuning of both the pretrained backbone and the classification head on downstream datasets. This setup is crucial for assessing the transferability and task-specific adaptability of representations learned through our trajectory-guided pretraining. In contrast to linear probing, which freezes the encoder, full finetuning enables the model to refine its temporal and semantic understanding based on the target task distribution. We evaluate our method on two representative benchmarks: Kinetics-400 and Something-Something V2 (SSv2), covering both appearance- and motion-focused benchmarks. We use the same evaluation protocol as the prior masked video modeling works [14, 23, 49], ensuring a fair comparison. Further details, are provided in the supplementary material.

We evaluate TrackMAE under two standard transfer regimes: in-domain transfer (pretraining and finetuning on the same dataset *i.e.*, K400) and cross-domain transfer (pretraining on K400 and finetuning on SSv2). The results, summarized in Tab. 2, show that TrackMAE consistently achieves strong performance across both settings. As with linear probing, we again observe that our TrackMAE with pixel reconstruction improves the VideoMAE baseline by +0.8% for in-domain transfer, and +1.6% for cross-domain transfer. Moreover, TrackMAE with pixel targets is on par or outperforms other prior methods with pixel targets, OmniMAE (1.1% on SSv2) MGMAE (1.2% on SSv2), even in full-finetuning settings. This again validates that adding our motion prediction task to pixel reconstruction is effective for learning better transferable video representations.

Finally, TrackMAE with CLIP targets outperforms all other prior methods, achieving state-of-the-art performance in all settings. In particular, TrackMAE outperforms SMILE, which also uses CLIP targets along with synthetic motion infusion for learning motion-aware representations by 0.7% and 0.5% on SSv2 and K400, demonstrating better motion encoding without the need for any synthetic motion priors. To summarize, adding motion prediction targets complements the spatial reconstruction targets to enrich the learned video representations for both appearance and motion-focused downstream tasks even in full-finetuning settings. We show a more detailed SOTA comparison with other pretraining settings in the supplementary material.

Table 3. **Reconstruction targets.** Trajectory reconstruction provides a strong supervisory signal to learn useful video representations and complements both pixels and CLIP reconstruction, consistently improving the downstream results.

Reconstruction Targets	$K400_s$	$SSv2_s$
Trajectory only	46.5	53.1
Pixels	46.0	52.2
Pixels + Trajectory	**48.9**	**55.7**
CLIP	52.7	57.1
CLIP + Trajectory	**55.8**	**61.1**

4.3. Ablations

To assess the contribution of each design component in our TrackMAE framework, we conduct a series of ablation experiments, as summarized in Tabs. 3 to 6. For computational efficiency, we adopt the ViT-S architecture and pretrain on a subset of Kinetics-400, referred to as $K400_s$, which includes $1/3$ of the total training videos. Evaluation is performed on both $K400_s$ and $SSv2_s$, also a reduced version of SSv2. By default we use pixel and motion trajectories as reconstruction targets, employ a grid size of 14×14, use random tube masking, set λ=1, and train for 200 epochs, unless specified otherwise.

Impact of reconstruction targets. Table 3 analyzes the impact of the different target reconstructions used. We first observe that trajectory reconstruction as a standalone task

Table 4. **Masking strategy.** Replacing random tube masking with our motion-based masking consistently improves the results for both pixels and pixels + trajectory reconstruction.

Target	Masking	$K400_s$	$SSv2_s$
Pixel	Tube	46.0	52.2
Pixel	Motion-aware	**46.6**	**52.6**
Pixel + Trajectory	Tube	48.9	55.7
Pixel + Trajectory	Motion-aware	**49.4**	**56.2**

Table 5. **Impact of dense tracking.** Increasing the tracker grid size consistently improves the results, but at a higher computational cost. Upsampling the tracker's prediction from 14×14 to 28×28 yields better performance without any added cost.

Grid Size	Upsampling	$K400_s$	$SSv2_s$
14×14	None	48.9	55.7
28×28	None	49.5	56.7
56×56	None	**50.0**	**57.0**
14×14	$14 \rightarrow 28$ (v=2)	**50.6**	**57.6**
14×14	$14 \rightarrow 56$ (v=4)	50.4	57.4

is very effective for learning useful video representations, indicating that it can act as a strong self-supervisory signal. Next, we observe that combining trajectory reconstruction with pixel reconstruction outperforms both trajectory-only and pixel-only reconstruction by ($+2.4\%$ and $+2.9\%$) on $K400_s$ and ($+2.6\%$ and $+3.5\%$) on $SSv2_s$, respectively. Finally, when combined with more high-level semantic targets like CLIP instead of pixel targets, we observe a significant improvement of ($+2.9\%$ and $+4.0\%$). One of the reasons for this is that trajectory targets are closer to pixel targets in the sense that they represent movement of pixels and might rely on the same semantics for solving the reconstruction tasks. On the other hand, CLIP targets are highly semantic and largely encode what is present but not how things move, and trajectory prediction fills that gap with a more complimentary reconstruction task. In summary, this clearly indicates that exploiting motion trajectories as a complementary target strengthens video representation learning.

Impact of motion-aware masking. Table 4 shows the impact of our proposed motion-aware masking over random tube masking. We compare our motion-aware masking against random tube masking under two regimes: pixel-only and pixel+trajectory reconstruction. Motion-aware masking yields consistent gains of roughly +0.5% at no extra computation, as the same trajectories used for supervision are repurposed to construct the masking prior.

Impact of dense tracking. As mentioned in Sec. 3.2, we track one point per patch from the first frame and initialize query points accordingly on a coarse grid of size 14×14 to extract sparse motion trajectories. In this ablation, we show the impact of extracting and predicting denser trajectories.

Table 6. **Motion ratio masking.** Equally sampling (50%) visible tokens from the high motion and low motion bins yields the best results compared to asymmetric sampling.

ρ_{motion}	$K400_m$	$SSv2_m$
25	48.7	55.4
50	**49.4**	**56.2**
75	49.0	55.9

Table 7. **Balancing the losses.** Equally balancing the loss ($\lambda = 1.0$) yields the best results compared to unbalanced setups.

λ	$K400_s$	$SSv2_s$
0.1	47.1	54.0
0.5	48.2	54.6
1.0	**48.9**	**55.7**
2.0	48.8	55.7

In particular, we initialize query points on denser grids (28×28 and 56×56) to extract denser trajectories. The results in Tab. 5 show that moving from a coarse to denser prediction consistently improves the results.

However, these gains come at a cost, since the computational cost of CoTracker3 is directly proportional to the query grid size. Table 5 also shows that by upsampling the sparse trajectories to denser trajectories via spatial interpolation from $14 \rightarrow 28$ (v=2) yields the strongest gains on both datasets at no cost ($+1.7\%$ and $+2.0\%$). However, upsampling from $14 \rightarrow 56$ (v=4) does not improve over $14 \rightarrow 28$ (v=2) setting. One of the reasons could be that spatial interpolation exploits the piecewise-smooth nature of local motion and densifies supervision at token locations, strengthening gradients without introducing tracker noise.

Impact of ρ_{mask}. Table 6 indicates how our motion-guided masking behaves when we vary the proportion of masked tokens sampled from high- and low-motion regions. Sampling too few (25%) or too many (75%) motion locations slightly degrades performance compared to 50% motion ratio, showing that equally biasing the mask towards both motion and static parts is a good balance.

Impact of λ. Table 7 shows that using the same weight for the losses yields the best results, indicating that both signals are useful during training.

4.4. Downstream Generalization on SEVERE

Next, following [40, 48], we assess the robustness of TrackMAE beyond standard action recognition on the SEVERE benchmark proposed in [46] and extended in [47]. It consists of a controlled suite of eight evaluations designed to probe four aspects of generalization: *domain shift*, *sample efficiency*, *action granularity*, and *task shift*. Concretely, we measure transfer under distribution shift on Something-Something V2 and FineGym (Gym99), low-data finetuning with 1K examples on UCF101 and FineGym, fine-grained discrimination on FX-S1 and UB-S1 splits of FineGym, and non-standard objectives via temporal repetition counting on UCFRep [57] and multi-label classification on Charades [43]. All experiments follow the official SEVERE protocols and are reported in Tab. 8. Implementation and training details are reported in the supplementary material.

Domain shift. On SSv2 and Gym99, TrackMAE with

Table 8. **Comparison on SEVERE generalization benchmark [46].** We compare prior video SSL methods on four generalization factors of SEVERE benchmark spanning a total of eight downstream settings. TrackMAE delivers consistently strong or superior performance across these diverse settings in both pixel and feature reconstruction modes, indicating that trajectory-guided masked pretraining improves robustness and generalization of learned video representations.

Method	Domain shift		Sample efficiency (10^3)		Action granularity		Task shift		Mean
	SSv2	Gym99	UCF	GYM	FX-S1	UB-S1	UCF-RC↓	Charades	
Pixel Reconstruction									
VideoMAE [49]	68.6	86.6	74.6	25.9	42.8	65.3	0.172	17.8	57.6
MVD [51]	70.0	82.5	67.1	17.5	31.3	50.5	0.184	16.1	52.1
MGMAE [23]	68.9	87.2	77.2	24.1	33.7	79.5	0.181	17.9	58.8
MGM [14]	71.1	89.1	78.4	26.4	38.6	86.9	**0.152**	22.5	62.2
TrackMAE	70.3	88.7	79.8	31.0	41.6	85.5	0.162	20.8	62.9
Feature Reconstruction									
MME [45]	70.1	89.7	79.2	29.8	55.5	87.2	0.155	23.6	65.0
SIGMA [40]	70.9	89.7	84.1	28.0	55.1	79.9	0.169	23.1	64.2
SMILE [48]	72.1	90.8	86.4	**35.1**	55.1	88.3	0.170	**32.5**	67.9
TrackMAE w/o motion	71.9	90.0	85.5	31.4	55.1	74.6	0.170	27.1	64.8
TrackMAE	**72.8**	**91.1**	**86.7**	34.4	**59.0**	**90.1**	0.170	30.5	**68.4**

CLIP+motion targets attains the strongest performance among masked video modeling approaches, slightly improving over SMILE and other feature-based baselines. Importantly, even the pixel-only TrackMAE variant improves over VideoMAE and remains competitive with motion-aware designs such as MGM and MGMAE, showing that explicit trajectory prediction enhances robustness to distribution shifts even when training purely in pixel space.

Sample efficiency. In the 1K-sample setting (UCF10^3, Gym10^3), TrackMAE maintains strong performance, confirming that trajectory-guided pretraining produces features that adapt well under limited supervision. The pixel-based TrackMAE variant already surpasses its pixel-only counterparts, indicating that injecting motion structure into masked prediction benefits low-shot recognition without relying solely on high-level feature targets.

Action granularity. For fine-grained splits FX-S1 and UB-S1, TrackMAE with CLIP+motion achieves the best results, and the pixel-based TrackMAE variant also improves over or is on par with motion-guided baselines like MGM and MGMAE. These gains underline that TrackMAE is particularly effective for capturing subtle temporal and spatial differences required in fine-grained action classification.

Task shift. For the task shift, TrackMAE shows improvements over baselines VideoMAE and MGMAE in the pixel space but is on par or slightly worst than the current best method SMILE in the feature reconstruction.

Summary. In the pixel space, TrackMAE outperforms all prior works with a significantly improving mean results by +5.3% over VideoMAE and by +4.1% over MGMAE, showing strong generalization capability. Furthermore, adding our motion prediction targets to CLIP-only reconstruction (denoted as TrackMAE w/o motion) results in a mean improvement of 3.4%, indicating the generalization capability is infact enhanced by the prediction of motion trajectories. Finally, TrackMAE with CLIP targets improves the previous state-of-the-art mean results by 0.5%.

5. Discussions

Computational cost. Point tracking comes at a non-negligible cost. In practice, we observe that the pretraining time increases by 50%. However, with stronger performance on linear probing and generalization, the tradeoff is reasonable. Furthermore, we are able to mitigate larger cost due to denser tracking with our upsampling strategy enabling our method to extract motion targets on the fly in a scalable manner.

Motion robustness. Point trackers are prone to incorrect prediction, often due to very high motion or occlusion. To evaluate the robustness of our method under jittered trajectories, we trained our model to reconstruct noisy predictions, either spatially or temporally, using Gaussian noise. In that setting, the performance degradation is around -0.5%, showing that even under noisy motion targets, our method is still able to learn meaningful motion-based representations.

6. Conclusion

We introduce TrackMAE, a new masked video modeling paradigm based on motion prediction. In particular, we leverage a sparse point tracker to a create motion-based reconstruction signal. To mitigate the computational cost needed to obtain denser trajectories, we show that spatially interpolating the tracker output yields better performance without additional cost. We also use the motion trajectories to propose a new motion-aware masking strategy that further improves the downstream performance. Across linear

probing, full fine-tuning, and SEVERE, TrackMAE shows consistent gains on appearance- and motion-centric tasks, translating into stronger discrimination and broader generalization than prior video SSL methods.

Acknowledgments

The present research benefited from computational resources made available on Lucia, the Tier-1 supercomputer of the Walloon Region, infrastructure funded by the Walloon Region under the grant agreement n°1910247. The research reported in this publication was supported by funding from King Abdullah University of Science and Technology (KAUST) - Center of Excellence for Generative AI, under award number 5940. For computing time, this research used Ibex managed by the Supercomputing Core Laboratory at King Abdullah University of Science & Technology (KAUST) in Thuwal, Saudi Arabia. We acknowledge EuroPC JU for awarding the project ID EHPC-DEV-2025D10-008 access to Leonardo on Leonardo Booster hosted by CINECA, Italy and access to MareNostrum5 on MN5 ACC hosted by BSC, Spain.

References

[1] 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 *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 15619–15629, Vancouver, Can., 2023. 2

[2] Wele Gedara Chaminda Bandara, Naman Patel, Ali Gholami, Mehdi Nikkhah, Motilal Agrawal, and Vishal M. Patel. AdaMAE: Adaptive masking for efficient spatiotemporal learning with masked autoencoders. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14507–14517, Vancouver, Can., 2023. 1, 2

[3] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–18, Virtual conference, 2022. 1, 2

[4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Int. Conf. Mach. Learn. (ICML)*, pages 813–824. ML Res. Press, 2021. 2

[5] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hessei, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Sutskever Ilya, and Dario Amodei. Language models are few-shot learners. *arXiv*, abs/2005.14165, 2020. 1, 2

[6] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 139–156, 2018. 1

[7] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 9912–9924, Virtual conference, 2020. Curran Assoc. Inc.

[8] Mathilde Caron, Hugo Touvron, Ishan Misra, Herve Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 9630–9640, Montréal, Can., 2021. 2, 3

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Int. Conf. Mach. Learn. (ICML)*, pages 1597–1607, 2020. 1

[10] Dading Chong, Helin Wang, Peilin Zhou, and Qingcheng Zeng. Masked spectrogram prediction for self-supervised audio pre-training. In *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, pages 1–5, Rhodes Island, Greece, 2023. 2

[11] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 886–893, San Diego, CA, USA, 2005. 2, 3

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. Conf. North Am. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, pages 4171–4186, Minneapolis, MN, USA, 2019. Assoc. Comput. Linguistics. 1, 2

[13] Alexandre Eymaël, Renaud Vandeghen, Anthony Cioppa, Silvio Giancola, Bernard Ghanem, and Marc Van Droogenbroeck. Efficient image pre-training with siamese cropped masked autoencoders. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 348–366, 2024. 2

[14] David Fan, Jue Wang, Shuai Liao, Yi Zhu, Vimal Bhat, Hector Santos-Villalobos, Rohith M V, and Xinyu Li. Motion-guided masking for spatiotemporal representation learning. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 5596–5606, Paris, Fr., 2023. 1, 2, 3, 5, 6, 8

[15] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 6804–6815, Montréal, Can., 2021. 2

[16] Christoph Feichtenhofer, haoqi fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 35946–35958, New Orleans, LA, USA, 2022. Curran Assoc. Inc. 2

[17] Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. OmniMAE: Single model masked pretraining on images and videos. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10406–10417, Vancouver, Can., 2023. 6, 1, 2

[18] Yuan Gong, Cheng-I Lai, Yu-An Chung, and James Glass. SSAST: Self-supervised audio spectrogram transformer. In *AAAI Conf. Artif. Intell.*, pages 10699–10709, Virtual venue, 2022. 2

[19] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 5843–5851, Venice, Italy, 2017. 1, 4

[20] Agrim Gupta, Jiajun Wu, Jia Deng, and Li Fei-Fei. Siamese masked autoencoders. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 1–18, New Orleans, LA, USA, 2023. Curran Assoc. Inc. 2

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9726–9735, Seattle, WA, USA, 2020. 1

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollar, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conf. Comput. Vis. Pattern*

Recognit. (CVPR), pages 15979–15988, New Orleans, LA, USA, 2022. 1, 2, 3

[23] Bingkun Huang, Zhiyu Zhao, Guozhen Zhang, Yu Qiao, and Limin Wang. MGMAE: Motion guided masking for video masked autoencoding. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 13447–13458, Paris, Fr., 2023. 1, 2, 3, 5, 6, 8

[24] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 28708–28720, New Orleans, LA, USA, 2022. Curran Assoc. Inc. 2

[25] Sunil Hwang, Jaehong Yoon, Youngwan Lee, and Sung Ju Hwang. EVEREST: Efficient masked video autoencoder by removing redundant spatiotemporal tokens. In *Int. Conf. Mach. Learn. (ICML)*, pages 20889–20907, Vienna, Austria, 2024. 5

[26] Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker3: Simpler and better point tracking by pseudo-labelling real videos. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 6013–6022, Honolulu, HI, USA, 2025. 2, 3, 4

[27] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *arXiv*, abs/1705.06950, 2017. 1, 5, 4

[28] Hilde Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso A. Poggio, and Thomas Serre. HMDB: A large video database for human motion recognition. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2556–2563, Barcelona, Spain, 2011. 4

[29] Zihang Lai and Andrea Vedaldi. Tracktention: Leveraging point tracking to attend videos faster and better. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 22809–22819, Nashville, TN, USA, 2025. 2

[30] Kunchang Li, Yali Wang, Gao Peng, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. UniFormer: Unified transformer for efficient spatial-temporal representation learning. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–19, Virtual conference, 2022. 1, 2

[31] Kunchang Li, Yali Wang, Yizhuo Li, Yi Wang, Yinan He, Limin Wang, and Yu Qiao. Unmasked teacher: Towards training-efficient video foundation models. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 19891–19903, Paris, Fr., 2023. 2

[32] Xueting Li, Sifei Liu, Shalini De Mello, Xiaolong Wang, Jan Kautz, and Ming-Hsuan Yang. Joint-task self-supervised learning for temporal correspondence. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 1–11, Vancouver, Can., 2019. Curran Assoc. Inc. 2

[33] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. MViTv2: Improved multiscale vision transformers for classification and detection. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 4794–4804, New Orleans, LA, USA, 2022. 1, 2

[34] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3192–3201, New Orleans, LA, USA, 2022. 1, 2

[35] Cheng-Ze Lu, Xiaojie Jin, Zhicheng Huang, Qibin Hou, Ming-Ming Cheng, and Jiashi Feng. CMAE-V: Contrastive masked autoencoders for video action recognition. *arXiv*, abs/2301.06018, 2023. 1, 2

[36] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 1:1–32, 2024. 1, 3

[37] Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and João F. Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 12493–12506, Virtual conference, 2021. Curran Assoc. Inc. 2

[38] Zhiwu Qing, Shiwei Zhang, Ziyuan Huang, Xiang Wang, Yuehuan Wang, Yiliang Lv, Changxin Gao, and Nong Sang. MAR: Masked autoencoders for efficient action recognition. *IEEE Trans. Multimedia*, 26:218–233, 2024. 2

[39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Int. Conf. Mach. Learn. (ICML)*, pages 8748–8763, Virtual Conf., 2021. ML Res. Press. 2, 3

[40] Mohammadreza Salehi, Michael Dorkenwald, Fida Mohammad Thoker, Efstratios Gavves, Cees G. M. Snoek, and Yuki M. Asano. SIGMA: Sinkhorn-guided masked video modeling. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 293–312, 2024. 2, 5, 6, 7, 8, 1, 4

[41] Mohammadreza Salehi, Shashanka Venkataramanan, Ioana Simion, Efstratios Gavves, Cees G. M. Snoek, and Yuki M. Asano. MoSiC: Optimal-transport motion trajectory for dense self-supervised learning. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Honolulu, HI, USA, 2025. 2

[42] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. FineGym: A hierarchical video dataset for fine-grained action understanding. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2613–2622, Seattle, WA, USA, 2020. 1, 4, 5

[43] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 510–526, 2016. 7, 5

[44] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv*, abs/1212.0402, 2012. 1, 4, 5

[45] Xinyu Sun, Peihao Chen, Liangwei Chen, Changhao Li, Thomas H. Li, Mingkui Tan, and Chuang Gan. Masked motion encoding for self-supervised video representation learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2235–2245, Vancouver, Can., 2023. 1, 2, 5, 6, 8

[46] Fida Mohammad Thoker, Hazel Doughty, Piyush Bagad, and Cees G. M. Snoek. How severe is benchmark-sensitivity in video self-supervised learning? In *Eur. Conf. Comput. Vis. (ECCV)*, pages 632–652, 2022. 1, 7, 8, 4, 5

[47] Fida Mohammad Thoker, Letian Jiang, Chen Zhao, Piyush Bagad, Hazel Doughty, Bernard Ghanem, and Cees G. M. Snoek. SEVERE++: Evaluating benchmark sensitivity in generalization of video representation learning. *arXiv*, abs/2504.05706, 2025. 1, 7

[48] Fida Mohammad Thoker, Letian Jiang, Chen Zhao, and Bernard Ghanem. SMILE: Infusing spatial and motion semantics in masked video learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 8438–8449, Nashville, TN, USA, 2025. 2, 5, 6, 7, 8, 1, 4

[49] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 10078–10093, New Orleans, LA, USA, 2022. Curran Assoc. Inc. 1, 2, 3, 5, 6, 8, 4

[50] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. VideoMAE V2: Scaling video masked autoencoders with dual masking. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14549–14560, Vancouver, Can., 2023. 2

[51] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Lu Yuan, and Yu-Gang Jiang. Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 6312–6322, Vancouver, Can., 2023. 1, 5, 8

[52] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2794–2802, Santiago, Chile, 2015. 2

[53] Xiaolong Wang, Allan Jabri, and Alexei A. Efros. Learning correspondence from the cycle-consistency of time. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2561–2571, Long Beach, CA, USA, 2019. 2

[54] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14648–14658, New Orleans, LA, USA, 2022. 2

[55] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. SimMIM: a simple framework for masked image modeling. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9643–9653, New Orleans, LA, USA, 2022. 2

[56] Haosen Yang, Bin Huang, Deng andi Wen, Jiannan Wu, Hongxun Yao, Yi Jiang, Xiatian Zhu, and Zehuan Yuan. MotionMAE: Self-supervised video representation learning with motion-aware masked autoencoders. In *Br. Mach. Vis. Conf. (BMVC)*, pages 1–14, Glasgow, UK, 2024. Br. Mach. Vis. Assoc. (BMVA). 2

[57] Huaidong Zhang, Xuemiao Xu, Guoqiang Han, and Shengfeng He. Context-aware and scale-insensitive temporal repetition counting. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 667–675, Seattle, WA, USA, 2020. 7, 5

[58] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–29, Virtual conference, 2022. 1, 2

TrackMAE: Video Representation Learning via Track Mask and Predict

Supplementary Material

7. Detailed SOTA Comparison

In the main paper, we restricted comparisons to self-supervised masked video models with the same backbone and similar pretraining schedule. We now broaden that comparison to include both self-supervised (MVM variants with pixels/HOG/DINO/CLIP targets) and supervised baselines on K400 and SSv2, reported with the ViT-B backbone and a range of pretraining epochs. We evaluate in the full finetuning setup on K400 and SSv2 datasets to assess in-domain and cross-domain transfer. The results are shown in Table 9.

Kinetics-400 (in-domain pretraining and finetuning). On K400, TrackMAE attains 83.6 Top-1 with 600 epochs, outperforming all listed masked-video baselines trained for equal or longer schedules: VideoMAE [49] (80.0/81.5 at 800/1600), CMAE-V [35] (80.2/80.9), OmniMAE (80.8), MGM (80.8/81.7), MGMAE [23] (81.2/81.8), MME [45] (81.5/81.8), and SIGMA [40] (81.5). TrackMAE also edges SMILE [48] (83.1 at 600 and 83.4 at 1200), indicating a stronger accuracy–compute trade-off at shorter schedules. Compared to supervised architectures, TrackMAE surpasses MViTv2-B [33] (82.9) and Uniformer-B [30] (83.0), despite using the standard ViT-B backbone and a self-supervised objective.

SSv2 (cross-domain: K400→SSv2). When pretrained on K400 and finetuned on SSv2, TrackMAE reaches 72.8, improving over SMILE [48] (72.1 at 600; 72.4 at 1200) and clearly ahead of VideoMAE [49] (68.5), SIGMA [40] (71.1), MGM [14] (71.1), and MGMAE [23] (68.9). It also exceeds the supervised counterparts reported under the same column (e.g., VideoSwin [34] 69.6, MViTv2-B [33] 70.5, Uniformer-B [30] 71.2). These results indicate better domain transfer to motion-centric SSv2, consistent with the hypothesis that explicit motion supervision complements CLIP-space semantics.

SSv2 (in-domain: SSv2→SSv2). With SSv2 pretraining, TrackMAE attains 72.5, matching the best reported MVM result in the table (SMILE [48] at 72.5) and exceeding other MVM baselines like MGMAE [23] (72.0 at 1600), MGM [14] (71.8 at 1600), CMAE-V [35] (70.5 at 1600), VideoMAE [49] (69.6 at 800), and OmniMAE [17] (69.5). This suggests TrackMAE's motion signal remains beneficial even when the pretraining domain already contains substantial temporal variation.

In summary, TrackMAE achieves state-of-the-art results among masked-video models under comparable settings and is competitive with, or outperforms, strong supervised models. The pattern supports the central claim that explicit motion supervision paired with CLIP-space reconstruction produces video representations that are both semantically strong and motion-aware, yielding robust transfer in-domain and under-domain shift.

8. Additional Ablations

Impact of masking. Table 10a shows the impact of the masking ratio used during pretraining on the finetuning performance. We observe that a masking ratio of 90%, similar to previous methods [23, 49], shows that learning from highly masked spatial and *trajectory* tokens works well in practice.

Impact of separate decoding. We study in Tab. 10b the impact of training with a joint or separate decoders to reconstruct both the spatial and trajectory targets. As shown in the table, separate decoders work best, which may indicate that involving the same decoder for both tasks may lead to information leakage, degrading the signal.

Impact of noisy tracks. As mentioned in the main paper that our method shows robustness against noisy target tracks. The results are shown in Tab. 10c. We observe that our method is robust for both spatial and temporal noise, meaning that even under a noisy tracker, our model would still be able to learn meaningful features.

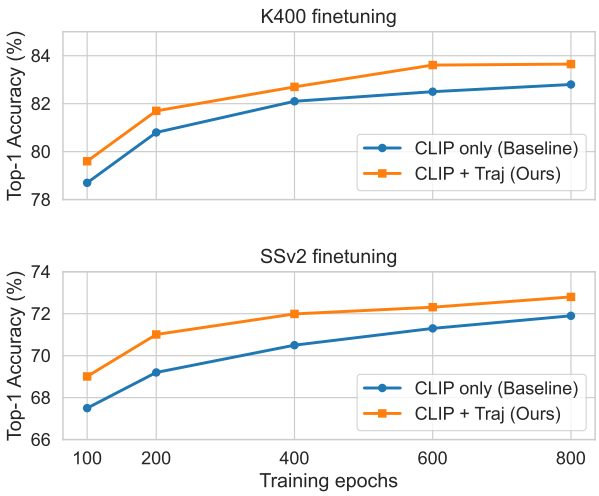


Figure 4. **Training evolution.** We report the Top-1 Accuracy for K400 and SSv2 finetuning at different pretraining epochs. Our model trained with both CLIP and trajectory reconstructions consistently outperforms the CLIP reconstruction only.

Table 9. **Detailed SOTA comparison of masked video modeling methods on Something-Something V2 and Kinetics-400 for full finetuning action recognition.** Our TrackMAE outperforms many supervised approaches and achieves the best performance masked video modeling methods with similar pretraining setups.

				SSv2 Pretraining	K400 Pretraining	
Method	**Backbone**	**Targets**	**Epochs**	**SSv2 Top-1**	**SSv2 Top-1**	**K400 Top-1**
Supervised						
Mformer [37]	Mformer-B	-	-	-	66.7	79.7
VideoSwin [34]	Swin-B	-	-	-	69.6	80.6
TimeSformer [4]	ViT-B	-	-	-	59.5	80.7
MViTv1 [15]	MViTv1-B	-	-	-	67.7	80.2
MViTv2 [33]	MViTv2-B	-	-	-	70.5	82.9
Uniformer-B [30]	Uformer-B	-	-	-	71.2	83.0
Self-supervised						
VideoMAE [49]	ViT-B	Pixel	800	69.6	68.5	80.0
VideoMAE [49]	ViT-B	Pixel	1600	69.6	-	81.5
CMAE-V [35]	ViT-B	Pixel	800	69.7	-	80.2
CMAE-V [35]	ViT-B	Pixel	1600	70.5	-	80.9
OmniMAE [17]	ViT-B	Pixel	800	69.5	69.0	80.8
MGM [14]	ViT-B	Pixel	800	70.6	71.1	80.8
MGM [14]	ViT-B	Pixel	1600	71.8	-	81.7
MGMAE [23]	ViT-B	Pixel	800	71.0	68.9	81.2
MGMAE [23]	ViT-B	Pixel	1600	72.0	-	81.8
MME [45]	ViT-B	HOG	800	70.0	70.5	81.5
MME [45]	ViT-B	HOG	1600	-	-	81.8
SIGMA [40]	ViT-B	DINO	800	71.2	71.1	81.5
SMILE [48]	ViT-B	CLIP	600	**72.5**	72.1	83.1
SMILE [48]	ViT-B	CLIP	1200	-	72.4	83.4
TrackMAE (Ours)	ViT-B	CLIP	600	**72.5**	**72.8**	**83.6**

Table 10. **Additional ablations on our proposed TrackMAE.** The default setting uses pixel reconstruction and motion prediction with a grid size of 14, masking ratio of 90%, $\lambda = 1$, and two separate decoders.

Ratio	$K400_s$	$SSv2_s$
95%	48.1	54.4
90%	**49.4**	**56.2**
80%	49.2	56.0

(a) **Masking ratio.** Masking 90% of the tokens, as in previous methods, works best.

Decoder	$K400_s$	$SSv2_s$
Joint	48.7	55.5
Separate	**49.4**	**56.2**

(b) ***Joint* or *Separate* decoders**. Using separate decoder for motion prediction outperforms using the joint decoder.

Noise	$K400_s$	$SSv2_s$
None	**49.4**	**56.2**
Spatial	48.9	55.7
Temporal	48.8	55.8

(c) **Impact of noisy tracks.** Adding spatial or temporal noise to the tracker trajectories does not affect our method much.

9. Convergence Results

We analyze the convergence behavior of our approach compared to the baseline. We evaluate our CLIP + Trajectory and CLIP-only reconstruction at different pretraining epochs on both K400 and SSv2 finetuning tasks. Figure 4 shows that our model trained with both the spatial and trajectory targets consistently outperforms the model trained with the spatial targets only. We observe that with fewer epochs, our performance is much better than the baseline; however, the gap is narrowed with more pretraining epochs.

10. Discussion on Motion Masking

In this section, we expand the discussion on our motion-aware strategy. In particular, we expand the discussion on different sampling distributions Sec. 10.1, and the different sampling strategies Sec. 10.2 based on motion trajecotories.

Figure 5. **Sampling strategies.** We show how we can use the motion information to create different sampling distribution.

10.1. Sampling Distribution in Masking

As mentioned in the main paper, we use the motion trajectories to create a sampling distribution. We explore two different ways to do this: (1) we accumulate the displacement made by each point of the grid through time with respect to the first frame, (2) we accumulate the displacement made by each point of the grid through time in the next consecutive frame. In other words, the first strategy gives the average motion information over time of *how much* a given point is moving, without taking into account the trajectory. The second strategy gives the average motion information over time of *how much* and *where* a given point is moving. Those 2 sampling strategies are depicted in Fig. 5.

Both sampling strategies have their own merits, *i.e*. the first strategy will put some emphasis towards tokens that are likely to move, without guarantee that they are seen in the following frames. Our intuition is that it forces the model towards learning some motion dynamics of the different moving objects. For the second strategy, our intuition follows the same reasoning, but with visible tokens sampled along the trajectory. In practice, we see both version works on par, with slightly better results for the first strategy, which is used for the results in the main paper. It is also worth mentioning that whatever the strategy used, it is always impacted by the input data. We observe that the K400 dataset is prone to have erratic movements, which may lead to poor motion information. In such a case, our sampling distribution tends to behave more like a uniform sampling, thus falling back to the original random tube masking strategy.

10.2. Sampling Strategy in Masking

Besides different sampling strategy, there are also many different ways to sample visible tokens from it. For computational reasons and for its intuitive soundness, we only evaluated the motion bins strategy described in the main paper. However, we describe below some other strategies.

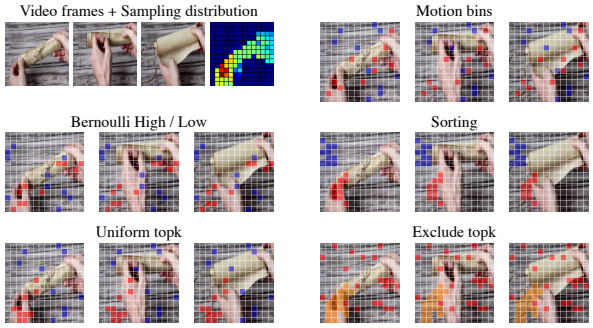


Figure 6. **Masking strategies.** We show how we can use our sampling distribution to create different masking maps. Red squares are high-motion visible tubes, blue squares are low-motion visible tokens. For the "exclude topk" strategy, we also show orange the tokens excluded from being sampled.

Motion bins. We sample visible tokens from high- and low-motion regions of the sampling distribution using 2 uniform bins. Depending on the value of ρ_{motion}, we control the number of tokens sampled coming from each bins. Our intuition is that this strategy gives, on average, a good balance between high- and low-motion visible and masked tokens. This is the strategy used in the main paper.

Bernoulli High/Low. Instead of uniformly sampling from high- and low-motion regions, we can sample using a Bernoulli distribution. For the high-motion tokens, we directly sample from the sampling distribution, and for the low-motion token, we sample from the "1-motion" distribution, excluding tokens already sampled from the high-motion part. Similarly as for the motion bins strategy, we can use the same parameter to control the number of tokens for each high- and low-motion regions. However, we visually observe that this strategy tends to create blob regions, which would usually hurts training.

Sorting. Based on the previous strategy, we can directly sort high- and low-motion tokens and follow their ordering, instead of sampling from the distribution. This strategy creates bigger blobs, which we believe would hurt even more the training.

Uniform topk. To find a better balance between Bernoulli and sorting, we can uniformly sample from the Top-k most moving tokens. However, in order to sample from a smaller set than in the motion bins strategy, we need to specify how many samples are used in the Top-k operation, adding a new hyperparameter to tune.

Exclude topk. In contrast to previous strategies, which use the motion to guide the sampling of visible tokens, we can use the motion information to decide where *we should not* sample visible tokens. From the Top-k most moving tokens, we can chose to explicitly remove them from the sampling

Table 11. **Datasets details.** Splits of datasets used in full finetuning and linear probing.

Dataset	Abbrev.	#Classes	#Train	#Test
Kinetics-400	K400	400	240K	19K
UCF-101	UCF	101	9.5K	3.8K
HMDB-51	HMDB	51	4.8K	2K
Something-Something V2	SSv2	174	169K	24.8K
FineGYM	GYM	99	20.5K	8.5K

distribution, as learning to reconstruct those tokens may be interesting. Then, we can use any strategy to sample the visible tokens. All sampling strategies are shown in Fig. 6

We leave the exploration of such masking strategies, their impact on different pretraining data and downstream tasks for future work.

11. Experimental Details

11.1. Datasets

Kinetics-400 [27] (K400) is a large-scale YouTube-sourced corpus with 400 human action categories and over 306k short clips. It remains a canonical benchmark for learning generalizable video representations.

Something-Something V2 [19] (SSv2) emphasizes object-centric, first-person interactions that differ markedly from K400's Internet footage. It comprises 168,913 training and 24,777 test samples spanning 174 categories, stressing temporal reasoning and commonsense dynamics.

UCF-101 [44] (UCF) collects 9,537 training and 3,783 test clips from YouTube across 101 action classes. Although coarser in granularity and overlapping with K400 categories, it is widely used for transfer evaluation in self-supervised video learning.

HMDB-51 [28] (HMDB) contains 6,766 clips from varied sources (films, archives, web videos) covering 51 classes (at least 100 videos per class). Its heterogeneity challenges models to cope with diverse cinematic and real-world content.

FineGYM [42] (GYM) targets fine-grained action understanding in gymnastics. We use the Gym-99 subset (99 classes) with 20,484 training and 8,521 test samples, focusing on subtle motion differences within highly structured routines.

SEVERE Benchmark [46] SEVERE aggregates eight evaluation settings across SSv2, UCF, FineGYM, and Charades to stress sample efficiency, granularity, and task shift. Table 12 details each subset and metric.

11.2. Training and Evaluation Details

Pretraining Details. We pretrain on K400 [27] and SSv2 [19]. Following VideoMAE [49], we sample clips of 16 frames at 224×224 with a temporal stride of 2 on SSv2 and 4 on K400. We compute space–time tube tokens via a 3D convolution, treating each 2×16×16 cube as a token. For each sampled 16-frame clip, we temporally downsample it with a stride of 2 to extract the trajectories from CoTracker3 [26], matching the total number of trajectory tokens with space-time cubes. In practice, we use the normalized temporal differences of the extracted motion trajectories as the target, rather than absolute values.

All hyperparameters are shown in Table Tab. 13, following [48, 49]. All pretraining runs use 16×NVIDIA A100 GPUs. For downstream tasks, we discard the decoders and use only the pretrained encoder with a task-specific head (*e.g.*, a linear classifier for action recognition).

Linear probing details. We strictly follow the settings in [48] and train the linear head on top of the frozen backbone with the target dataset, as shown in Tab. 14. Experiments are run on 4×V100 GPUs.

Full finetuning details, We strictly follow the settings in [48] and train the backbone + head with the taget dataset, as shown in Tab. 15. Experiments are run on 4×V100 GPUs.

SEVERE benchmark details. Following [40, 48], we evaluate with the official SEVERE codebase [46], strictly reusing the provided training and evaluation configurations to ensure a fair comparison, also shown in Tab. 15. Experiments are run on 4×V100 GPUs.

Table 12. **SEVERE benchmark.** Subsets, protocols, and metrics following [46].

Dataset	Experiment	Setup Group	Task	#Classes	Finetune	Test	Metric
FineGym [42]	Gym99	Full	Action Class.	99	20,484	8,521	Top-1 Acc.
UCF 101 [44]	UCF (10^3)	Sample Efficiency	Action Class.	101	1,000	3,783	Top-1 Acc.
FineGym [42]	Gym (10^3)	Sample Efficiency	Action Class.	99	1,000	8,521	Top-1 Acc.
FineGym [42]	FX-S1	Action Granularity	Action Class.	11	1,882	777	Mean-per-class
FineGym [42]	UB-S1	Action Granularity	Action Class.	15	3,511	1,471	Mean-per-class
UCFRep [57]	UCF-RC	Task Shift	Repetition Counting	–	421	105	Mean Error
Charades [43]	Charades	Task Shift	Multi-label Class.	157	7,985	1,863	mAP

Table 13. **Pretraining configuration.**

Shared		
Optimizer	AdamW	
Base learning rate	1.5×10^{-4}	
Weight decay	0.05	
Momentum (Betas)	β_1=0.9, β_2=0.95	
Batch size	512	
LR schedule	cosine decay	
Warmup epochs	40	
Augmentation	MultiScaleCrop(1, 0.875)	
Dataset-specific	**Epochs**	**FlipAug.**
SSv2	800	no
K400	600	yes

Table 14. **Linear probing configuration.**

config	K400	HMDB	SSv2	GYM
optimizer	AdamW			
base learning rate	1×10^{-3}			
weight decay	0.05			
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$			
layer-wise lr decay	0.75			
batch size	128			
learning rate schedule	cosine decay			
training epochs	30	100	50	100
flip augmentation	*yes*	*yes*	*no*	*yes*

Table 15. **Full finetuning configuration.**

Parameter	**Value**		
Optimizer	AdamW		
Base learning rate	1.0×10^{-3}		
Weight decay	0.05		
Momentum (Betas)	$\beta_1 = 0.9,\ \beta_2 = 0.999$		
Layer-wise LR decay	0.75		
LR schedule	cosine decay		
Warmup epochs	5		
RandAug	(9, 0.5)		
Label smoothing	0.1		
Mixup	0.8		
CutMix	1.0		
Drop path	0.1		
Dataset-specific	**Batchsize**	**Epochs**	**FlipAug.**
SSv2	32	40	no
K400	16	100	yes
SEVERE	16	100	yes

7.2 EPILOGUE

This chapter extends masked modeling from appearance-centric pretraining to motion-centric representation learning. TrackMAE combines trajectory prediction with motion-aware masking. Together, these two components provide a stronger temporal signal than random masking with spatial reconstruction alone.

Empirically, the method improves transfer across diverse downstream settings and is especially effective when temporal reasoning is critical. Together with Chapter 6, this result establishes a coherent progression within the thesis: the same masked modeling principle can be specialized first for object-centric invariances in images, then for motion-centric abstractions in video, by adapting masking and reconstruction design.

The method also has practical constraints. It depends on tracker quality, and tracking adds computation during pretraining. In addition, sparse trajectory supervision does not capture all motion phenomena, especially under severe occlusion or very long temporal dependencies. These limits motivate future work on tighter integration between tracking reliability, masking policy, and reconstruction targets.

Part III

CONCLUSION

8

CONCLUSION

This thesis studied one central question introduced in Chapter 1: how to leverage unlabeled data efficiently. The work addressed this question through two complementary paradigms: semi-supervised learning and self-supervised learning.

In semi-supervised object detection (Part I), the thesis made two contributions. First, Chapter 3 showed that pseudo-label uncertainty should be handled explicitly during student training and that confidence-aware loss weighting improves robustness. Second, Chapter 4 introduced adaptive threshold selection to reduce dependence on costly manual search while improving transfer across datasets.

In self-supervised learning (Part II), the thesis examined masked modeling objectives for images and videos. Chapter 6 demonstrated that object-centric representations can be learned efficiently from Siamese image crops, reducing the need for video-only pretraining. Chapter 7 then extended masked video modeling with explicit motion supervision, improving motion sensitivity through trajectory prediction and motion-aware masking.

Taken together, these contributions support a common conclusion: unlabeled data are most useful when training objectives explicitly encode uncertainty, structure, and invariances that match the target task.

Despite these contributions, several limitations remain.

- **Pseudo-label dependence.** Semi-supervised detection pipelines remain sensitive to teacher quality, especially for rare classes and domain shifts.
- **Offline assumptions.** Adaptive threshold estimation was designed for offline pseudo-labeling and is not directly transferable to fully online settings.
- **Objective mismatch risk.** Masked reconstruction objectives can still favor appearance shortcuts when temporal supervision is weak.
- **Auxiliary module dependence.** Motion-aware pretraining can depend on external components, such as tracker reliability and computational overhead.

Several directions follow naturally from this work.

- **Online adaptive pseudo-labeling.** Extending adaptive thresholding to streaming settings with dynamic score statistics.
- **Class-aware and uncertainty-aware selection.** Improving pseudo-label selection for long-tail categories with calibrated confidence and better class balancing.

- **Unified spatial-temporal objectives.** Designing pretraining losses that jointly optimize appearance semantics and motion consistency without relying on non-semantic reconstruction.
- **Task-adaptive pretraining.** Learning objectives that adapt to downstream requirements, instead of using a fixed pretext across tasks.

As a final conclusion, we show in this thesis that *learning with unlabeled data* is possible and can be done in various forms. While the work in the field of semi-supervised object detection has slowed down recently, the line of work in the self-supervised learning paradigm is still in full swing. We hope that the contributions presented in this thesis can contribute to the scientific community, and that we will continue, one way or another, to learn with unlabeled data in the future.

"Anyone who stops learning is old, whether at twenty or eighty. Anyone who keeps learning stays young."

Henry Ford

BIBLIOGRAPHY

Massih-Reza Amini, Vasilii Feofanov, Loïc Pauletto, Liès Hadjadj, Émilie Devijver, and Yury Maximov. Self-training: A survey. *Neurocomputing*, 616:1–14, 2025.

Adrià Arbués Sangüesa, Adriàn Martín, Javier Fernández, Coloma Ballester, and Gloria Haro. Using player's body-orientation to model pass feasibility in soccer. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW)*, pages 3875–3884, Seattle, WA, USA, 2020.

Maruthavanan S. Archana and M. Kalaiselvi Geetha. An efficient ball and player detection in broadcast tennis video. In *Intelligent Systems Technologies and Applications*, pages 427–436, 2015.

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 *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 15619–15629, Vancouver, Can., 2023.

Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montréal, Can., 2014. Curran Assoc. Inc.

Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Gregoire Mialon, Yuandong Tian, and et al. A cookbook of self-supervised learning. *arXiv*, abs/2304.12210, 2023.

Wele Gedara Chaminda Bandara, Naman Patel, Ali Gholami, Mehdi Nikkhah, Motilal Agrawal, and Vishal M. Patel. AdaMAE: Adaptive masking for efficient spatiotemporal learning with masked autoencoders. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14507–14517, Vancouver, Can., 2023.

Albert Bandura. *Social Learning Theory*. Prentice Hall, Englewood Cliffs, NJ, USA, 1977.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–18, Virtual conference, 2022a.

Zhipeng Bao, Pavel Tokmakov, Allan Jabri, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert. Discovering objects that can move. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 11779–11788, New Orleans, LA, USA, 2022b.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Int. Conf. Mach. Learn. (ICML)*, pages 813–824. ML Res. Press, 2021.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. MixMatch: A holistic approach to semi-supervised learning. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, Canada, 2019. Curran Assoc. Inc.

David Berthelot, Nicholas Carlini, Ekin Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *Int. Conf. Learn. Represent. (ICLR)*, Addis Abada, Ethiopia, 2020.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. New York City, NY, USA, 2006.

John D. Bransford, Ann L. Brown, and Rodney R. Cocking. *How People Learn: Brain, Mind, Experience, and School: Expanded Edition*. National Academies Press, Washington, DC, USA, 2000.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 737–744, 1993.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hessei, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Sutskever Ilya, and Dario Amodei. Language models are few-shot learners. *arXiv*, abs/2005.14165, 2020.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 961–970, Boston, MA, USA, 2015.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 139–156, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 9912–9924, Virtual conference, 2020. Curran Assoc. Inc.

Mathilde Caron, Hugo Touvron, Ishan Misra, Herve Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 9630–9640, Montréal, Can., 2021.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 4724–4733, Honolulu, HI, USA, 2017.

Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. SoftMatch: Addressing the quantity-quality tradeoff in semi-supervised learning. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–21, Kigali, Rwanda, 2023a.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *Int. Conf. Mach. Learn. (ICML)*, pages 1691–1703, New York City, NY, USA, 2020a. ML Res. Press.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Int. Conf. Mach. Learn. (ICML)*, pages 1597–1607, 2020b.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 15745–15753, Nashville, TN, USA, 2021.

Xiaokang Chen, Mingyu Ding, Xiaodi Wang, Ying Xin, Shentong Mo, Yunhao Wang, Shumin Han, Ping Luo, Gang Zeng, and Jingdong Wang. Context autoencoder for self-supervised representation learning. *Int. J. Comput. Vis.*, 132(1):208–223, 2023b.

Dading Chong, Helin Wang, Peilin Zhou, and Qingcheng Zeng. Masked spectrogram prediction for self-supervised audio pre-training. In *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, pages 1–5, Rhodes Island, Greece, 2023.

Anthony Cioppa, Adrien Deliège, Maxime Istasse, Christophe De Vleeschouwer, and Marc Van Droogenbroeck. ARTHuS: Adaptive real-time human segmentation in sports through online distillation. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW), CVsports*, pages 2505–2514, Long Beach, CA, USA, 2019.

Anthony Cioppa, Adrien Deliège, Noor Ul Huda, Rikke Gade, Marc Van Droogenbroeck, and Thomas B. Moeslund. Multimodal and multiview distillation for real-time player detection on a football field. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW), CVsports*, pages 3846–3855, Seattle, WA, USA, 2020.

Anthony Cioppa, Adrien Deliège, Silvio Giancola, Floriane Magera, Olivier Barnich, Bernard Ghanem, and Marc Van Droogenbroeck. Camera calibration and player localization in SoccerNet-v2 and investigation of their representations for action spotting. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW), CVsports*, pages 4532–4541, Nashville, TN, USA, 2021.

Anthony Cioppa, Silvio Giancola, Adrien Deliège, Le Kang, Xin Zhou, Zhiyu Cheng, Bernard Ghanem, and Marc Van Droogenbroeck. SoccerNet-tracking: Multiple object tracking dataset and benchmark in soccer videos. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW), CVsports*, pages 3490–3501, New Orleans, LA, USA, 2022.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3213–3223, Las Vegas, NV, USA, 2016.

Ekin D. Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. AutoAugment: Learning augmentation strategies from data. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 113–123, Long Beach, CA, USA, 2019.

Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. RandAugment: Practical automated data augmentation with a reduced search space. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 18613–18624. Curran Assoc. Inc., 2020.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 886–893, San Diego, CA, USA, 2005.

Ishan Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. TCLR: Temporal contrastive learning for video representation. *Comput. Vis. Image Underst.*, 219:1–9, 2022.

Jan De Houwer, Dermot Barnes-Holmes, and Agnes Moors. What is learning? on the nature and merits of a functional definition of learning. *Psychon. Bull. & Rev.*, 20(4):631–642, 2013.

Adrien Deliège, Anthony Cioppa, Silvio Giancola, Meisam J. Seikavandi, Jacob V. Dueholm, Kamal Nasrollahi, Bernard Ghanem, Thomas B. Moeslund, and Marc Van Droogenbroeck. SoccerNet-v2: A dataset and benchmarks for holistic understanding of broadcast soccer videos. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW)*, pages 4503–4514, Nashville, TN, USA, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 248–255, Miami, FL, USA, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, MN, USA, 2019a. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. Conf. North Am. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, pages 4171–4186, Minneapolis, MN, USA, 2019b. Assoc. Comput. Linguistics.

Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2070–2079, Venice, Italy, 2017.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent. (ICLR)*, Virtual conference, 2021.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *Int. J. Comput. Vis.*, 88(2):303–338, 2010.

Alexandre Eymaël, Renaud Vandeghen, Anthony Cioppa, Silvio Giancola, Bernard Ghanem, and Marc Van Droogenbroeck. Efficient image pre-training with siamese cropped masked autoencoders. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 348–366, 2024.

David Fan, Jue Wang, Shuai Liao, Yi Zhu, Vimal Bhat, Hector Santos-Villalobos, Rohith M V, and Xinyu Li. Motion-guided masking for spatiotemporal representation learning. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 5596–5606, Paris, Fr., 2023.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 6804–6815, Montréal, Can., 2021.

Christoph Feichtenhofer, haoqi fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 35946–35958, New Orleans, LA, USA, 2022. Curran Assoc. Inc.

Zhanzhou Feng and Shiliang Zhang. Evolved part masking for self-supervised learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10386–10395, Vancouver, Can., 2023.

Silvio Giancola, Mohieddine Amine, Tarek Dghaily, and Bernard Ghanem. SoccerNet: A scalable dataset for action spotting in soccer videos. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW)*, pages 1792–179210, Salt Lake City, UT, USA, 2018.

Rohit Girdhar and Deva Ramanan. Attentional pooling for action recognition. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 1–12, Long Beach, CA, USA, 2017. Curran Assoc. Inc.

Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. OmniMAE: Single model masked pretraining on images and videos. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10406–10417, Vancouver, Can., 2023.

Ross Girshick. Fast R-CNN. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 1440–1448, Santiago, Chile, 2015.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 580–587, Columbus, OH, USA, 2014.

Yuan Gong, Cheng-I Lai, Yu-An Chung, and James Glass. SSAST: Self-supervised audio spectrogram transformer. In *AAAI Conf. Artif. Intell.*, pages 10699–10709, Virtual venue, 2022.

Christina Gough. Market size of the sports analytics industry worldwide in 2020 and 2028, 2021.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: Training ImageNet in 1 hour. *arXiv*, abs/1706.02677, 2017a.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 5843–5851, Venice, Italy, 2017b.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent – a new approach to self-supervised learning. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 21271–21284. Curran Assoc. Inc., 2020.

Agrim Gupta, Jiajun Wu, Jia Deng, and Li Fei-Fei. Siamese masked autoencoders. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 1–18, New Orleans, LA, USA, 2023. Curran Assoc. Inc.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 1735–1742, New York City, NY, USA, 2006.

Anaïs Halin, Sébastien Piérard, Renaud Vandeghen, Benoît Gérin, Maxime Zanella, Martin Colot, Jan Held, Anthony Cioppa, Emmanuel Jean, Gianluca Bontempi, Saïd Mahmoudi, Benoît Macq, and Marc Van Droogenbroeck. Physically interpretable probabilistic domain characterization. In *Asian Conf. Comput. Vis. Work. (ACCV Work.)*, pages 17–35, Hanoi, Vietnam., 2025.

Trevor Hastie, robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction.* New York City, NY, USA, second edition, 2009.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 770–778, Las Vegas, NV, USA, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2980–2988, Venice, Italy, 2017.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9726–9735, Seattle, WA, USA, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollar, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 15979–15988, New Orleans, LA, USA, 2022.

Jan Held, Renaud Vandeghen, Abdullah Hamdi, Adrien Deliège, Anthony Cioppa, Silvio Giancola, Andrea Vedaldi, Bernard Ghanem, and Marc Van Droogenbroeck. 3D convex splatting: Radiance field rendering with 3D smooth convexes. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 21360–21369, Nashville, TN, USA, 2025a.

Jan Held, Renaud Vandeghen, Sanghyun Son, Daniel Rebain, Matheus Gadelha, Yi Zhou, Ming C. Lin, Marc Van Droogenbroeck, and Andrea Tagliasacchi. Triangle splatting+: Differentiable rendering with opaque triangles. *arXiv*, abs/2509.25122, 2025b.

Jan Held, Sanghyun Son, Renaud Vandeghen, Daniel Rebain, Matheus Gadelha, Yi Zhou, Anthony Cioppa, Ming C. Lin, Marc Van Droogenbroeck, and Andrea Tagliasacchi. MeshSplatting: Differentiable rendering with opaque meshes. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Denver, CO, USA, 2026a.

Jan Held, Renaud Vandeghen, Adrien Deliège, Daniel Hamdi, Abdullah Rebain, Silvio Giancola, Anthony Cioppa, Andrea Vedaldi, Bernard Ghanem, Andrea Tagliasacchi, and Marc Van Droogenbroeck. Triangle splatting for real-time radiance field rendering. In *Int. Conf. 3D Vis. (3DV)*, pages 1–10, Vancouver, Can., 2026b.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv*, abs/1606.08415, 2016.

Elad Hoffer, Tal Ben-Nun, Itay Hubara, Niv Giladi, Torsten Hoefler, and Daniel Soudry. Augment your batch: Improving generalization through instance repetition. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 8126–8135, Seattle, WA, USA, 2020.

Bingkun Huang, Zhiyu Zhao, Guozhen Zhang, Yu Qiao, and Limin Wang. MGMAE: Motion guided masking for video masked autoencoding. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 13447–13458, Paris, Fr., 2023.

Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 28708–28720, New Orleans, LA, USA, 2022. Curran Assoc. Inc.

Samuel Hurault, Coloma Ballester, and Gloria Haro. Self-supervised small soccer player detection and tracking. In *Int. ACM Work. Multimedia Content Anal. Sports (MMSports)*, pages 9–18, Seattle, WA, USA, 2020.

Sunil Hwang, Jaehong Yoon, Youngwan Lee, and Sung Ju Hwang. EVEREST: Efficient masked video autoencoder by removing redundant spatiotemporal tokens. In *Int. Conf. Mach. Learn. (ICML)*, pages 20889–20907, Vienna, Austria, 2024.

Mordor Intelligence. Sports analytics market – Growth, trends, COVID-19 impact, and forecasts (2022 - 2027), 2022.

Allan Jabri, Andrew Owens, and Alexey A. Efros. Space-time correspondence as a contrastive random walk. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*. Curran Assoc. Inc., 2020.

Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, Canada, 2019. Curran Assoc. Inc.

Debesh Jha, Pia H. Smedsrud, Michael A. Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D. Johansen. Kvasir-SEG: A segmented polyp dataset. In *Int. Conf. Multimedia Retr.*, pages 451–462, 2019.

Hueihan Jhuang, Juergen Gall, Silvia Zuffi, Cordelia Schmid, and Michael J. Black. Towards understanding action recognition. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 3192–3199, Sydney, New South Wales, Aust., 2013.

Yudong Jiang, Kaixu Cui, Leilei Chen, Canjin Wang, and Changliang Xu. SoccerDB: A large-scale database for comprehensive video understanding. In *Int. ACM Work. Multimedia Content Anal. Sports (MMSports)*, pages 1–8, Seattle, WA, USA, 2020. ACM.

Zhouqiang Jiang, Bowen Wang, Tong Xiang, Zhaofeng Niu, Hong Tang, Guangshun Li, and Liangzhi Li. Concatenated masked autoencoders as spatial-temporal learner. *arXiv*, abs/2311.00961, 2023.

Paresh R. Kamble, Avinash G. Keskar, and Kishor M. Bhurchandi. A deep learning ball tracking system in soccer videos. *Opto-Electronics Review*, 27(1):58–69, 2019.

Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker3: Simpler and better point tracking by pseudo-labelling real videos. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 6013–6022, Honolulu, HI, USA, 2025.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *arXiv*, abs/1705.06950, 2017.

JongMok Kim, JooYoung Jang, Seunghyeon Seo, Jisoo Jeong, Jongkeun Na, and Nojun Kwak. MUM: Mix image tiles and UnMix feature tiles for semi-supervised object detection. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14492–14501, New Orleans, LA, USA, 2022.

Alex Krizhevsky. Learning multiple layers of features from tiny images, 2009. Technical report, University of Toronto.

Hilde Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso A. Poggio, and Thomas Serre. HMDB: A large video database for human motion recognition. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2556–2563, Barcelona, Spain, 2011.

DTAI Sports Analytics Lab. Why sports analytics, 2019.

Zihang Lai and Andrea Vedaldi. Tracktention: Leveraging point tracking to attend videos faster and better. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 22809–22819, Nashville, TN, USA, 2025.

Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–10, Toulon, Fr., 2017.

Dong-hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on Challenges in Representation Learning*, pages 1–6, Atlanta, Georgia, USA, 2013.

Gang Li, Xiang Li, Yujie Wang, Yichao Wu, Ding Liang, and Shanshan Zhang. PseCo: Pseudo labeling and consistency training for semi-supervised object detection. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 457–472, 2022a.

Ke Li, Gang Wan, Gong Cheng, Liqiu Meng, and Junwei Han. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS J. Photogramm. Remote Sens.*, 159: 296–307, 2020a.

Kunchang Li, Yali Wang, Gao Peng, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. UniFormer: Unified transformer for efficient spatial-temporal representation learning. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–19, Virtual conference, 2022b.

Kunchang Li, Yali Wang, Yizhuo Li, Yi Wang, Yinan He, Limin Wang, and Yu Qiao. Unmasked teacher: Towards training-efficient video foundation models. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 19891–19903, Paris, Fr., 2023.

Xueting Li, Sifei Liu, Shalini De Mello, Xiaolong Wang, Jan Kautz, and Ming-Hsuan Yang. Joint-task self-supervised learning for temporal correspondence. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 1–11, Vancouver, Can., 2019. Curran Assoc. Inc.

Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, and Boqing Gong. Improving object detection with selective self-supervised self-training. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 589–607, 2020b.

Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. MViTv2: Improved multiscale vision transformers for classification and detection. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 4794–4804, New Orleans, LA, USA, 2022c.

Tsung-Yi. Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 740–755, 2014.

Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2117–2125, Honolulu, HI, USA, 2017a.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. *arXiv*, abs/1708.02002, 2017b.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander Berg. SSD: Single shot multibox detector. *arXiv*, abs/1512.02325, 2016.

Yang Liu, Luiz Hafemann, Michael Jamieson, and Mehrsan Javan. Detecting and matching related objects with one proposal multiple predictions. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW)*, pages 4515–4522, Nashville, TN, USA, 2021a.

Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Pzizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. In *Int. Conf. Learn. Represent. (ICLR)*, 2021b.

Yen-Cheng Liu, Chih-Yao Ma, and Zsolt Kira. Unbiased teacher v2: Semi-supervised object detection for anchor-free and anchor-based detectors. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9809–9818, New Orleans, LA, USA, 2022a.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3192–3201, New Orleans, LA, USA, 2022b.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–16, Toulon, France, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Int. Conf. Learn. Represent. (ICLR)*, New Orleans, LA, USA, 2019.

Cheng-Ze Lu, Xiaojie Jin, Zhicheng Huang, Qibin Hou, Ming-Ming Cheng, and Jiashi Feng. CMAE-V: Contrastive masked autoencoders for video action recognition. *arXiv*, abs/2301.06018, 2023.

Mehrtash Manafifard, Hamid Ebadi, and Hamid Abrishami Moghaddam. A survey on player tracking in soccer videos. *Comput. Vis. Image Underst.*, 159:19–46, 2017.

Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 527–544, 2016.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 69–84, 2016.

Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montréal, Can., 2018. Curran Assoc. Inc.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 1:1–32, 2024.

Luca Pappalardo, Paolo Cintia, Alessio Rossi, Emanuele Massucco, Paolo Ferragina, Dino Pedreschi, and Fosca Giannotti. A public data set of spatio-temporal match events in soccer competitions. *Sci. Data*, 6(1):1–15, 2019.

Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and João F. Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 12493–12506, Virtual conference, 2021. Curran Assoc. Inc.

Hieu Pham, Zihang Dai, QIzhe Xie, Minh-Thang Luong, and Quoc Le. Meta pseudo labels. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 11557–11568, Nashville, TN, USA, 2021.

Jean Piaget. *The Origins of Intelligence in Children.* International Universities Press, New York City, NY, USA, 1952.

Sébastien Piérard, Anthony Cioppa, Anaïs Halin, Renaud Vandeghen, Maxime Zanella, Benoît Macq, Saïd Mahmoudi, and Marc Van Droogenbroeck. Mixture domain adaptation to improve semantic segmentation in real-world surveillance. In *IEEE/CVF Winter Conf. Appl. Comput. Vis. Work. (WACVW)*, pages 22–31, Waikoloa, HI, USA, 2023.

Miran Pobar and Marina Ivasic-Kos. Mask R-CNN and optical flow based method for detection and marking of handball actions. In *Int. Congr. Image Signal Process. Biomed. Eng. Informatics (CISP-BMEI)*, pages 1–6, Beijing, China, 2018.

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*, abs/1704.00675, 2017.

Zhiwu Qing, Shiwei Zhang, Ziyuan Huang, Xiang Wang, Yuehuan Wang, Yiliang Lv, Changxin Gao, and Nong Sang. MAR: Masked autoencoders for efficient action recognition. *IEEE Trans. Multimedia*, 26:218–233, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Int. Conf. Mach. Learn. (ICML)*, pages 8748–8763, Virtual Conf., 2021. ML Res. Press.

Upendra M. Rao and Umesh C. Pati. A novel algorithm for detection of soccer ball and player. In *Int. Conf. Commun. Signal Process. (ICCSP)*, pages 344–348, Melmaruvathur, India, 2015.

Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *arXiv*, abs/1804.02767, 2018.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 779–788, Las Vegas, NV, USA, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39 (6):1137–1149, 2017.

C. Rosenberg, M. Hebert, and H. Schneiderman. Semi-supervised self-training of object detection models. In *IEEE Workshops on Applications of Computer Vision (WACV/MOTION)*, pages 29–36, Breckenridge, CO, USA, 2005.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015.

Jenny R. Saffran, Richard N. Aslin, and Elissa L. Newport. Statistical learning by 8-month-old infants. *Science*, 274(5294):1926–1928, 1996.

Melike Sah and Cem Direkoglu. Evaluation of image representations for player detection in field sports using convolutional neural networks. In *Int. Conf. Theory Appl. Fuzzy Syst. Soft Comput. (ICAFS)*, pages 107–115, 2018.

Mohammadreza Salehi, Michael Dorkenwald, Fida Mohammad Thoker, Efstratios Gavves, Cees G. M. Snoek, and Yuki M. Asano. SIGMA: Sinkhorn-guided masked video modeling. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 293–312, 2024.

Mohammadreza Salehi, Shashanka Venkataramanan, Ioana Simion, Efstratios Gavves, Cees G. M. Snoek, and Yuki M. Asano. MoSiC: Optimal-transport motion trajectory for dense self-supervised learning. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Honolulu, HI, USA, 2025.

Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. Time-contrastive networks: Self-supervised learning from video. In *IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 1134–1141, Brisbane, Aust., 2018.

Long Sha, Jennifier Hobbs, Panna Felsen, Winyu Wei, Patrick Lucey, and Sujoy Ganguly. End-to-end camera calibration for broadcast videos. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 13627–13636, Seattle, WA, USA, 2020.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Camb. Univ. Press, Cambridge, England, United Kingd., 2014.

Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. FineGym: A hierarchical video dataset for fine-grained action understanding. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2613–2622, Seattle, WA, USA, 2020.

Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 510–526, 2016.

Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Han Kurakin, Alexand Zhang, and Colin Raffel. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 596–608. Curran Assoc. Inc., 2020a.

Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv*, abs/2005.04757, 2020b.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv*, abs/1212.0402, 2012.

Gidaris Spyros, Singh Praveer, and Komodakis Nikos. Unsupervised representation learning by predicting image rotations. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–16, Vancouver, Can., 2018.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15 (1):1929–1958, 2014.

Xinyu Sun, Peihao Chen, Liangwei Chen, Changhao Li, Thomas H. Li, Mingkui Tan, and Chuang Gan. Masked motion encoding for self-supervised video representation learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2235–2245, Vancouver, Can., 2023.

Mingxing Tan, Ruoming Pang, and Quoc V. Le. EfficientDet: Scalable and efficient object detection. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10778–10787, Seattle, WA, USA, 2020.

Yuki Tanaka, Shuhei M. Yoshida, and Makoto Terao. Non-iterative optimization of pseudo-labeling thresholds for training object detection models from multiple datasets. In *IEEE Int. Conf. Image Process. (ICIP)*, pages 1676–1680, Bordeaux, France, 2022.

Peng Tang, Chetan Ramaiah, Yan Wang, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. In *IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, pages 2290–2300, Waikoloa, HI, USA, 2021a.

Yihe Tang, Weifeng Chen, Yijun Luo, and Yuting Zhang. Humble teachers teach better students for semi-supervised object detection. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3131–3140, Nashville, TN, USA, 2021b.

Joshua B. Tenenbaum, Charles Kemp, Thomas L. Griffiths, and Noah D. Goodman. How to grow a mind: Statistics, structure, and abstraction. *Science*, 331(6022):1279–1285, 2011.

Fida Mohammad Thoker, Hazel Doughty, Piyush Bagad, and Cees G. M. Snoek. How severe is benchmark-sensitivity in video self-supervised learning? In *Eur. Conf. Comput. Vis. (ECCV)*, pages 632–652, 2022.

Fida Mohammad Thoker, Letian Jiang, Chen Zhao, Piyush Bagad, Hazel Doughty, Bernard Ghanem, and Cees G. M. Snoek. SEVERE++: Evaluating benchmark sensitivity in generalization of video representation learning. *arXiv*, abs/2504.05706, 2025a.

Fida Mohammad Thoker, Letian Jiang, Chen Zhao, and Bernard Ghanem. SMILE: Infusing spatial and motion semantics in masked video learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 8438–8449, Nashville, TN, USA, 2025b.

Graham Thomas, Rikke Gade, Thomas B. Moeslund, Peter Carr, and Adrian Hilton. Computer vision for sports: current applications and research topics. *Comput. Vis. Image Underst.*, 159: 3–18, 2017.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 10078–10093, New Orleans, LA, USA, 2022. Curran Assoc. Inc.

Renaud Vandeghen, Anthony Cioppa, and Marc Van Droogenbroeck. Semi-supervised training to improve player and ball detection in soccer. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW), CVsports*, pages 3480–3489, New Orleans, LA, USA, 2022.

Renaud Vandeghen, Gilles Louppe, and Marc Van Droogenbroeck. Adaptive self-training for object detection. In *IEEE Int. Conf. Comput. Vis. Work. (ICCV Work.)*, pages 914–923, Paris, France, 2023.

Renaud Vandeghen, Fida Mohammad Thoker, Bernard Ghanem, and Marc Van Droogenbroeck. TrackMAE: Video representation learning via track mask and predict. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Denver, CO, USA, 2026.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 6000–6010, Long Beach, CA, USA, 2017.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Int. Conf. Mach. Learn. (ICML)*, pages 1096–1103, Helsinki, Finland, 2008. ACM Press.

Lev S. Vygotsky. *Mind in Society: The Development of Higher Psychological Processes*. Harv. Univ. Press, Cambridge, MA, USA, 1978.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. VideoMAE V2: Scaling video masked autoencoders with dual masking. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14549–14560, Vancouver, Can., 2023a.

Pei Wang, Zhaowei Cai, Hao Yang, Gurumurthy Swaminathan, Nuno Vasconcelos, Bernt Schiele, and Stefano Soatto. Omni-DETR: Omni-supervised object detection with transformers. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9357–9366, New Orleans, LA, USA, 2022.

Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Lu Yuan, and Yu-Gang Jiang. Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 6312–6322, Vancouver, Can., 2023b.

Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 2794–2802, Santiago, Chile, 2015.

Xiaolong Wang, Allan Jabri, and Alexei A. Efros. Learning correspondence from the cycle-consistency of time. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2561–2571, Long Beach, CA, USA, 2019.

Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Bernt Schiele, and Xing Xie. FreeMatch: Self-adaptive thresholding for semi-supervised learning. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–20, Kigali, Rwanda, 2023c.

Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14648–14658, New Orleans, LA, USA, 2022.

Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3733–3742, Salt Lake City, UT, USA, 2018.

Tete Xiao, Xiaolong Wang, Alexei A. Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. In *Int. Conf. Learn. Represent. (ICLR)*, Virtual conference, 2021.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. Unsupervised data augmentation for consistency training. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 6256–6268. Curran Assoc. Inc., 2020a.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves ImageNet classification. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10684–10695, Seattle, WA, USA, 2020b.

Rongchang Xie, Chunyu Wang, Wenjun Zeng, and Yizhou Wang. An empirical study of the collapsing problem in semi-supervised 2D human pose estimation. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 11220–11229, Montréal, Can., 2021.

Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. SimMIM: a simple framework for masked image modeling. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9643–9653, New Orleans, LA, USA, 2022.

Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 3040–3049, Montréal, Can., 2021.

Haosen Yang, Bin Huang, Deng andi Wen, Jiannan Wu, Hongxun Yao, Yi Jiang, Xiatian Zhu, and Zehuan Yuan. MotionMAE: Self-supervised video representation learning with motion-aware masked autoencoders. In *Br. Mach. Vis. Conf. (BMVC)*, pages 1–14, Glasgow, UK, 2024. Br. Mach. Vis. Assoc. (BMVA).

Qize Yang, Xihan Wei, Biao Wang, Xian-Sheng Hua, and Lei Zhang. Interactive self-training with mean teachers for semi-supervised object detection. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 5937–5946, Nashville, TN, USA, 2021.

Yukun Yang, Min Xu, Wanneng Wu, Ruiheng Zhang, and Yu Peng. 3D multiview basketball players detection and localization based on probabilistic occupancy. In *Digit. Image Comput. Tech. Appl.*, pages 1–8, Canberra, ACT, Australia, 2018.

Rui Yao, Guosheng Lin, Shixiong Xia, Jiaqi Zhao, and Yong Zhou. Video object segmentation and tracking. *ACM Trans. Intell. Syst. Technol.*, 11(4):1–47, 2020.

Junqing Yu, Aiping Lei, Zikai Song, Tingting Wang, Hengyou Cai, and Na Feng. Comprehensive dataset of broadcast soccer videos. In *IEEE Conf. Multimedia Inf. Process. Retr. (MIPR)*, pages 418–423, Miami, FL, USA, 2018.

Runtian Zhai, Tianle Cai, Di He, Chen Dan, Kun He, John Hopcroft, and Liwei Wang. Adversarially robust generalization just requires more unlabeled data. *arXiv*, abs/1906.00555, 2019a.

Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4L: Self-supervised semi-supervised learning. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 1476–1485, Seoul, South Korea, 2019b.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. FlexMatch: Boosting semi-supervised learning with curriculum pseudo labeling. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 18408–18419, Virtual conference, 2021. Curran Assoc. Inc.

Fangyuan Zhang, Tianxiang Pan, and Bin Wang. Semi-supervised object detection with adaptive class-rebalancing self-training. In *AAAI Conf. Artif. Intell.*, pages 3252–3261. Association for the Advancement of Artificial Intelligence (AAAI), 2022.

Huaidong Zhang, Xuemiao Xu, Guoqiang Han, and Shengfeng He. Context-aware and scale-insensitive temporal repetition counting. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 667–675, Seattle, WA, USA, 2020.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *Int. J. Comput. Vis.*, 127(3):302–321, 2018a.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *Int. Conf. Learn. Represent. (ICLR)*, pages 1–29, Virtual conference, 2022.

Qixian Zhou, Xiaodan Liang, Ke Gong, and Liang Lin. Adaptive temporal encoding network for video instance-level human parsing. In *ACM Int. Conf. Multimedia (MM)*, pages 1527–1535, Seoul, South Korea, 2018b. ACM.

Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, and Hao Li. Instant-teaching: An end-to-end semi-supervised object detection framework. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 4079–4088, Nashville, TN, USA, 2021.

Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le. Rethinking pre-training and self-training. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 3833–3845. Curran Assoc. Inc., 2020.