ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240812

Bridging Continual Learning of Motion and Self-Supervised Representations

Matteo Tiezzi[a,*,1], Simone Marullo[b,1], Alessandro Betti[c], Michele Casoni[d] and Stefano Melacci[d]

[a]PAVIS, Istituto Italiano di Tecnologia, Genoa, Italy
[b]DINFO, University of Florence, Italy
[c]IMT School for Advanced Studies, Lucca, Italy
[d]DIISM, University of Siena, Italy

Abstract. Efficiently learning unsupervised pixel-wise visual representations is crucial for training agents that can perceive their environment without relying on heavy human supervision or abundant annotated data. Motivated by recent work that promotes motion as a key source of information in representation learning, we propose a novel instance of contrastive criterions over time and space. In our architecture, pixel-wise motion field and representations are extracted by neural models, trained from scratch in an integrated fashion. Learning proceeds online over time, exploiting also a momentum-based moving average to update the feature extractor, without replaying any large buffers of past data. Experiments on real-world videos and on a recently introduced benchmark, with photorealistic streams generated from a 3D environment, confirm that the proposed model can learn to estimate motion and jointly develop representations. Our model nicely encodes the variable appearance of the visual information in space and time, significantly overcoming a recent approach and it also compares favourably with convolutional and Transformer-based networks, offline-pre-trained on large collections of supervised and unsupervised images.

1 Introduction and Related Work

Several studies in the context of perception highlighted that the ability of biological systems to properly identify and segment visual patterns into specific entities (objects, animals, etc.) largely improves in presence of motion [37, 30]. This statement aligns with the *Gestalt Principles of common fate* [44], stating that we perceive visual elements having similar motion as parts of a single stimulus. Recently, motion-based principles are being exploited in computer vision to develop visual skills [32, 25, 20, 4]. As a matter of fact, motion "connects" different views, poses, orientations of the same entity along the temporal dimension. Relating different views, although not relying on motion, is also popular in self-supervised techniques, usually building image-level representations [23]. Self-supervised methods specifically designed to learn pixel-level representations can highly improve the spatial awareness of neural models, as briefly explored by recent research [45], and this is the context in which we believe motion-driven learning can play a major role.

Learning from a single, continuous, stream of non-i.i.d. data represents an attractive challenge for designing agents that either learn from scratch or that progressively adapt to the dynamics of the external stimuli over time [3]. It is widely known that the continual learning setting implies concerns about the plasticity and stability of neural models [31, 13], but it represents the most natural learning setting in the case of data streamed over time. Indeed, continuous visual streams constitute the ideal workbench to devise novel learning algorithms, capable of dealing with dynamically evolving conditions without losing the relevant concepts acquired during the learning process (i.e., without *forgetting*). In this setting, the additional cues brought by temporal information and, in particular, by the motion field can be exploited to learn equivariant representations [4].

Following these intuitions, in this paper we describe CMOSS (Continual MOtion-based Self-Supervised Learning), a *motion-driven* contrastive criterion over time and space. Frames are processed in an online manner with a two-branch neural architecture, continuously-and-jointly learning to extract pixel-wise (a) motion-field and (b) visual features. Noteworthy, motion plays a dual role in representation learning by favouring the development of features that are consistent with the motion field and by driving our novel stochastic spatio-temporal contrastive learning process. The contributions of this paper are the following: (1) we propose a novel model that jointly learns to estimate motion and to represent pixels, processing data from a potentially lifelong video stream and without any supervision; (2) we introduce a novel motion-based SSL contrastive criterion for dense representations, designed with motion-induced sampling; (3) we introduce ad-hoc integrated solutions to deal with continual learning. In the following, we highlight connections to existing literature.

Continual learning (CL). Despite the recent surge in research on neural models capable of learning over time, most of the literature is about supervised CL, with few notable unsupervised exceptions [24, 35]. Multiple approaches have emerged [43, 29], namely context-specific components, parameter or functional regularization, replay, template-based methods. Here we focus on unsupervised long visual experiences unrolling over time, learning in an online manner, without replays and with motion-induced regularization.

Unsupervised learning by motion cues. The idea of exploiting motion cues to support learning has been exploited for designing pretext tasks in the unsupervised learning community [25], aligning the similarity between pairs of feature vectors to the similarity between

* Corresponding Author. Work done while at DIISM, University of Siena, Italy. Email: matteo.tiezzi@iit.it
[1] Equal contribution.

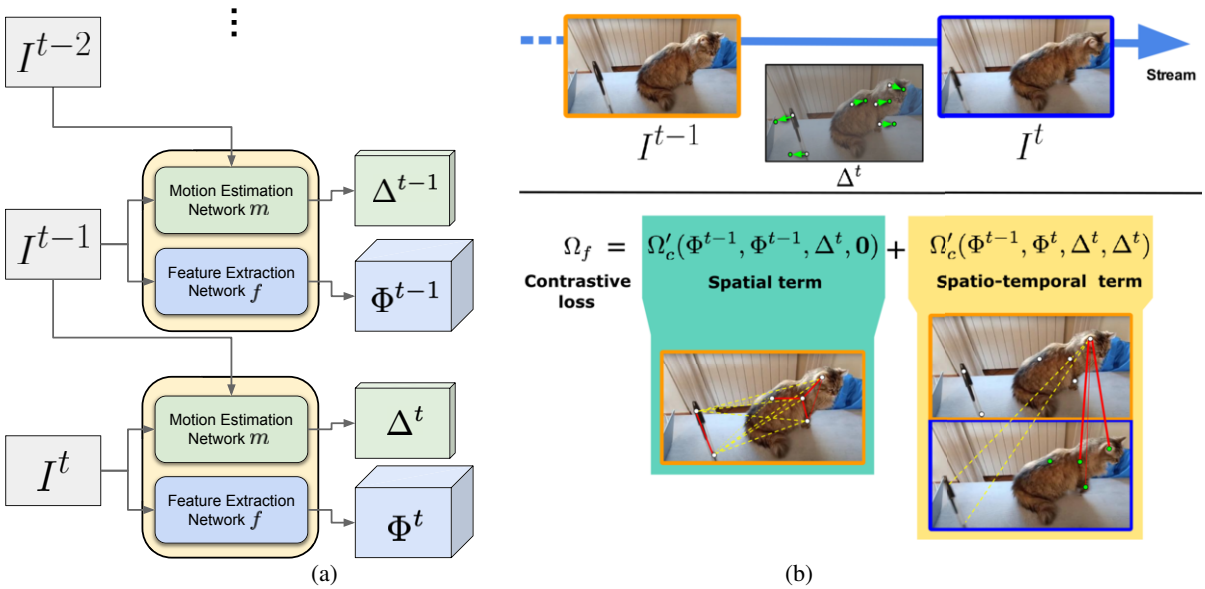

Figure 1. (a) The architecture of CMOSS at inference time. (b) Contrastive criterion. (b–top) Given a pair of frames (I^{t-1}, I^t) in which a cat moves rightward and a pen moves leftward, the motion field Δ^t is computed by network m, and some pixels (white circles) are sampled (sampling involves static pixels as well, not shown for clarity). Displacements for the sampled pixels are represented with green arrows. (b–bottom) Contrastive loss Ω_f consists of two motion-dependent instances of Ω'_c. One is about spatial relationships (left), the other is about spatio-temporal relationships (right). Positive pairs are solid-red (nearby points with similar motion) while negative pairs are dashed-yellow (distant points with dissimilar motion). Thinner links are more uncertain.

corresponding flow vectors or using segments [32] from low-level motion-based grouping to train CNNs. [42] train models in video streams, exploiting a given motion cue to enforce uniform representations over moving entities along an attention trajectory (external component)[41]. Conversely CMOSS is trained by a more sophisticated self-supervised objective that is not limited to an attention trajectory, and, noticeably, our approach also learns to estimate the motion field [26], without any additional inputs. Our learning criterion is driven by the idea of bridging motion prediction and feature extraction while jointly learning such skills over time, that, to our best knowledge, is a novel direction.

Contrastive self-supervised learning (SSL). Many existing approaches rely on memory banks or large batches where to sample positive and negative pairs [9], while a few recent works have replaced negative pairs with asymmetries in the way features are extracted [15, 10], with still limited but improving theoretical understanding [10, 49]. There exist just a few recent works on contrastive SSL for pixel-wise features [46, 45], even if trained offline on large scale data or learning in a latent space at smaller resolution, thus not at a truly pixel-wise level. Differently, CMOSS is designed to learn pixel-wise representations from a single long video stream, working in a continual manner, which is also different from the literature on SSL that typically exploits collections of different videos, such as small clips, that are offline-processed by stochastic optimization [33].

2 Model

We are given a sequence of frames (a video stream $\mathcal{S}$) as our unique source of information. At a generic time step $t > 0$, frame I^t is yielded from $\mathcal{S}$, at the resolution of $w \times h$ pixels. We use the notation I^t_x to indicate the color representation (e.g., RGB) of the pixel of I^t located at the 2-dimensional coordinates $x \in \mathcal{X}$, with $\mathcal{X}$ the set of valid pixel coordinates. Frames are continuously streamed at a constant frame rate, without any temporal limits (t could be potentially ∞) and they smoothly change over time, thus they are *not* independent. We propose a model that, for each pixel of the input frame, computes both $(i.)$ visual features and $(ii.)$ motion estimation. Such information is strongly coupled, since the visual feature extractor and the motion estimator are learned in a joint manner. In particular, motion is the only driving signal for the development of the visual feature extractor, which conquers equivariance properties naturally induced by the motion field. The model continuously learns from the video stream by means of self-supervised learning, without any external supervision signals. A neural network f, named *feature extractor*, with weights and biases collected in θ^t, processes the currently available input frame I^t, returning a set Φ^t of wh pixel-wise representations, also referred to as feature vectors, each of them of length d,

$$\Phi^t = f(I^t, \theta^t). \tag{1}$$

We indicate with Φ^t_x the feature vector of the pixel of I^t whose coordinates are x. Another neural net m, named *motion estimator*, with learnable parameters γ_t, estimates the apparent motion field between I^{t-1} and I^t , i.e., the optical flow. In particular, following common implementations [14, 26], m returns the displacement vectors of the frame pixels,

$$\Delta^t = m(I^{t-1}, I^t, \gamma^t), \tag{2}$$

being Δ^t_x the 2-dimensional displacement vector at x, time t. We introduce the short-hand notation $x(\Delta^t) = x + \Delta^t_x$ to indicate the coordinates of the pixel in frame I^t that corresponds to x of frame I^{t-1}. As a result, frame I^{t-1} can be approximately reconstructed by warping I^t with Δ^t, thus setting the value of I^{t-1}_x to the one of $I^t_{x(\Delta^t)}$, applying an appropriate interpolation procedure for non-integer displacements and handling border issues [6]. The whole pipeline is shown in Fig. 1 (a). Learning proceeds by connecting the extracted features with the predicted motion, processing pairs of consecutive frames in an *online* manner, with a *single-pass* on each pair and without additional memory buffers. Of course, replays or other continual learning strategies could be added to our approach, but it goes beyond the study of this paper, that is about the more challenging but also more natural scenario of plain continual online learning [42]. At

each time step t, parameters θ^{t+1} and γ^{t+1} are obtained by updating θ^t and γ^t, with the goal of minimizing the loss function $\mathcal{L}$,

$$\mathcal{L}(\Phi^{t-1}, \Phi^t, \Delta^t) = \mathcal{L}_f(\Phi^{t-1}, \Phi^t, \Delta^t) + \lambda_m \mathcal{L}_m(\Delta^t). \quad (3)$$

The first term in the right-hand side of Eq. 3 drives the development of the feature extractor, while the second one is about learning how to predict motion (with $\lambda_m > 0$). The notation highlights the presence of the motion signal Δ^t in both the components, as well as the dependence on the time-related index t. This opens to the following sections, that are about $\mathcal{L}_f$, $\mathcal{L}_m$ (Section 2.1), and continuously learning over time (Section 2.3), respectively.

2.1 Motion-driven Learning

The concept of learning features that are consistent with motion has been explored in several works [5, 32]. The recently proposed theory of vision in [4] formalized the notion of "conjugate field with respect to motion", described by constraints that model the relationships between a set of arbitrary pixel-wise properties Ψ and the motion signal. Each constraint is fulfilled when motion is predicted consistently with the selected field over space/time, and vice-versa, i.e., when pixel-wise properties Ψ are computed such that they are equivariant with respect to the motion field Δ. In the discrete case this corresponds to $\Psi_x^{t-1} - \Psi_{x(\Delta^t)}^t = 0$, for all the frame pixels, that is the minimum of the following loss function with an appropriate penalty ρ,

$$\mathcal{L}_{\bowtie}\left(\Psi^{t-1}, \Psi^t, \Delta^t\right) = \frac{1}{wh} \sum_x \rho\left(\Psi_x^{t-1} - \Psi_{x(\Delta^t)}^t\right). \quad (4)$$

We select ρ to be the generalized Charbonnier photometric distance, $\rho(a) = (\|a\|^2 + \epsilon)^\zeta$ [38], with $\epsilon = 0.001$ and $\zeta = 0.5$, as in [26]. Now we are going to exploit this general loss function to relate pixel intensities I, representations Φ and the motion field Δ.

Learning motion. Learning to estimate motion with a neural model can be achieved as in classic optical flow approaches, minimizing $\mathcal{L}_{\bowtie}$ given the original frames I^{t-1} and I^t as properties Ψ in Eq. 4, paired with an additional term $\Omega(\Delta^t)$ that favours spatial regularity of the displacements [17]. Following previous work [18], we define $\mathcal{L}_m$ of Eq. 3 as

$$\mathcal{L}_m(\Delta^t) = \mathcal{L}_{\bowtie}\left(I^{t-1}, I^t, \Delta^t\right) + \beta_m \Omega_m(\Delta^t), \quad (5)$$

where $\Omega_m(\Delta^t) = (wh)^{-1} \sum_{i=1}^{wh} \|\nabla\left(\Delta_{x,1}^t\right)\|^2 + \|\nabla\left(\Delta_{x,2}^t\right)\|^2$, being ∇ the discrete spatial derivative operator, and $\Delta_{x,j}^t$ the j-th component of Δ_x^t, $j \in \{1, 2\}$ (horizontal, vertical). Since I^t and I^{t-1} are given, Ω_m makes the learning problem well-defined [17]. The weighing coefficient $\beta_m > 0$ affects the extent of regularization: smaller β_m's yield more fine-grained estimations, while larger β_m's favour larger blobs with similar motion. We embraced a minimalist strategy for flow estimation, assuming that movements between consecutive frames are generally small or slow. Of course, employing a multi-scale/pyramidal architecture could enhance the estimation of larger displacements, and our approach would remain applicable. Additionally, there are no inherent limitations on the dynamics or frequency with which objects enter or exit the scene, even if introducing the agent to a strongly dynamical scene from the early stages could harness the robustness of motion estimation.

Learning pixel-wise representations. As discussed in [4], directly minimizing Eq. 4 evaluated with learnable Φ is not enough to effectively learn motion and features, since trivial solutions do exist (e.g., spatially uniform features that do not change over time are perfectly consistent with every motion field, even a random one; temporally constant features and null motion are another solution). For this reason, both features and motion must be specifically characterized by introducing additional constraints in the learning process. In particular, there are two terms that contribute to the definition of the proposed $\mathcal{L}_f$ in Eq. 3, that drives learning of the pixel-wise feature extractor, i.e.,

$$\begin{aligned}\mathcal{L}_f(\Phi^{t-1}, \Phi^t, \Delta^t) = \mathcal{L}_{\bowtie}\left(\Phi^{t-1}, \Phi^t, \Delta^t\right) \\ + \beta_f \Omega_f(\Phi^{t-1}, \Phi^t, \Delta^t),\end{aligned} \quad (6)$$

with $\beta_f > 0$. The leftmost term, $\mathcal{L}_{\bowtie}$ favours the development of features that are consistent with the motion signal Δ^t. The rightmost term, Ω_f, is what correctly characterizes the learned features, avoiding trivial solutions. Differently from Eq. 5, a bare spatial regularity penalty would not be appropriate for Ω_f, since Φ^{t-1} and Φ^t are subject to the effects of the learning procedure (differently from I^{t-1} and I^t in Eq. 5, which are given), and that would easily lead to spatio-temporally uniform features. As regularization, we propose to introduce a novel motion-based contrastive learning approach, whose loss function is Ω_f, the rightmost term in Eq. 6. Differently from common contrastive approaches that work at image-level [23, 9], here we tackle the case of pixel-wise features.

2.2 Self-Supervised Learning

For any pixel x, all frame's spatial coordinates are evaluated to identify positive and negative examples. CMOSS leverages the motion field Δ, assuming [44] that nearby pixels moving like x are likely to belong to the same object, thus they should have similar representation. Differently, what moves in a dissimilar manner and is far away from x is likely to belong to a different object, suggesting different representations. Of course, this criterion is not expected to hold strictly (a non-rigid object will have different motion patterns in different parts of its surface, or there could be a static clone of a moving instance, etc.), but this heuristic guides motion-driven pixel-wise representation development [23, 42]. Thus, a pair that is composed of x and one of the nearby pixels with similar motion is a *positive pair*, while a a distant pixel with different motion forms a *negative pair*. Handling the uncertainty of the process is crucial. We indicate with $p_{x,y}(\Delta)$ the positive-pair confidence score, while $n_{x,y}(\Delta)$ denotes the negative-pair confidence score, both in $[0, 1]$. Using cosine similarity $\mathrm{sim}(a, b) := a \cdot b/(\|a\|\|b\|)$, with Euclidean norm $\|\cdot\|$:

$$n_{x,y}(\Delta) = [\mathrm{sim}(\Delta_x, \Delta_y) \leq \tau_n] \frac{\|x - y\|}{\sqrt{h^2 + w^2}}, \quad (7)$$

$$p_{x,y}(\Delta) = [\mathrm{sim}(\Delta_x, \Delta_y) > \tau_p] \left(1 - \frac{\|x - y\|}{\sqrt{h^2 + w^2}}\right) \quad (8)$$

where $[\cdot]$ is 1 if the condition in brackets is true, otherwise it is 0. Thresholds $\tau_p, \tau_n \in [-1, 1]$ are selected to filter out pairs with uncertain similarities, with the condition $\tau_p \geq \tau_n$ which ensures that p and n are non-zero in a mutually-exclusive manner. The rightmost operands of the products in Eq. 8 are the distance between x and y scaled in $[0, 1]$ and 1 minus such a distance, respectively.[2]

Constrastive loss Ω_f. For each point x and a positive pair in which it is involved, say (x, y), the loss function enhances the similarity of the representations of x and y and the dissimilarities of the

[2] Motion direction is not significant for static points (motion vector is smaller than a fixed threshold τ_m): a moving point and a static point are marked as dissimilar with maximum confidence score, independently on their distance, while a pair of static points is neither similar nor dissimilar. See also Fig. 1 (b).

representations in the negative pairs $(x, \cdot)$. Similarity scores are normalized to a probability distribution (softmax with temperature $\tau > 0$) and the loss is evaluated for all the pixel coordinates involved in positive pairs, with terms weighted according to the confidence estimated by p, n. We consider both the case in which the representations of the components of each pair belong to the same frame (spatial links) and when they belong to consecutive frames (spatio-temporal links). Formally, with $s(u, v, \Phi, \tilde{\Phi}, \Gamma) := \tau^{-1}\text{sim}(\Phi_u, \tilde{\Phi}_{v(\Gamma)})$ the τ-scaled motion-aware similarity:

$$\Omega'_c(\Phi, \tilde{\Phi}, \Delta, \Gamma) = \\ -\sum_{x,y \in \mathcal{X}} \frac{p_{x,y}(\Delta)}{Z} \log \frac{e^{s(x,y,\Phi,\tilde{\Phi},\Gamma)}}{e^{s(x,y,\Phi,\tilde{\Phi},\Gamma)} + \sum_{z \in \mathcal{X}} n_{x,z}(\Delta) e^{s(x,z,\Phi,\tilde{\Phi},\Gamma)}}, \quad (9)$$

for representations $\Phi, \tilde{\Phi}$ and motion fields Δ, Γ, with normalization factor $Z = \sum_{u,v \in \mathcal{X}} p_{u,v}(\Delta)$.[3] If representations belong to the same frame, then $\Delta = \mathbf{0}$ (tensor with all-zeros, i.e., null motion). Now we are ready for the comprehensive definition of contrastive term Ω_f in Eq. 6:

$$\Omega_f(\Phi^{t-1}, \Phi^t, \Delta^t) = \Omega'_c(\Phi^{t-1}, \Phi^{t-1}, \Delta^t, \mathbf{0}) \\ + \Omega'_c(\Phi^{t-1}, \Phi^t, \Delta^t, \Delta^t), \quad (10)$$

that is the sum of a spatial contrastive loss applied to the single frame $t-1$ and a spatio-temporal contrastive criterion computed on the current frame pair $(t-1, t)$. See also Fig. 1 (b-bottom).

Sampling. To tackle computational costs arising from the quadratic relationship with pixel count in Ω_f, we propose a selective criterion guided by motion and representations. We ensure that $(i.)$ sampling probability of sampling in moving areas is the same as sampling in static areas, and that $(ii.)$ the probability of sampling in areas where the j-th feature has the strongest activation (absolute value) is the same for all j's. We aim to $(i.)$ bias sampling towards motion areas even in mostly static shots (critical for motion-based contrastive loss), and $(ii.)$ encourage the development of all the d features, fostering rich and compact visual descriptions. We sample $\ell > 1$ pixel coordinates for each t, collecting them into $\mathcal{X}^t$ (see Appendix A.1 in [1]). Then, x, y, z values involved in the sums of Eq. 9 are restricted to $\mathcal{X}^t \subset \mathcal{X}$, making the computation viable also in low-latency settings, that are typical of models learning from streamed data. In Fig. 1 (b) white circles illustrate elements of $\mathcal{X}^t$ (toy example).

Remarks. While the contrastive term (Eq. 10) and $\mathcal{L}_{\bowtie}$ in Eq. 6 share space-time consistency, both serve distinct purposes. The contrastive loss of Eq. 10 $(i.)$ applies solely to sampled coordinates and $(ii.)$ only focuses on feature direction (due to the cosine similarity). $\mathcal{L}_{\bowtie}$ spans the whole frame area, additionally constraining the length of the representations. Minimizing $\mathcal{L}_{\bowtie}$ in Eq. 6 aligns features with predicted motion and vice versa, while the contrastive term lacks differentiability with respect to motion.

2.3 Learning Over Time

Learning involves a single parameter update given frames I^t and I^{t-1} (cached), at each time step t, with the goal of minimizing the total loss (Eq. 3). Such a loss function gains inherent temporal regularization from $\mathcal{L}_{\bowtie}$ terms in Eqs. 5, 6, making it a natural instance of regularization-based approaches to continual learning. Our contrastive loss (Eq. 10) also contributes a spatio-temporal bridge effect on sampled points.

To enhance protection against forgetting, we draw inspiration from models utilizing Exponential Moving Average (EMA) for weight updates [39, 16, 7, 2]. Such models employ an EMA-updated teacher network while ensuring consistency with a continuously updated student network via gradient descent. The EMA-updated network acts as a slowly progressing encoder, with parameters that evolve smoothly [16], and it can be employed both to stabilize the learning of the student network and to better preserve information from the past. In our method, we extract (I^{t-1}) features with network f using weights θ^t_{GRA}, and $I^t)$ features with an EMA-updated network with the same architecture and weights θ^t_{EMA} (illustrated in Appendix A.2 in [1]). The networks are naturally connected through loss function $\mathcal{L}$ (Eq. 3), that enforces motion-driven coherence. While the GRA network is updated by the gradient of $\mathcal{L}$ and learning rate $\alpha_f > 0$, the other network is updated by EMA with coefficient $\xi \in [0, 1)$,

$$\theta^{t+1}_{GRA} = \theta^t_{GRA} - \alpha_f \nabla_\theta \mathcal{L}(\Phi^{t-1}, \hat{\Phi}^t, \Delta^t), \quad (11)$$

$$\theta^{t+1}_{EMA} = \xi \theta^t_{EMA} + (1 - \xi)\theta^{t+1}_{GRA}, \quad (12)$$

where $\nabla_\theta \mathcal{L}$ is the gradient with respect to the second argument of f and $\hat{\Phi}^t$ indicates that Φ^t is treated here as a constant value (it is the output of the EMA net). We recall that, at each t, the proposed model outputs features computed by the EMA net. This choice is motivated by the fact that the EMA net is characterized by a smoother developmental process, which we found to be appropriate in the continual unsupervised learning scenario, with the purpose of reducing issues related to stability and forgetting. Differently, specific CL methods are not needed [26] for the motion estimator m. The learning procedure is further illustrated in Appendix A.2 in the supplementary materials [1]. We empirically found the EMA net to be helpful in our goal to bridge motion and feature learning over time. Other CL techniques [29] can be used as well, but it goes beyond what we propose.

3 Experiments

We implemented CMOSS using PyTorch,[4] running experiments on a Linux machine–NVIDIA GeForce RTX 3090 GPU (24 GB). We investigate the capability of CMOSS to develop pixel-wise representations and motion field in a continual online self-supervised manner, following the recent experimental framework of [42], devised to evaluate pixel-wise features in continual learning.

Data. The first benchmark consists of three 256×256 streams, obtained by rendering three different photo-realistic scenes from the 3D Virtual Environment SAILenv [27, 28] (samples from the streams are depicted in the first row of Fig. 7-right). In each of them, some target objects, one at a time, perform different complete routes (hereinafter referred to as *lap*) around their starting locations. Objects perform complex movements consisting in rotations/scale/pose variations with respect to a fixed camera. EMPTYSPACE is about a room with uniform background and four moving objects (i.e., chair, laptop, pillow, ewer). It includes a grayscale (-BW suffix) and an RGB (-RGB) stream. SOLID consists of three white 3D shapes (cube, cylinder, sphere) moving in a gray-scale environment. LIVINGROOM (-BW/-RGB) is a complex scene with the same four objects of EMPTYSPACE that move around. It features a heterogeneous background composed of non-target objects (i.e., couch, tables, staircase, door,

[3] Notice that, due to the mutual exclusivity of p and n, the term $n_{x,z}$ in Eq. 9 also acts as a mask that excludes all the points z that are not considered dissimilar to x.

[4] https://github.com/sailab-code/cmoss

floor) and static copies of the EMPTYSPACE objects. The second benchmark includes two real-world videos, RAT (256×128) and HORSE (256×192), proposed by [21], where the objects of interest are rat and horse+jockey, respectively. These videos[5] are used to demonstrate CMOSS's ability to generalize to real-world scenarios, with non-fixed camera.

Setup. We compared CMOSS against the recently proposed model of [42] in pixel-wise classification, following the experimental conditions defined in the considered benchmark: unsupervised learning for 30 laps per object, where during the last 5 laps 3 supervised representations (templates) per-object are saved, one every 100 frames. Such external cues are used for a distance-based open-set class incremental evaluation procedure, and they do not affect representation learning. In the case of CMOSS, we used the first 5 laps to only start training the neural motion estimator.[6] Performance is measured in a last additional lap per object, computing the F1 score (averaged over the available object categories and the background class) over all the pixels of the frames. Following the evaluation protocol of the selected benchmark, the optimal values for the hyperparameters are determined by maximizing the F1 along the trajectory (1 pixel per frame) of a (given) human-like attention model. We report in Appendix A.4 (see [1]) the parameter grid we explored and the optimal parameter values we found.

Compared models. We implemented CMOSS using UNet-like [36] networks for both the functions m and f. In particular, we took the so-called RESUNET from [42], reducing the number of convolutional filters in every layer by a factor of 4. We compared CMOSS to the learning approach by [42], considering the two models presented therein, i.e., RESUNET and FCN-ND (6 convolutional layers, no downsampling) —see Appendix A.3 in [1] for further details. For reference, we compared also to state-of-the-art models pre-trained in semantic segmentation tasks and in self-supervised learning, without any attempts to overcome results from such large-scale offline-trained models. We selected the ResNet101-based DEEPLABV3 [8] and a Dense Prediction Transformer - DPT [34], evaluating both the upsampled representations from the backbones (-B suffix, trained on ImageNet-1M) or from the classification heads (-C suffix, task-specialized higher-level features trained on COCO [22] and ADE20k [50]), and the features yielded by the pre-trained backbones (ResNet-50) of recent self-supervised methods, i.e. MOCO V1-3 [16, 11, 12], PIXPRO [45]. For the sake of completeness, we also report the performances achieved by our CMOSS approach when the stream is processed in an offline manner (i.e. the frames are randomly shuffled). Additionally, we consider the baseline RAW IMAGE, i.e., the original pixel representations (brightness) as features.

Main result. Table 1 reports the results of our comparison. Focusing on the related continual self-supervised competitors (bottom part of the table), the proposed CMOSS significantly overcomes them in all the video streams, with the exception of EMPTYSPACE-BW, where it is on par with the best competitor. We remark that CMOSS learns to estimate motion from scratch, while the competitors are using a pre-computed motion signal that is flawlessly produced by the rendering engine (or by pretrained optical flow SOTA model in the case of real-world videos [40]). Notice that CMOSS also uses a reduced number of output features with respect to the competitors (32 vs. up-to-128), and is characterized by a number of learnable parameters that is way smaller than the best (on average) competitor (1.1M vs. 17.8M, third column of Tab. 1), confirming its capability of developing informed but more compact representations. Our method achieves competitive performance even when compared to offline-pre-trained large architectures, with several millions of parameters (upper part of the table—we recall that our goal is not to overcome these large-scale massively offline-trained models, since we learn from scratch in an online manner). Interestingly, CMOSS beats all these competitors in EMPTYSPACE-RGB and in general demonstrates superior performance compared to several competitors (e.g. PIXPRO, DEEPLAB-C). These results confirm its capability of adapting to the properties of the considered learning environment. Moreover, they are not far from the offline-trained CMOSS, where the training data (pairs of frames) were shuffled. In Fig. 2 (left) we report a qualitative showcase of the predictions on two frames sampled from EMPTYSPACE-RGB, HORSE (similar conclusions hold for the other streams). The features produced by CMOSS allow the classification procedure (not involved in the feature learning process) to almost completely disentangle all the different objects and their parts. Conversely, in Appendix A.8 (see [1]) we show that the selected competitor is not able to correctly classify the chair legs/thinner segments or the surface of the pillow, that get partially confused with the ewer. We notice that CMOSS-based predictions tend to be slightly thicker with respect to the ground-truth object borders. This is mostly due to the quality of the estimated motion field, that is not always perfect (while competitors uses a flawless/given motion field). In fact, motion guides the learning process, having an impact also in the sampling procedure of our stochastic contrastive criterion. Fig. 2 (right) reports examples of motion estimations obtained for two streams (RGB and BW). Noticeably, the estimated motion fields (second column) are very similar to the ground truth motion yielded by the 3D environment (third column), although a bit thicker, coherently with our previous comment. It should be noted that optical flow methods inherently find very challenging to extract motion on nearly texture-free surfaces, as the ones of the cylinder in the SOLID dataset. Conversely, the 3D engine possesses a complete understanding of the actual 3D motion field. Moreover, we did not leverage any specific trick to improve the flow prediction (e.g., spatial pyramids, occlusion handling, etc. [48]), for simplicity, as in [26].

Ablation studies. We ablated key components of CMOSS to assess their impact on results (findings in Fig. 3 and more results in Appendix A.7 – see [1]). The motion-feature coherence term $\mathcal{L}_{\bowtie}$ in the loss function (Eq. 6) improved all streams (Fig. 3-a), particularly in EMPTYSPACE. It also improves stability among the different runs (reduced std). Adopting a specific contrastive term sampling strategy (see Sec. 2.2) (Fig. 3-b)), guided by both Motion and Features (M.F.), influenced the metrics notably. Motion-only biased sampling (Mot.) can still outperform uniform sampling strategy (Plain), by focusing on small moving areas. Omitting the stabilizing EMA network (Fig. 3-c) systematically lowered performance, aligned with literature in contrastive and continual learning. The learning process is however still effective, thanks to regularization effects by the spatio-temporal loss terms, albeit less stable. We assessed the effect of sampled locations ℓ (Fig. 3-d): $\ell = 100$ proved effective while very different values reduced F1, with a milder impact than other discussed hyperparameters. This outcome stems from stream properties, where more dense sampling offers limited gains due to large static or uniform areas.

Online learning and stability. Given the considered open-world task-free scenario, where a few supervisions are provided in a short session after a longer unsupervised training stage, measuring the

[5] https://www.kaggle.com/datasets/gvclsu/long-videos

[6] Considering only the last term in Eq. 3, i.e., $\lambda_m \mathcal{L}_m(\Delta^t)$. We performed some preliminary experiments which showed, as expected, that activating the constrastive criterion with a randomly-initialized motion estimator slightly hinders the final outcome of the learning process.

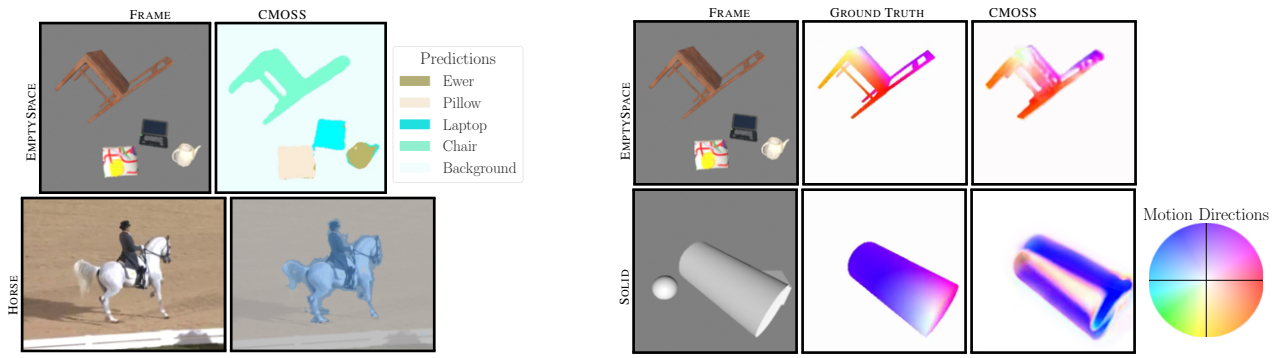

Figure 2. Left: frames from EMPTYSPACE-RGB, HORSE-RGB. The features extracted by CMOSS demonstrate effective classification capabilities, as depicted in the second column. Right: frames from EMPTYSPACE-RGB, SOLID-BW (first column), the ground truth dense motion fields from the 3D environment (second column) and motion estimated by CMOSS (third column).

Table 1. F1 scores (10 runs, mean ± std), over seven different video streams (EMPTYSPACE, SOLID, LIVINGROOM, RAT, HORSE) with some -BW variants. Bottom: the main competitors of the proposed model, including the raw-image (degenerate) baseline. The top part of the table collects reference results obtained with large offline-pretrained models, publicly available, and an offline-trained version of CMOSS.

		# Params	EMPTYSPACE BW	EMPTYSPACE RGB	SOLID BW	LIVINGROOM BW	LIVINGROOM RGB	RAT RGB	HORSE RGB
OFFLINE PRE-TRAINED	DPT-C [34]	121M	0.66	0.67	0.64	0.35	0.39	0.59	0.83
	DPT-B [34]	120M	0.71	0.69	0.68	0.39	0.39	0.58	0.87
	DEEPLAB-C [8]	58.6M	0.49	0.61	0.57	0.31	0.34	0.56	0.86
	DEEPLAB-B [8]	42.5M	0.70	0.65	0.66	0.34	0.44	0.57	0.81
	MOCO V1 [16]	8.5M	0.73	0.73	0.74	0.33	0.35	0.70	0.66
	MOCO V2 [11]	8.5M	0.75	0.76	0.74	0.41	0.43	0.59	0.79
	MOCO V3 [12]	8.5M	0.76	0.76	0.76	0.41	0.44	0.58	0.64
	PIXPRO [45]	8.5M	0.31	0.46	0.41	0.30	0.22	0.59	0.70
	CMOSS (Offline)	1.1M	0.64	0.82	0.65	0.43	0.40	0.69	0.81
CONTINUAL	RAW IMAGE	-	0.50	0.45	0.18	0.10	0.23	0.52	0.66
	HOURGLASS [42]	17.8M	0.55±0.03	0.71±0.03	0.50±0.01	0.31±0.04	0.25±0.07	0.59±0.08	0.71±0.07
	FCN-ND[42]	0.1M	**0.60**±0.05	0.51±0.07	0.48±0.03	0.24±0.01	0.28±0.03	0.42±0.05	0.50±0.08
	CMOSS	1.1M	0.60±0.06	**0.78**±0.06	**0.62**±0.02	**0.33**±0.03	**0.34**±0.05	**0.61**±0.05	**0.75**±0.06

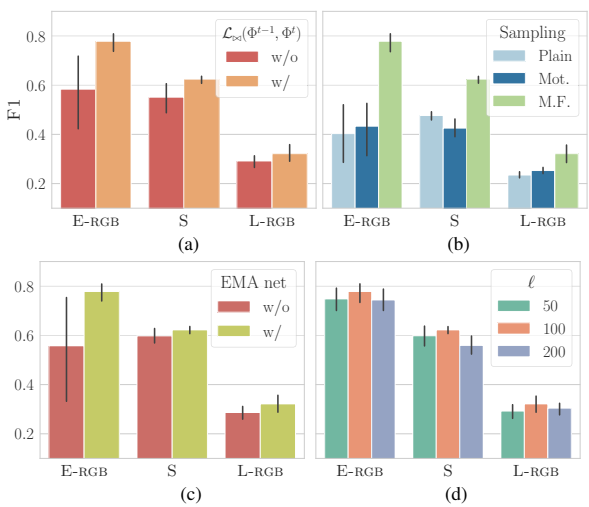

Figure 3. Ablation (F1 score) of the model components on selected streams (E, S, L stands for EMPTYSPACE, SOLID, LIVINGROOM). (a) features-motion consistency term $\mathcal{L}_{\bowtie}$; (b) sampling strategy; (c) EMA network; (d) number of sampled points ℓ. See the main text for details.

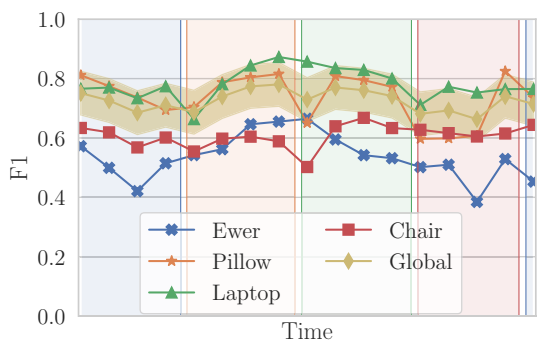

Figure 4. Evolution of the learning process (EMPTYSPACE) right after the time in which results of Tab. 1 were measured and until each object has performed an additional lap (avg of 10 runs). Lines are the F1 of object-specific predictors (ewer, pillow, laptop, chair) and the overall F1 (Global, with std reported as a semitransparent overlay). Vertical colored bands indicate when each object is moving.

usual forward/backward transfer would not be meaningful. However, to better asses the model performances over time, we provide in Fig. 4 a detailed analysis of the selected EMPTYSPACE-RGB model. Following prior experiments, we extended learning with an extra lap, tracking F1 evolution. Colored vertical bands depict time intervals of object motion. Given the influence of moving entities on the sampling process, representation capability subtly shifts over time, leading to slight F1 oscillations. Nonetheless, the variability band (overlaid in Fig. 6 (b)) around the averaged F1 (Global) is relatively small, comparable to the std range in Tab. 1. Notably, the ewer class's object-specific predictor exhibits more prominent variations (max-min), as expected due to its smaller size, where border errors carry greater impact.

4 Conclusions and Future Work

We introduced CMOSS, a method to jointly learns to extract motion and pixel-wise representations from scratch, processing a single stream of data in an online, continual, manner. Motion is the key element of a novel contrastive criterion that, when exploited in continual replay-free online learning, successfully deals with the variable appearance of visual information in space and time. Experimental results on recent benchmarks and on real-world videos show that CMOSS significantly overcomes its main competitor [42], and it also performs similarly to CNNs and Transformer-based networks (offline-pretrained on large datasets), at a fraction of the parameters. CMOSS (as well as its main competitor) is designed to learn in environments where background areas are characterized by motion patterns that are different from the ones of objects to which semantics are expected to be attached. We showed that the relatively slow camera motion in the real-world videos can be easily handled by using a larger τ_m or setting it to the average length of motion vectors in the current frame, as we did in our implementation. However, strongly moving cameras (e.g., egomotion) might lead to less discriminative features. Finally, more advanced continual learning strategies might be required to avoid forgetting issues in significantly longer streams or with many more object categories. Such limitations will be the subject of future studies.

Acknowledgements

This work was supported by the Italian Ministry of Research, under the complementary actions to the NRRP "Fit4MedRob-Fit for Medical Robotics" Grant (# PNC0000007).

References

[1] Bridging Continual Learning of Motion and Self-Supervised Representations – Supplementary material. https://github.com/sailab-code/cmoss/blob/main/cmoss_suppl.pdf.

[2] E. Arani, F. Sarfraz, and B. Zonooz. Learning fast, learning slow: A general continual learning method based on complementary learning system. In *ICLR*, 2022.

[3] A. Betti, M. Gori, S. Melacci, M. Pelillo, and F. Roli. Can machines learn to see without visual databases? In *NeurIPS Workshop on Data Centric AI - arXiv preprint arXiv:2110.05973*, 2021.

[4] A. Betti, M. Gori, and S. Melacci. *Deep Learning to See-Towards New Foundations of Computer Vision*. Springer, 2022.

[5] C. Bregler. Learning and recognizing human dynamics in video sequences. In *Proceedings of IEEE Computer Society CVPR*, pages 568–574, 1997.

[6] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision*, pages 25–36. Springer, 2004.

[7] Z. Cai, A. Ravichandran, S. Maji, C. Fowlkes, Z. Tu, and S. Soatto. Exponential moving average normalization for self-supervised and semi-supervised learning. In *Proceedings of the IEEE/CVF CVPR*, pages 194–203, 2021.

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 40(4):834–848, 2017.

[9] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[10] X. Chen and K. He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF CVPR*, pages 15750–15758, 2021.

[11] X. Chen, H. Fan, R. Girshick, and K. He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[12] X. Chen*, S. Xie*, and K. He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021.

[13] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE TPAMI*, 44(7):3366–3385, 2021.

[14] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2758–2766, 2015.

[15] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in NeurIPS*, 33:21271–21284, 2020.

[16] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF CVPR*, pages 9729–9738, 2020.

[17] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[18] R. Jonschkowski, A. Stone, J. T. Barron, A. Gordon, K. Konolige, and A. Angelova. What matters in unsupervised optical flow. In *European Conference on Computer Vision*, pages 557–572. Springer, 2020.

[19] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[20] T. Kipf, G. F. Elsayed, A. Mahendran, A. Stone, S. Sabour, G. Heigold, R. Jonschkowski, A. Dosovitskiy, and K. Greff. Conditional Object-Centric Learning from Video. In *ICLR*, 2022.

[21] Y. Liang, X. Li, N. Jafari, and J. Chen. Video object segmentation with adaptive feature bank and uncertain-region refinement. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in NeurIPS*, volume 33, pages 3430–3441. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/234833147b97bb6aed53a8f4f1c7a7d8-Paper.pdf.

[22] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014.

[23] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang. Self-supervised learning: Generative or contrastive. *IEEE TKDE*, 35 (1):857–876, 2021.

[24] D. Madaan, J. Yoon, Y. Li, Y. Liu, and S. J. Hwang. Representational continuity for unsupervised continual learning. In *ICLR*, 2022. URL https://openreview.net/forum?id=9Hrka5PA7LW.

[25] A. Mahendran, J. Thewlis, and A. Vedaldi. Cross pixel optical-flow similarity for self-supervised learning. In *Computer Vision–ACCV 2018, Revised Selected Papers, Part V 14*, pages 99–116. Springer, 2019.

[26] S. Marullo, M. Tiezzi, A. Betti, L. Faggi, E. Meloni, and S. Melacci. Continual unsupervised learning for optical flow estimation with deep networks. In *Conference on Lifelong Learning Agents*, pages 183–200. PMLR, 2022.

[27] E. Meloni, L. Pasqualini, M. Tiezzi, M. Gori, and S. Melacci. Sailenv: Learning in virtual visual environments made simple. In *ICPR*, pages 8906–8913, 2020.

[28] E. Meloni, A. Betti, L. Faggi, S. Marullo, M. Tiezzi, and S. Melacci. Evaluating continual learning algorithms by generating 3d virtual environments. In *International Workshop on Continual Semi-Supervised Learning*, pages 62–74. Springer, 2021.

[29] M. Mundt, Y. Hong, I. Pliushch, and V. Ramesh. A wholistic view of continual learning with deep neural networks: Forgotten lessons and the bridge to active and open world learning. *Neural Networks*, 160:306–336, 2023.

[30] Y. Ostrovsky, E. Meyers, S. Ganesh, U. Mathur, and P. Sinha. Visual parsing after recovery from blindness. *Psychological Science*, 20(12): 1484–1491, 2009.

[31] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter. Continual lifelong learning with neural networks: A review. *Neural networks*, 113: 54–71, 2019.

[32] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan. Learning features by watching objects move. In *Proceedings of the IEEE CVPR*, pages 2701–2710, 2017.

[33] R. Qian, T. Meng, B. Gong, M.-H. Yang, H. Wang, S. Belongie, and Y. Cui. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF CVPR*, pages 6964–6974, 2021.

[34] R. Ranftl, A. Bochkovskiy, and V. Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021.

[35] D. Rao, F. Visin, A. Rusu, R. Pascanu, Y. W. Teh, and R. Hadsell. Continual unsupervised representation learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in NeurIPS*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/861578d797aeb0634f77aff3f488cca2-Paper.pdf.

[36] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015.

[37] E. S. Spelke. Principles of object perception. *Cognitive science*, 14(1): 29–56, 1990.

[38] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *International CVPR*, pages 2432–2439, 2010. doi: 10.1109/CVPR.2010.5539939.

[39] A. Tarvainen and H. Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in NeurIPS*, volume 30. Curran Associates, Inc., 2017.

[40] Z. Teed and J. Deng. Raft: Recurrent all-pairs field transforms for optical flow, 2020.

[41] M. Tiezzi, S. Melacci, A. Betti, M. Maggini, and M. Gori. Focus of attention improves information transfer in visual features. *Advances in NeurIPS*, 33:22194–22204, 2020.

[42] M. Tiezzi, S. Marullo, L. Faggi, E. Meloni, A. Betti, and S. Melacci. Stochastic coherence over attention trajectory for continuous learning in video streams. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 3480–3486, 7 2022.

[43] G. M. van de Ven, T. Tuytelaars, and A. S. Tolias. Three types of incremental learning. *Nature Machine Intelligence*, 4(12):1185–1197, 2022. doi: 10.1038/s42256-022-00568-3. URL https://doi.org/10.1038/s42256-022-00568-3.

[44] M. Wertheimer. Laws of organization in perceptual forms. *A source book of Gestalt psychology*, pages 71–88, 1938.

[45] Z. Xie, Y. Lin, Z. Zhang, Y. Cao, S. Lin, and H. Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF CVPR*, pages 16684–16693, 2021.

[46] Y. Xiong, M. Ren, W. Zeng, and R. Waabi. Self-supervised representation learning from flow equivariance. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10171–10180, 2021.

[47] D. Zanca, M. Gori, S. Melacci, and A. Rufa. Gravitational models explain shifts on human visual attention. *Scientific Reports*, 10(1):1–9, 2020.

[48] M. Zhai, X. Xiang, N. Lv, and X. Kong. Optical flow and scene flow estimation: A survey. *Pattern Recognition*, 114:107861, 2021.

[49] C. Zhang, K. Zhang, C. Zhang, T. X. Pham, C. D. Yoo, and I. S. Kweon. How does simsiam avoid collapse without negative samples? a unified understanding with self-supervised contrastive learning. In *ICLR*, 2022.

[50] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *IJCV*, 127(3):302–321, 2019.