MaCLR: Motion-aware Contrastive Learning of Representations for Videos

Fanyi Xiao[*], Joseph Tighe, and Davide Modolo

AWS AI Labs
fyxiao@ucdavis.edu, {tighej,dmodolo}@amazon.com

Abstract. We present MaCLR, a novel method to explicitly perform cross-modal self-supervised video representations learning from visual and motion modalities. Compared to previous video representation learning methods that mostly focus on learning motion cues implicitly from RGB inputs, MaCLR enriches standard contrastive learning objectives for RGB video clips with a cross-modal learning objective between a Motion pathway and a Visual pathway. We show that the representation learned with our MaCLR method focuses more on foreground motion regions and thus generalizes better to downstream tasks. To demonstrate this, we evaluate MaCLR on five datasets for both action recognition and action detection, and demonstrate state-of-the-art self-supervised performance on all datasets. Furthermore, we show that MaCLR representation can be as effective as representations learned with full supervision on UCF101 and HMDB51 action recognition, and even outperform the supervised representation for action recognition on VidSitu and SSv2, and action detection on AVA.

1 Introduction

Supervised learning has enjoyed great successes in many computer vision tasks in the past decade. One of the most important fuel in this successful journey is the availability of large amount of high-quality labeled data. Notably, the ImageNet [17] dataset for image classification was the spark that ignited the deep learning revolution in vision. In the video domain, the Kinetics dataset [41] has long been regarded as the "ImageNet for videos" and has enabled the "pretrain-then-finetune" paradigm for many video tasks. Interestingly, though years old, ImageNet and Kinetics are still the to-go datasets for pretraining that are publicly available. This shows how much effort is needed to create these large-scale labeled datasets.

To mitigate the reliance on large-scale labeled datasets, *self-supervised learning* came with the promise to learn useful representations from large amount of *unlabeled* data. Following the recent success in NLP (e.g., BERT, GPT-3 [18,7]), some works have attempted to find its counterpart in vision. Among them, pioneering research has been conducted in the image domain to produce successful

[*] *Work done while at Amazon, now at Meta AI*

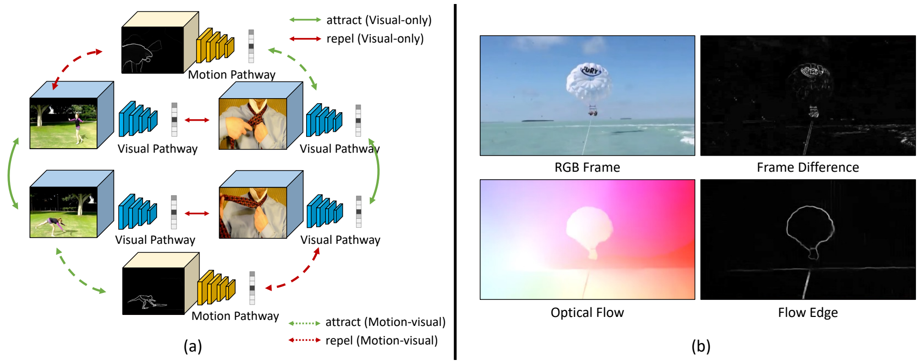

Fig. 1: **(a) Cross-modal motion-visual learning**. We propose Motion-aware Contrastive Learning of Representations (MaCLR) as an explicit method to learn motion-aware video representations without labels – Our visual pathway features are pushed to learn a representation aligns with our motion pathway and in doing so learn features that more robustly capture motion in the video. **(b) Motion inputs**. Given the RGB input (top-left), we compare three options for motion inputs. Best viewed on screen.

methods like MoCo [36] and SimCLR [12]. Compared to images, large-scale video datasets induce even higher annotation costs, making it even more important to develop effective self-supervised methods to learn generalizable representations for videos. Some recent video works attempted to learn such representations by training their models to solve pretext tasks, like predicting the correct temporal order of clips [49,28,6,78,9], predict future frames [19] and predict whether a video is played at its intrinsic speed [4]. Though successful to a certain extent, these methods do not explicitly make use of motion information derived from the temporal sequence, which has been shown to be important for supervised action recognition tasks [63,26,74].

In this paper, we propose MaCLR, a novel self-supervised video representation learning method that *explicitly* models motion cues during training. MaCLR (Motion-aware Contrastive Learning of Representations) consists of two pathways: Visual and Motion. It uses both pathways during self-supervised pretraining, but only transfers the Visual to downstream tasks. When trained alone, the Visual pathway learns from RGB inputs using the contrastive InfoNCE objective, which mostly focuses on visual semantic information. To help enriching the representation of Visual and make it motion-aware, we introduce a Motion pathway trained on motion inputs. We then connect Motion to Visual using a novel cross-modal contrastive objective that enables the Motion pathway to guide the learning of Visual towards relevant motion cues. As our experiments show, this formulation leads to rich video representations that capture both visual semantics and motion patterns.

To evaluate MaCLR, we perform self-supervised pretraining on Kinetics-400 and transfer its representation to 5 video datasets for both action recognition (UCF101 [65], HMDB51 [43], Something-Something [1], VidSitu [59]) and action detection (AVA [32]). Without bells and whistle, MaCLR outperforms all previous video self-supervised methods on all datasets, under all evaluation settings. For example, MaCLR improves top-1 accuracy by 17% and 16.9% on UCF101 and HMD51, over previous SOTA trained on Kinetics-400. Furthermore, on Something-Something, VidSitu and AVA, MaCLR even outperforms its fully-supervised counterparts, demonstrating the strength of our approach.

2 Related Work

Self-supervised image representation learning. The goal of self-supervised image representation learning is to learn useful representations from large collections of unlabeled images. Early work focused on designing different pretext tasks with the intent of inducing generalizable semantic representations [20,50,51,84]. Though producing promising results, these methods could not match the performance of fully-supervised trained representations [42], as it is hard to prevent the network from utilizing shortcuts to solve pretext tasks (e.g., "chromatic aberration" in context prediction [20]). This changed when researchers re-visited the decade-old technique of contrastive learning [33,80]. Some of these recent work started to successfully produce results that were comparable to those of supervised learning on images [36,12,13,48,31,14,10]. Though related, these work were designed to learn from static images and thus cannot utilize the rich temporal information contained in videos.

Self-supervised video representation learning. Videos present unique opportunities to extract self-supervision and the literature offers different directions. The first line of research focuses on designing video-specific *pretext tasks*. Besides the work mentioned earlier [49,28,78,19,4], others attempt to learn video representations by either tracking across frames patches [76], pixels [77], colors [70], predicting temporal context for videos [58,73], or by enforcing consistency along videos semantics and play speeds [38]. A more recent line of work overcomes the need for pretext tasks by leveraging the *contrastive learning* paradigm [57,24]. Though successful to a certain extent, none of above methods *explicitly* make use of the important motion cues derived from the video temporal sequence. To better exploit such important information, [72] applies a pretext task of regressing motion statistics, [35,30] mine and cluster RGB images with similar motion cues, while [46,60,39] exploit the correspondences between RGB and motion *pixels*. MaCLR belongs to this recent class of works that aim at improving video representation using motion cues. However, it differs from previous works in the way it utilizes visual-motion correspondence in a cross-modal contrastive framework at a higher level than pixels, which yields a a method that is simpler, more robust and achieves considerably better results.

Motion in video tasks. Motion information has been heavily studied for many video tasks. As a prominent motion representation, optical flow has been utilized in many video action classification methods, either in the form of classical hand-crafted spatiotemporal features [44,16,71], or serve as input to deep CNN systems trained with supervised learning [25,26,74]. In contrast, our method focuses on exploiting motion information in the context of self-supervised learning. Beyond video classification, motion has also been exploited in many other tasks like video object detection [85,40,27,81], video frame prediction [61,45], video segmentation [68,3,15], object tracking [37,5,54], and 3D reconstruction [69].

3 MaCLR

We design MaCLR as a two-branch network consisting of a Visual pathway and a Motion pathway (Fig. 1a). The Visual pathway takes as input visual[1] clips and produces their visual embeddings. Similarly, the Motion pathway operates on motion clips (we will study different motion inputs in Sec. 3.2) and generates motion embeddings. MaCLR is trained using three contrastive learning objectives (Sec. 3.1): (i) a visual-only loss that pulls together visual clip embeddings that are sampled from the same video (solid green arrow in Fig. 1a) and pushes away that of different videos (solid red arrow); (ii) a motion-only loss that operates like (i), but on motion clips (omitted in Fig. 1a to avoid clutter) and (iii) a motion-visual loss to enforce alignment between embeddings of the visual and motion inputs (dashed arrows). As shown in Fig. 1a, we generate positive pairs from clips extracted from the same video (green arrows) and negative pairs from clips extracted from different videos (red arrows). After pretraining with MaCLR, *we then remove the Motion pathway and transfer the Visual pathway to target datasets for task-specific finetuning.*

3.1 Training MaCLR

Visual-only learning. We model this using a contrastive learning objective. Similar to [57], our model takes as input random clips with spatiotemporal jitterring. As shown in Fig. 1a, given a random clip we produce its embedding v^q (query), and sample a second positive clip from the same video and produce its embedding v^k (key), as well as N negative embeddings v_i^n, $i \in \{1, ..., N\}$ from other videos. Then, we train the Visual pathway with the InfoNCE objective $\mathcal{L}_v = \mathrm{IN}(v^q, v^k, v^n)$ [52,36]:

$$\mathcal{L}_v = -\log \frac{\exp(v^q \cdot v^k / \tau)}{\exp(v^q \cdot v^k / \tau) + \sum_{i=1}^{N} \exp(v^q \cdot v_i^n / \tau)}, \tag{1}$$

where τ is a temperature parameter. This objective ensures that our Visual pathway pulls together embeddings v^q and v^k, while pushing away those of all the negative clips v_i^n.

[1] *Sometimes also referred to as “RGB” in the literature.*

Motion-only learning. To improve the discriminativeness of the Motion pathway, we add another InfoNCE objective $\mathcal{L}_m = \text{IN}(m^q, m^k, m^n)$, which is trained in a similar way to $\mathcal{L}_v$ but this time on motion embeddings m^q, m^k (both are sampled from the same video as v_q) and m^n (which denotes a set of negative motion embeddings). This ensures that the Motion pathway is able to embed similar motion patterns close to each other.

Motion-Visual learning. We model this also with a contrastive learning objective, but with a different purpose compared to the previous two. Here, we aim at enriching the Visual pathway to be motion-aware with the help of the Motion pathway. Specifically, we train the model using the following InfoNCE objectives:

$$\mathcal{L}_{mv} = \text{IN}(v^q, m^k, v^n) + \text{IN}(m^q, v^k, m^n). \tag{2}$$

Note that v^q is *not necessarily in temporal synchronization* with m^k, but rather just a motion clip sampled from the same video (same for v^k and m^q). In our ablation, we show that allowing for this misalignment encourages the embedding to better learn semantic abstraction of visual and motion patterns, which leads to better performance.

One key difference to visual-only contrastive learning is on how we sample motion clips for both motion-only and motion-visual learning. Instead of sampling randomly, we constrain to only sample in temporal regions with strong motion cues. Specifically, we compute the sum of pixels P_i on the motion input and only sample frames with $\sum_{i=1}^{K} P_i/K > \gamma$, where K is the total number of pixels in a frame and γ is the threshold. This process helps avoid sampling irrelevant regions with no motion and thus leads to better representations.

Final training objective. The final training objective for MaCLR is the sum of all aforementioned loss functions:

$$\mathcal{L} = \mathcal{L}_v + \mathcal{L}_m + \mathcal{L}_{mv}. \tag{3}$$

Training MaCLR end-to-end is non-trivial, as video representation are expensive to compute and to maintain (as contrastive learning requires large batch sizes [12]). Inspired by [80,36], we solve this problem by adopting the idea of memory bank for negative samples. Specifically, we construct two memory banks of negative samples for visual and motion inputs, and maintain a momentum version of the Motion and Visual pathways updated as a moving average of their online counterparts with momentum coefficient λ: $\theta' \leftarrow \lambda\theta' + (1-\lambda)\theta$, where θ and θ' are weights for the online and momentum version of the model respectively. One caveat is that when pushing negatives into the pool, we push the video index, along with the embedding, so that we can avoid sampling visual or motion clips that are from the same video as positive clips, which would otherwise confuse the network and hurt the representations. Similar to [36], we forward queries through the online model and keys through the momentum model to produce embeddings.

3.2 Design of motion inputs

There are many ways to represent a motion input. A straightforward way is to directly compute the difference of pixel values between two consecutive frames. While capturing motion to a certain extent, it also captures undesired signals like pixel value shifts caused by background motion (e.g., sea-wave in Fig. 1b top-right). A more appropriate representation might be optical flow [8,79,21,67]. However, a disadvantage of feeding in raw optical flow (or flow vector magnitude, as used in [29]) is that it is heavily influenced by factors like illumination change (Fig. 1b bottom-left) and it also captures absolute flow magnitude, which is not very useful for learning general motion patterns. To overcome these limitations, in MaCLR, we propose to use flow edge maps as inputs to the Motion pathway network. Specifically, we apply a Sobel filter [64] onto the flow magnitude map to produce the flow edges (Fig. 1b bottom-right). In our experiments, this simple operation turns out to produce significantly better motion representations that focus on foreground motion regions.

3.3 Visual and Motion pathway architectures

Our *Visual pathway* is a 3D ResNet50 (R3D-50) with a structure similar to that of "Slow-only" in [23,57], which features 2D convs in res_2, res_3 and non-degenerate 3D convs in res_4, res_5. It takes as input a tensor of size $3{\times}8{\times}224^2$, capturing 8 frames of size 224×224. The sampling stride is 8, which means that the visual input clip spans 8×8 frames, corresponding to ~2 seconds for videos at 30 FPS. To have larger temporal receptive field, we set the temporal kernel size of $\texttt{conv}_1$ to 5 following [57].

Our *Motion pathway* is a 2D ResNet50. and it takes as input a tensor of size $3{\times}16{\times}224^2$, stacking 16 motion frames. We use a sampling stride of 4, so that it spans for the same time as the visual input (i.e., ~2 secs). Following the design philosophy of SlowFast Networks [23], we design our Motion pathway to be much more lightweight compared to our Visual pathway (1/8 channel sizes across the network), as motion inputs have intrinsically less variability (i.e., no variations on colors, illumination, etc.).

4 Experiments

4.1 Implementation Details

MaCLR training details. We train MaCLR on the Kinetics-400 (K400) dataset (CC-BY-4.0) [41]. The dataset consists of ~240k video clips that span at most 10 seconds. These were originally annotated with 400 different action classes, but we *do not* use any of these labels. We train MaCLR for 600 epochs on the whole 240k videos when we compare against the literature. For our ablation study, instead, we compare different variants of MaCLR trained for 100 epochs on a subset of 60k videos ("K400-mini"). We use a pool size (N in Eq. 1) of 65536

method	data	UCF	HMDB
V-only	K400-mini	63.6	33.7
M-only	K400-mini	66.4	45.1
MaCLR	K400-mini	**78.1**	**47.2**
V-only	K400	74.6	46.3
MaCLR	K400	**85.5**	**57.7**

(a) **Motion-visual learning**

inputs	UCF	HMDB
Diff	71.6	40.1
Flow	74.1	44.2
Edge	**78.1**	**47.2**

(b) **Motion inputs**

components	UCF	HMDB
MaCLR	**78.1**	**47.2**
−t. jitter	77.4	47.1
−m. thresh	77.3	46.4
$-\mathcal{L}_m$	77.8	46.6

(c) **Dissect components**

Table 1: **Ablating MaCLR**. We present top-1 classification accuracy using the Linear Layer Training evaluation protocol (sec. 4.2). In (a), V-only and M-only refers to the visual and motion only pretraining. In (b), Diff, Flow and Edge refer to motion inputs in the form of Frame Difference, Optical Flow and Flow Edges, respectively. Experiments in (b) and (c) are conducted on K400-mini. We use 8×8 R3D-50 model for finetuning.

negative samples for both visual and motion inputs. We set the momentum update coefficient $\lambda = 0.999$ and temperature τ to 0.1. The embedding dimension is set to 128 for both Visual and Motion pathways. For the visual inputs, we apply random spatial cropping, temporal jittering, $p = 0.2$ probability grayscale conversion, $p = 0.5$ horizontal flip, $p = 0.5$ Gaussian blur, and $p = 0.8$ color perturbation on brightness, contrast and saturation, all with 0.4 jittering ratio. For motion inputs, we randomly sample flow edge clips in high motion regions (with motion threshold γ set to 0.02) and skip other augmentations. Our codebase is based on PySlowFast [22].

Flow Edge Maps. To compute flow edge map for frame t, we first compute optical flow from frame t to $t-5$, using `RAFT-things` [67] model trained entirely on synthetic data without human annotations. We hypothesize it would also work with flow computed from closer pairs, as long as the motion threshold γ is adjusted accordingly. Then, we apply a Sobel filter onto the magnitude map of optical flow and clamp the resulting edge map in [0, 10] as the final flow edge map. We note that this is an offline pre-processing that only needs to be done once and reused throughout training (and never during inference).

Baselines. We compare against two baselines: (i) *Self-Supervised Visual-only* is a strong self-supervised representation trained from RGB inputs using only the contrastive learning objective of Eq. 1 (i.e., without our motion learning objectives $\mathcal{L}_{mv}$ and $\mathcal{L}_m$); and (ii) *Supervised* is a fully supervised model trained for action classification on K400. Both baselines use a R3D-50 backbone.

4.2 Action Recognition on UCF101 and HMDB51

Datasets and evaluation protocol. We first evaluate MaCLR for action recognition on the two most popular datasets in the literature: UCF101 [65] and HMDB51 [43] (CC-BY-3.0). We follow the standard settings to perform self-supervised training on K400 and then transfer the learned weights to target datasets for evaluation. Two evaluation protocols are employed in the literature to evaluate the quality of the self-supervised representation: (i) *Linear Layer Training* freezes the trained backbone and simply trains a linear classifier on the target dataset, while (ii) *Full Network Training* finetunes the entire network end-to-end on the target dataset. For completeness, we evaluate using both protocols and report action classification top-1 accuracy. For all experiments on UCF101 and HMDB51, we report results using `split1` for train/test split. In total, there are 9.5k/3.7k train/test videos with 101 action classes in UCF101, and 3.5k/1.5k train/test videos with 51 actions in HMDB51. We use the standard 10 (temporal) ×3 (spatial) crop sampling during test [75,23]. We use these two datasets to compare against the state-of-the-art (SOTA). Additionally, we use K400-mini to conduct an extensive ablation study on the components of MaCLR. For the comparison with SOTA, we pretrain MaCLR with 8×8 inputs for 600 epochs on K400, and finetune with 32×8 inputs on downstream tasks, as these leads to the best performance. In our ablation study instead we simplify these settings for efficiency and pretrain for only 100 epochs and use 8×8 inputs for finetuning.

Ablation: motion-visual learning (Table 1a). First and foremost, we study the importance of enriching visual embeddings with motion cues using the proposed motion-visual learning objective of Eq. 2. Results show that MaCLR improves substantially over Visual-only on both UCF and HMDB, when pretrained with either K400 or K400-mini. To understand if the benefit comes purely from the new motion objective $\mathcal{L}_m$, we also trained a Motion-only model on K400-mini. Interestingly, this model performs slightly better than Visual-only, but much worse than MaCLR, showing the importance of training a video representation that can capture *both* semantic and motion features. Finally, note how MaCLR trained on K400-mini also outperforms the Visual-only baseline pretrained on the full K400 (4× more data): +3.5/+0.9 on UCF/HMDB.

Ablation: motion representations (Table 1b). In Sec. 3 we discussed some conceptual advantages of using flow edge maps and here we evaluate it against two popular motion alternatives: Frame Difference and Optical Flow. As shown in Table 1b, Flow Edges is indeed the best way to represent motion for self-supervised training, thanks to its ability to prune background motion noise and absolute motion magnitude. That being said, even the much weaker Frame Difference representation outperforms the Visual-only baseline (Table 1a) by +8.0 top-1 accuracy on UCF and +6.4 on HMDB. This further confirms the importance of enriching video representations with motion cues.

Ablation: MaCLR components (Table 1c). We now dissect MaCLR to study the importance of its components.
Temporal Jittering. Unlike previous work that learn self-supervised representation by exploiting pixel-level correspondences between RGB and optical flow inputs [46,60], we demonstrate that it's more effective to learn self-supervised representations by introducing temporal "misalignment" between them. Specif-

ically, we compare MaCLR, which trains on RGB and motion clips that are temporally jittered, against a variant that is trained on synchronized RGB and motion clips (i.e., sync pairs [v^q, m^k] and [m^q, v^k] in Eq. 2). Our results show that the misaligned inputs lead to better representations (+0.7 on UCF), as it prevents the model from exploiting the shortcut of finding pixel correspondences using low-level visual cues.

Motion thresholding. Next, we study the motion input sampling strategy discussed in Sec. 3.1. We compare MaCLR to a variant which randomly samples motion input clips, without removing those with little motion (i.e., setting threshold $\gamma = 0$, Sec. 4.1). Without this threshold, top-1 accuracy degrades by -0.8 on both datasets, due to the noise introduced by clips with too little motion.

Motion loss $\mathcal{L}_m$. Finally, we study whether it's necessary to have the extra contrastive objective $\mathcal{L}_m$ between motion inputs (Eq. 3), which is included to help training more discriminative motion embeddings. Results show that this motion discrimination objective is indeed useful as it improves top-1 acc by +0.3 and +0.6 on UCF101 and HMDB51.

Comparison to state-of-the-art (Table 2). We now compare MaCLR against previous self-supervised video representation learning methods in the literature using both the evaluation protocols introduced at the beginning of Sec. 4.2: Linear (✓ for column "Frozen") and Full (✗).

By only training a linear layer on top of our self-supervised learned representation, our method is able to achieve significantly better top-1 classification accuracy compared to the previous state-of-the-art trained on K400: +16.7 and +16.9 over CoCLR on UCF101 and HMDB51, respectively. Only the recent CVRL method comes close to our results on UCF, but still lacks on HMDB (−4.7). Moreover, MaCLR outperforms all previous methods, including those trained on 100×more data than K400 (e.g., IG65M and Youtube8M), and those that use extra modalities like audio and text (e.g., XDC, MIL-NCE).

Results using the end-to-end full training evaluation protocol show similar observations to the linear evaluation protocol: MaCLR again achieves competitive results among the methods trained on K400. When compared to previous approaches, only ρBYOL, XDC and GDT produce results comparable to MaCLR. Among them, ρBYOL is conceptually similar to our visual-only method, but augmented with the idea of sampling multiple clips ($\rho = 4$) per video for training, which is complementary to our main contribution. On the other hand, both XDC and GDT are trained on 270×more data (IG65M contains 21 years of video content vs.K400 only 28 days) and use extra audio modality as inputs. Furthermore, towards making the best effort in enabling fair comparison against the literature, we also present the results of a weaker MaCLR model trained with a smaller backbone (R18) and a smaller input size (128×128). Under this setting, our model again convincingly outperforms models with similar backbone and input resolutions (e.g., 3D-RotNet, CBT, GDT, CoCLR).

We also tried to keep the Motion pathway during inference and ensemble its prediction with those of the Visual pathway ("V+F"). This produces results

Method	Date	Data (duration)	Arch.	Size	Modality	Frozen	UCF	HMDB
MemDPC [34]	2020	K400 (28d)	R-2D3D-34	224^2	V	✓	54.1	30.5
MIL-NCE [47]	2020	HTM (15y)	S3D	224^2	V+T	✓	82.7	53.1
MIL-NCE [47]	2020	HTM (15y)	I3D	224^2	V+T	✓	83.4	54.8
XDC [2]	2020	IG65M (21y)	R(2+1)D	224^2	V+A	✓	85.3	56.0
ELO [55]	2020	YT-8M (8y)	R(2+1)D	224^2	V+A	✓	–	64.5
AVSlowFast [82]	2020	K400 (28d)	AVSlowFast-50	224^2	V+A	✓	77.4	42.2
CoCLR [35]	2020	K400 (28d)	S3D	128^2	V	✓	74.5	46.1
CVRL [57]	2021	K400 (28d)	R3D-50	224^2	V	✓	89.8	58.3
MLFO [56]	2021	K400 (28d)	R3D-18	112^2	V	✓	63.2	33.4
BraVe [58]	2021	K600 (36d)	R3D-50	224^2	V	✓	88.8	61.8
MaCLR		K400 (28d)	R3D-18	128^2	V	✓	**90.4**	**57.5**
MaCLR		K400 (28d)	R3D-50	224^2	V	✓	**91.5**	**63.0**
w/o Pretrain		-	R3D-50	224^2	V	✗	69.0	22.7
CBT [66]	2019	K600+ (273d)	S3D	112^2	V	✗	79.5	44.6
DynamoNet [19]	2019	YT-8M-1 (58d)	STCNet	112^2	V	✗	88.1	59.9
XDC [2]	2020	IG65M (21y)	R(2+1)D	224^2	V+A	✗	94.2	67.4
AVSlowFast [82]	2020	K400 (28d)	AVSlowFast-50	224^2	V+A	✗	87.0	54.6
SpeedNet [4]	2020	K400 (28d)	S3D-G	224^2	V	✗	81.1	48.8
MemDPC [34]	2020	K400 (28d)	R-2D3D-34	224^2	V	✗	86.1	54.5
CoCLR [35]	2020	K400 (28d)	S3D	128^2	V	✗	87.9	54.6
GDT [53]	2020	K400 (28d)	R(2+1)D	112^2	V+A	✗	89.3	60.0
GDT [53]	2020	IG65M (21y)	R(2+1)D	112^2	V+A	✗	95.2	72.8
MIL-NCE [47]	2020	HTM (15y)	S3D	224^2	V+T	✗	91.3	61.0
ELO [55]	2020	YT-8M-2 (13y)	R(2+1)D	224^2	V+A	✗	93.8	67.4
CVRL [57]	2021	K400 (28d)	R3D-50	224^2	V	✗	92.9	67.9
MLFO [56]	2021	K400 (28d)	R3D-18	112^2	V	✗	79.1	47.6
ρBYOL [24]	2021	K400 (28d)	R3D-50	224^2	V	✗	**94.2**	**72.1**
MotionFit [30]	2021	K400 (28d)	S3D-G	224^2	V	✗	90.1	50.6
ASCNet [38]	2021	K400 (28d)	S3D-G	224^2	V	✗	90.8	60.5
BraVe [58]	2021	K600 (36d)	R3D-50	224^2	V	✗	92.6	69.2
MaCLR		K400 (28d)	R3D-18	128^2	V	✗	91.3	62.1
MaCLR		K400 (28d)	R3D-50	224^2	V	✗	94.0	67.4
MaCLR		K400 (28d)	R3D-50	224^2	V+F	✗	**94.2**	67.3
Fully-Supervised [83]		K400 (28d)	S3D	224^2	V	✗	96.8	75.9

Table 2: **Comparison with state-of-the-art approaches.** We report top-1 accuracy. In parenthesis, we show the total video duration in time (**d** for day, **y** for year). The top half of the table contains results for the Linear protocol (Frozen ✓), whereas the bottom half shows results for the Full end-to-end finetuning protocol (Frozen ✗). For Modality, V: visual only, A: audio, T: text narration.

that are nearly identical to those obtained using only our motion-aware Visual pathway ("V"), which suggests that our novel training paradigm is indeed able to successfully "distill" motion information into the Visual pathway during pretraining. Finally, we also performed k-nearest-neighbor video retrieval to compare to the recent ASCNet work [38] in Table 3a. Despite similar accuracy when k=10, we largely outperform under the strictest 1-NN setting (+2.8%), which shows the higher precision of our representations.

Low-shot finetuning. We further investigate how the performance of MaCLR varies with respect to the amount of data available for finetuning on the target task. We evaluate using the Full Training protocol on the UCF101 dataset starting from just 1% of its training data (1 video per class) and gradually increase that to 100% (9.5k videos). We compare results against our two baselines:

method	1-NN	10-NN
ASCNet R18	58.9	82.2
MaCLR R18	**61.7**	**82.2**
MaCLR R50	73.4	88.2

(a) **kNN video retrieval**

method	1%	5%	20%	40%	60%	80%	100%
Supervised	69.3	85.1	93.0	94.5	94.7	95.8	95.4
Visual-only	32.9	62.8	82.2	86.5	87.8	89.5	89.0
MaCLR	42.8	71.9	89.1	91.3	92.9	93.4	94.0
Δ	+9.9	+9.1	+6.9	+4.8	+5.1	+3.9	+5.0

(b) **Low-shot learning on UCF101**

Table 3: **Video retrieval and low-shot learning on UCF101.** (a) reports kNN video retrieval results on split1 of UCF101. A video is considered to be correctly predicted if its ground-truth label is among the labels of its k nearest neighbors retrieved from the training set. (b) Rows indicate different pretrainings on K400, while columns vary the % of UCF training data used for finetuning. All results are top-1 accuracy.

Visual-only and Supervised (Table 3b). MaCLR outperforms Visual-only across all training set sizes and it only requires 20% of the training videos to match the performance of Visual-only with 100% (89.1 vs 89.0). Another interesting observation is that the gap Δ between MaCLR and Visual-only reaches its maximum with the smallest training set (1%), suggesting that motion-visual learning is particularly helpful for generalization in low-shot scenarios.

4.3 Action Recognition on Something-Something

Next we evaluate MaCLR on Something-Something-v2 (SSv2) [1], a challenging action classification dataset that is heavily focused on motion. Different from UCF101 and HMDB51 which contain action classes similar to K400, SSv2 contains a very different set of actions featuring complex human object interactions, like "Moving something up" and "Pushing something from left to right". The dataset consists of 168k training, 24k validation and 24k test videos, all annotated with 174 action classes. We finetune on SSv2 with a recipe that mostly follows the official implementation of [23]: we use a clip size of 16×8 and a batch-size of 16 (over 8 GPUs); we train for 22 epochs with an initial learning rate of 0.03 and decay it by 10× twice at 14 and 18 epochs; and a learning rate warm-up is scheduled for 0.19 epochs starting from a learning rate of 0.0001.

We evaluate using both the Linear and Full finetune protocol. We compare methods that are pretrained in different ways: MaCLR and the Visual-only baseline are pretrained self-supervisedly on K400, whereas R3D-50 [23] is pretrained with full supervision on K400. Rand Init is a randomly initialized network without pretraining (Table 4a).

For the Full protocol evaluation, it's clear that pretraining on K400 is beneficial and improves by almost +10 top-1 accuracy. Next, MaCLR outperforms the Visual-only baseline, showing once more the importance of learning from the added Motion pathway. Finally, when comparing to R3D-50 pretrained with full

method	pretrain	sup.	acc@1	acc@5	rec@5
Slow+NL	K400	✓	29.1	58.7	**19.2**
R3D-50	K400	✓	38.3	69.3	18.7
Visual-only	K400	✗	32.8	61.6	13.6
MaCLR	K400	✗	**43.0**	**73.2**	17.5

(a) **Action classification on SSv2**

method	pretrain	sup.	Full	Linear
R3D-50	K400	✓	55.5	16.3
Rand Init	-	✗	45.4	-
Visual-only	K400	✗	54.9	16.6
MaCLR	K400	✗	**57.4**	**27.1**

(b) **Verb prediction on VidSitu**

Table 4: **Results on SSv2 and VidSitu.** (a) reports top-1 accuracy. For finetuning, we use 16×8 clip as input following [23]. (b) reports top-1, top-5 accuracy and macro-averaged recall with five predictions on val set following [59]. All models use a R3D-50 backbone with 16×4 inputs.

supervision, MaCLR not only closes the gap between self-supervised and fully-supervised methods, but even outperforms the supervised pretraining (+1.9).

Furthermore, we test with the Linear protocol, which is much more challenging due to the large difference between the label spaces of K400 and SSv2. As expected, Table 4a shows that the accuracy of all methods is much lower compared to their Full finetune results. However, it's notable that the gap between MaCLR and Visual-only significantly increases (+10.5 vs +2.5) compared to the Full protocol, which further demonstrates our method's generalization strength. Moreover, it's interesting to see the supervised baseline underperform both self-supervised methods, as it's harder to overcome taxonomy bias under Linear protocol compared to the Full protocol for a representation pretrained with a fixed label taxonomy. We believe this is a promising example showing how self-supervised training can remove the label taxonomy bias that is inevitable under supervised settings, and lead to more general video representations that can be better transferred to new domains.

In this section we evaluate how MaCLR pretrained on YouTube-style short clips (K400) generalizes to a very different video domain: movie clips. For this, we evaluate our video representation on the recent VidSitu benchmark [59] which features 30k movie clips from 3k different movies. Specifically, we benchmark on the *verb prediction* task of VidSitu, which contains 1560 action classes (e.g., speak, walk, run, climb). We compare different pretraining strategies, using the same R3D-50 backbone with 16×4 inputs, and we evaluate verb prediction results with top-1/top-5 accuracy and the macro-averaged recall metric with five predictions, as in [59]. The results are shown in Table 4b. For the supervised "R3D-50" baseline, we pretrained its backbone using the K400 labels and then fine-tuned it on VidSitu. As for self-supervised pretraining, we evaluate both the Visual-only baseline and our MaCLR. Similar to our observations on SSv2, MaCLR outperforms all methods on both acc@1 and acc@5 metrics. The improvement over the Visual-only baseline is also particularly substantial, which suggests that motion information is particularly important to help self-supervised representation generalize to different video domains.

method	pretrain dataset	sup.	mAP
Faster-RCNN [23]	ImageNet	✓	15.3
Faster-RCNN [23]	K400	✓	21.9
Rand Init	-	✗	6.6
CVRL [57]	K400	✗	16.3
Visual-only	K400	✗	18.6
MaCLR	K400	✗	**22.1**

Table 5: **Action detection on AVA.** We use 8×8 clip as input for finetuning [23]. CVRL numbers are taken from [57].

4.4 Action Detection on AVA

In the previous section we showed that MaCLR can generalize to new video domains within the same downstream task (i.e., action recognition). However, we believe that our self-supervised representation can go beyond that and also generalize to novel downstream tasks, since it is not optimized for any task specific objective. To test this, we transfer MaCLR representation to the new task of action detection, which requires not only to recognize the action class, but also localize the person performing the action.

We evaluate action detection on the AVA dataset (CC-BY-4.0) [32] which contains 211k training and 57k validation videos. Spatiotemporal labels (i.e., action classes and bounding boxes) are provided at 1 FPS rate. We follow the standard evaluation protocol and compute mean Average Precision over 60 classes, using an IOU threshold of 0.5. We follow the Faster-RCNN detector design of [23] and use the Visual pathway architecture of Sec. 3.3 as the detector backbone. We fix the training schedule to 20 epochs with an initial learning rate of 0.1 and a batch size of 64 [23].

Results are shown in Table 5. Clearly, video pretraining plays a critical role in action detection, as demonstrated by the low mAP of 6.6 when training from scratch and the substantially lower AP achieved by supervised pretraining on ImageNet (pretrained 2D convs are inflated into 3D for fine-tuning [11]) compared to supervised pretraining on K400. As for self-supervised pretraining, both the Visual-only baseline and MaCLR outperform ImageNet supervised pretraining, again demonstrating the importance of pretraining on videos. Moreover, MaCLR again outperforms both the Visual-only baseline and the recent CVRL approach, which also only uses RGB inputs for pretraining.

Finally, note how MaCLR even outperforms the supervised Faster-RCNN pretrained on K400. To the best of our knowledge, we are the first to demonstrate that self-supervised video learning can transfer to action detection and match the performance of fully-supervised pretraining.

4.5 Visualizing MaCLR Representations

To gain deeper insights on what MaCLR has learned in its representations, we adopt Grad-CAM [62] to visualize the spatiotemporal regions that contribute

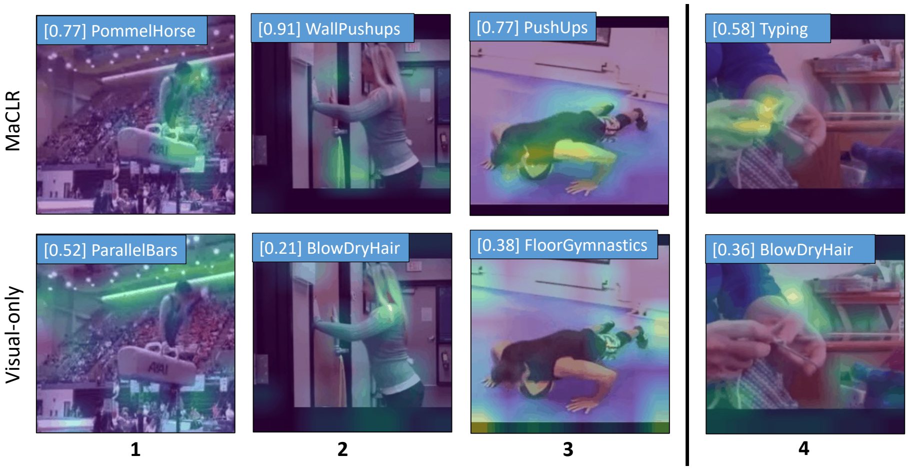

Fig. 2: **Grad-CAM visualization for MaCLR (top) and Visual-only (bottom) representations**. Predictions are overlaid on each frame.

the most to the classification decisions on UCF101. As shown in Fig. 2, we observe that the representation learned by MaCLR focuses more on the "motion-sensitive" regions (i.e., regions where object motion likely occur). For example, in col-1, MaCLR makes the correct prediction of "PommelHorse" by focusing its attention on the person carrying out the motion. The Visual-only model, on the other hand, incorrectly predicted "ParallelBars" as it finds "bar-like" straight lines in the background. This pattern can also be observed in col-2 (Visual-only model predicts "BlowDryHair" after finding hair textures). Furthermore, we can observe another type of behavior in col-3. In both examples, the background scenes (gym) are associated with many fine-grained action classes (different gym activities), our model is able to distinguish them by focusing on the actual motion pattern. The baseline, instead, gets confused as it focuses too much on the background. Finally, we present a failure case in the last column where MaCLR correctly focuses on the right motion region (fingers), but confuses the finger motion of "Knitting" with "Typing".

Conclusion

We presented MaCLR to learn self-supervised video representations with explicit cross-modal motion-visual contrastive learning. We demonstrated SOTA self-supervised performance with MaCLR across various datasets and tasks. Moreover, we showed that MaCLR representations can be as effective as representations learned with full supervision for SSv2 action recognition, VidSitu verb prediction and AVA action detection. Given the simplicity of our method, we hope it will serve as a strong baseline for future research in self-supervised video representation learning.

References

1. 20BN-Something-Something Dataset V2 3, 11
2. Alwassel, H., Mahajan, D., Korbar, B., Torresani, L., Ghanem, B., Tran, D.: Self-supervised learning by cross-modal audio-video clustering. In: NeurIPS (2020) 10
3. Bao, L., Wu, B., Liu, W.: CNN in MRF: Video object segmentation via inference in a CNN-based higher-order spatio-temporal MRF. In: CVPR (2018) 4
4. Benaim, S., Ephrat, A., Lang, O., Mosseri, I., Freeman, W.T., Rubinstein, M., Irani, M., Dekel, T.: SpeedNet: Learning the speediness in videos. In: CVPR (2020) 2, 3, 10
5. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.: Fully-convolutional siamese networks for object tracking. In: ECCV (2016) 4
6. Brattoli, B., Buchler, U., Wahl, A.S., Schwab, M.E., Ommer, B.: LSTM self-supervision for detailed behavior analysis. In: CVPR (2017) 2
7. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: NeurIPS (2020) 1
8. Brox, T., Malik, J.: Large displacement optical flow: descriptor matching in variational motion estimation. T-PAMI (2011) 6
9. Buchler, U., Brattoli, B., Ommer, B.: Improving spatiotemporal self-supervision by deep reinforcement learning. In: ECCV (2018) 2
10. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. In: NeurIPS (2020) 3
11. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: CVPR (2017) 13
12. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: ICML (2020) 2, 3, 5
13. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020) 3
14. Chen, X., He, K.: Exploring simple siamese representation learning. In: CVPR (2021) 3
15. Cheng, J., Tsai, Y.H., Hung, W.C., Wang, S., Yang, M.H.: Fast and accurate online video object segmentation via tracking parts. In: CVPR (2018) 4
16. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: ECCV (2006) 4
17. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A Large-Scale Hierarchical Image Database. In: CVPR (2009) 1
18. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL (2019) 1
19. Diba, A., Sharma, V., Gool, L.V., Stiefelhagen, R.: DynamoNet: Dynamic action and motion network. In: ICCV (2019) 2, 3, 10
20. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised Visual Representation Learning by Context Prediction. In: ICCV (2015) 3
21. Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazrba, C., Golkov, V., Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. In: ICCV (2015) 6
22. Fan, H., Li, Y., Xiong, B., Lo, W.Y., Feichtenhofer, C.: Pyslowfast. `https://github.com/facebookresearch/slowfast` (2020) 7

23. Feichtenhofer, C., Fan, H., Malik, J., He, K.: SlowFast Networks for Video Recognition. In: ICCV (2019) 6, 8, 11, 12, 13
24. Feichtenhofer, C., Fan, H., Xiong, B., Girshick, R., He, K.: A large-scale study on unsupervised spatiotemporal representation learning. In: CVPR (2021) 3, 10
25. Feichtenhofer, C., Pinz, A., Wildes, R.: Spatiotemporal residual networks for video action recognition. In: NeurIPS (2016) 4
26. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: CVPR (2016) 2, 4
27. Feichtenhofer, C., Pinz, A., Zisserman, A.: Detect to track and track to detect. In: ICCV (2017) 4
28. Fernando, B., Bilen, H., Gavves, E., Gould, S.: Self-supervised video representation learning with odd-one-out networks. In: CVPR (2017) 2, 3
29. Fragkiadaki, K., Arbelaez, P., Felsen, P., Malik, J.: Learning to segment moving objects in videos. In: CVPR (2015) 6
30. Gavrilyuk, K., Jain, M., Karmanov, I., Snoek, C.G.: Motion-augmented self-training for video recognition at smaller scale. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021) 3, 10
31. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.A., Guo, Z.D., Azar, M.G., et al.: Bootstrap your own latent: A new approach to self-supervised learning. In: NeurIPS (2020) 3
32. Gu, C., Sun, C., Ross, D.A., Vondrick, C., Pantofaru, C., Li, Y., Vijayanarasimhan, S., Toderici, G., Ricco, S., Sukthankar, R., et al.: AVA: A video dataset of spatio-temporally localized atomic visual actions. In: CVPR (2018) 3, 13
33. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: CVPR (2006) 3
34. Han, T., Xie, W., Zisserman, A.: Memory-augmented dense predictive coding for video representation learning. In: ECCV (2020) 10
35. Han, T., Xie, W., Zisserman, A.: Self-supervised co-training for video representation learning. In: NeurIPS (2020) 3, 10
36. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020) 2, 3, 4, 5
37. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. T-PAMI (2014) 4
38. Huang, D., Wu, W., Hu, W., Liu, X., He, D., Wu, Z., Wu, X., Tan, M., Ding, E.: ASCNet: Self-supervised video representation learning with appearance-speed consistency. In: ICCV (2021) 3, 10
39. Huang, L., Liu, Y., Wang, B., Pan, P., Xu, Y., Jin, R.: Self-supervised video representation learning by context and motion decoupling. In: CVPR (2021) 3
40. Kang, K., Li, H., Xiao, T., Ouyang, W., Yan, J., Liu, X., Wang, X.: Object detection in videos with tubelet proposal networks. In: CVPR (2017) 4
41. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017) 1, 6
42. Kolesnikov, A., Zhai, X., Beyer, L.: Revisiting self-supervised visual representation learning. In: CVPR (2019) 3
43. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: ICCV (2011) 3, 7
44. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR (2008) 4
45. Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., Yang, M.H.: Flow-grounded spatial-temporal video prediction from still images. In: ECCV (2018) 4

46. Mahendran, A., Thewlis, J., Vedaldi, A.: Cross pixel optical-flow similarity for self-supervised learning. In: ACCV (2018) 3, 8
47. Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., Zisserman, A.: End-to-end learning of visual representations from uncurated instructional videos. In: CVPR (2020) 10
48. Misra, I., Maaten, L.v.d.: Self-supervised learning of pretext-invariant representations. In: CVPR (2020) 3
49. Misra, I., Zitnick, C.L., Hebert, M.: Shuffle and learn: unsupervised learning using temporal order verification. In: ECCV (2016) 2, 3
50. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: ECCV (2016) 3
51. Noroozi, M., Pirsiavash, H., Favaro, P.: Representation learning by learning to count. In: ICCV (2017) 3
52. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018) 4
53. Patrick, M., Asano, Y.M., Kuznetsova, P., Fong, R., Henriques, J.F., Zweig, G., Vedaldi, A.: Multi-modal self-supervision from generalized data transformations. In: ICCV (2021) 10
54. Perazzi, F., Khoreva, A., Benenson, R., Schiele, B., Sorkine-Hornung, A.: Learning video object segmentation from static images. In: CVPR (2017) 4
55. Piergiovanni, A., Angelova, A., Ryoo, M.S.: Evolving losses for unsupervised video representation learning. In: CVPR (2020) 10
56. Qian, R., Li, Y., Liu, H., See, J., Ding, S., Liu, X., Li, D., Lin, W.: Enhancing self-supervised video representation learning via multi-level feature optimization. In: ICCV (2021) 10
57. Qian, R., Meng, T., Gong, B., Yang, M.H., Wang, H., Belongie, S., Cui, Y.: Spatiotemporal contrastive video representation learning. In: CVPR (2021) 3, 4, 6, 10, 13
58. Recasens, A., Luc, P., Alayrac, J.B., Wang, L., Strub, F., Tallec, C., Malinowski, M., Patraucean, V., Altché, F., Valko, M., et al.: Broaden your views for self-supervised video learning. In: ICCV (2021) 3, 10
59. Sadhu, A., Gupta, T., Yatskar, M., Nevatia, R., Kembhavi, A.: Visual semantic role labeling for video understanding. In: CVPR (2021) 3, 12
60. Sayed, N., Brattoli, B., Ommer, B.: Cross and learn: Cross-modal self-supervision. In: German Conference on Pattern Recognition (2018) 3, 8
61. Sedaghat, N., Zolfaghari, M., Brox, T.: Hybrid learning of optical flow and next frame prediction to boost optical flow in the wild. arXiv preprint arXiv:1612.03777 (2016) 4
62. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: ICCV (2017) 13
63. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. In: ICLR (2015) 2
64. Sobel, I.: History and definition of the sobel operator (2014) 6
65. Soomro, K., Zamir, A.R., Shah, M.: A dataset of 101 human action classes from videos in the wild. In: ICCV Workshops (2013) 3, 7
66. Sun, C., Baradel, F., Murphy, K., Schmid, C.: Contrastive bidirectional transformer for temporal representation learning. arXiv preprint arXiv:1906.05743 (2019) 10
67. Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: ECCV (2020) 6, 7

68. Tsai, Y.H., Yang, M.H., Black, M.J.: Video segmentation via object flow. In: CVPR (2016) 4
69. Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., Brox, T.: Demon: Depth and motion network for learning monocular stereo. In: CVPR (2017) 4
70. Vondrick, C., Shrivastava, A., Fathi, A., Guadarrama, S., Murphy, K.: Tracking emerges by colorizing videos. In: ECCV (2018) 3
71. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: ICCV (2013) 4
72. Wang, J., Jiao, J., Bao, L., He, S., Liu, Y., Liu, W.: Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In: CVPR (2019) 3
73. Wang, J., Bertasius, G., Tran, D., Torresani, L.: Long-short temporal contrastive learning of video transformers. arXiv preprint arXiv:2106.09212 (2021) 3
74. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: ECCV (2016) 2, 4
75. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: CVPR (2018) 8
76. Wang, X., Gupta, A.: Unsupervised Learning of Visual Representations using Videos. In: ICCV (2015) 3
77. Wang, X., Jabri, A., Efros, A.A.: Learning correspondence from the cycle-consistency of time. In: CVPR (2019) 3
78. Wei, D., Lim, J.J., Zisserman, A., Freeman, W.T.: Learning and using the arrow of time. In: CVPR (2018) 2, 3
79. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: Deepflow: Large displacement optical flow with deep matching. In: ICCV (2013) 6
80. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: CVPR (2018) 3, 5
81. Xiao, F., Lee, Y.J.: Video object detection with an aligned spatial-temporal memory. In: ECCV (2018) 4
82. Xiao, F., Lee, Y.J., Grauman, K., Malik, J., Feichtenhofer, C.: Audiovisual slowfast networks for video recognition. arXiv preprint arXiv:2001.08740 (2019) 10
83. Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In: ECCV (2018) 10
84. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV (2016) 3
85. Zhu, X., Wang, Y., Dai, J., Yuan, L., Wei, Y.: Flow-guided feature aggregation for video object detection. ICCV (2017) 4