Self-Supervised Spatial Temporal Feature Learning for Video Correspondence

Anonymous Author(s)
Affiliation
Address
email

Abstract

This paper proposes to learn reliable dense correspondence from videos in a self-supervised manner. Our learning process integrates two highly related tasks: tracking large image regions and establishing fine-grained pixel-level associations between consecutive video frames. We exploit the synergy between both tasks through a shared inter-frame affinity matrix, which simultaneously models transitions between video frames at both the region- and pixel-levels. While region-level localization helps reduce ambiguities in fine-grained matching by narrowing down search regions; fine-grained matching provides bottom-up features to facilitate region-level localization. Our method outperforms the state-of-the-art self-supervised methods on a variety of visual correspondence tasks, including video-object and part-segmentation propagation, keypoint tracking, and object tracking. Our self-supervised method even surpasses the fully-supervised affinity feature representation obtained from a ResNet-18 pre-trained on the ImageNet.

1 Introduction

Learning representations for video correspondence is a fundamental problem in computer vision, which is closely related to different video applications, including optical flow estimation, video object segmentation, keypoint tracking, etc. However, supervising such a representation requires a large number of dense annotations, which is unaffordable. Thus most approaches acquire supervision from simulations or limited annotations, which result in poor generalization in different downstream tasks. Recently, self-supervised feature learning is gaining significant momentum, and several pretext tasks are designed for space-time visual correspondence using abundant unlabeled videos.

The key to this task lies in two different perspectives. The first one is **temporal feature learning**, which aims to learn the fine-grained correspondence of pixel/object between frames. With the nature of temporal coherence in the video, the temporal feature learning can be formed as a reconstruction task, where the query pixel in the target frame can be reconstructed by leveraging the information of adjacent reference frames within a local region. Then a reconstruction loss is applied to minimize the photometric error between the raw frame and its reconstruction [1][2]. However, in real videos, the temporal discontinuity occurs frequently due to the occlusions, illumination changes, and deformations, especially for pixels in each frame with severe down-sampling. In such scenarios, the frame reconstruction loss apparently becomes invalid. To alleviate the problem, MAST[1] proposes to apply frame reconstruction with a higher feature resolution by decreasing the stride of the backbone, which requires a larger memory and computation cost. Another way to exploit the free temporal supervision is by applying object-level reconstruction. [3] track object forward and backward with the objective of maximizing the temporal cycle correspondence consistency with reconstruction loss. However, compared to object-level reconstruction, which needs an extra localization module, the frame reconstruction is conducted on raw image space, which provides more

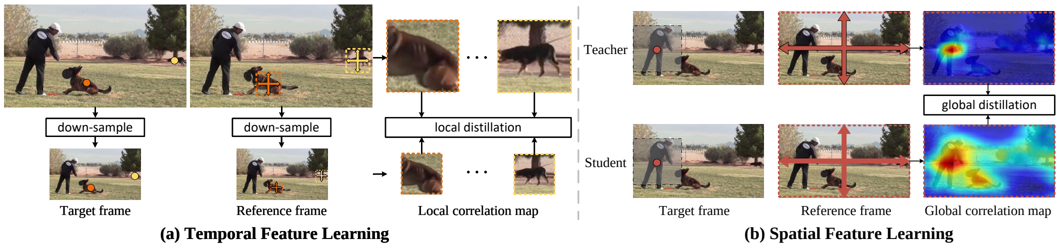

Figure 1: tissor.

accurate supervision for learning fine-grained correspondence.

The second one is **spatial feature learning**, which pays more attention to learning the object appearance that is invariant to viewpoint and deformation changes. [4] adopt a novel intra-inter consistency loss to learn inter-video discriminative feature. [5] learn the space-time correspondence through a frame-wise contrastive loss. However, both methods are trained on video datasets and try to realize the spatial and temporal feature learning in a unified framework, which is sub-optimal for each of them. Recently, the contrastive model pre-trained on image data shows impressive performance for dense representation. This motivates us to design a framework that learns the spatial and temporal feature independently with image and video data.

In this paper, we decouple video correspondence learning into two separate processes, including spatial and temporal feature learning. To achieve this, we first train the model in a contrastive learning paradigm on ImageNet, which gives the model the ability to capture object appearance. Then, instead of training with a large video dataset, i.e., Kinetics[4] with 300k videos, we perform the temporal feature learning on YouTube-VOS, which consists of 3.5k videos. However, apart from the severe information lost and temporal discontinuity due to large spatial down-sampling on frames, directly fine-tuning the old model with only new data will leads to a well-known phenomenon of catastrophic forgetting, which degrades the performance. To address the first problem, we propose a novel pyramid learning framework. The frame reconstruction is applied at different levels of network to better exploit the free temporal supervision. The pixels of target and reference frame with higher resolution have a lower chance of occurring temporal discontinuity, which induce a more accurate local correlation map. Thus we design a new loss named **local correlation distillation loss** that supports explicitly learning of the correlation map at the region with high uncertainty, which is achieved by taking the finest local correlation map as pseudo labels. At the same time, we freeze the model pre-trained on ImageNet as teacher. Then a **global correlation distillation loss** is proposed to keep the student the ability of instance discrimination which is closely related to object appearance modeling.

To sum up, our main contributions include: (a) We proposed a novel decoupled self-supervised video correspondence learning paradigm, including spatial and temporal feature learning. (b) We proposed a pyramid learning framework with local and global distillation loss to enable the model to estimate fine-grained correspondence and capture object appearance. (c) Last but not least, we verify the proposed approach in a series of correspondence-related tasks including video object segmentation, pose tracking, etc. Our approach consistently outperforms previous state-of-the-art self-supervised methods and is even comparable with some task-specific fully-supervised algorithms.

2 Related Work

Object-level correspondence. The goal of visual tracking is to determine a bounding box in each frame based on an annotated box in the reference image. Most methods belong to one of the two categories that use: (a) the tracking-by-detection framework [1, 20, 46, 25], which models tracking as detection applied independently to individual frames; or (b) the tracking-by-matching framework that models cross-frame relations and includes several early attempts, e.g., mean-shift trackers [8, 54], kernelized correlation filters (KCF) [14, 27], and several works that model correlation filters as differentiable blocks [32, 33, 7, 47]. Most of these methods use annotated bounding boxes [52] in every frame of the videos to learn feature representations for tracking. Our work can be viewed as exploiting the tracking-by-matching framework in a self-supervised manner.

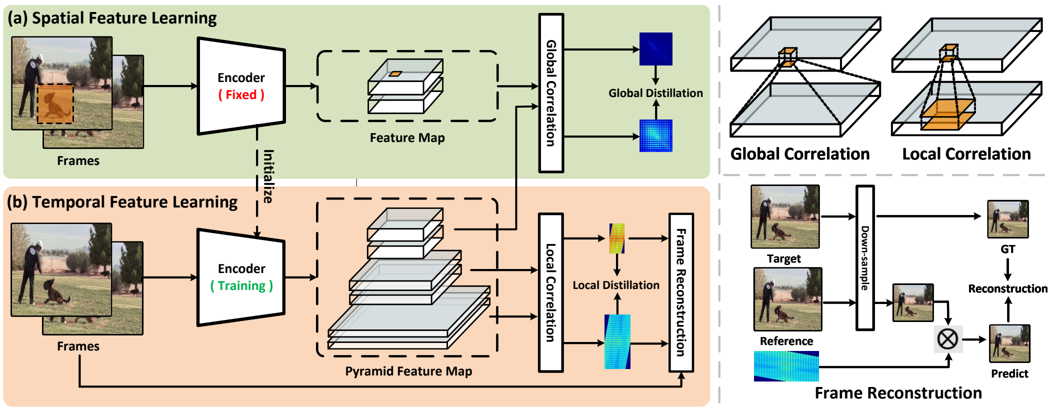

Figure 2: An overview of our spatial temporal feature learning framework. Our method decouples video correspondence learning into two separate processes including spatial feature learning and temporal feature learning. Specifically, the **spatial feature learning** first exploits the contrastive loss which is analogous to that of instance discrimination to learn the object appearance with image data. Then we perform the self-supervised training with video data in the next step. To maintain the ability to capture object appearance, we fix the pre-trained network as teacher and a global correlation distillation is devised. For **temporal feature learning**, we propose a pyramid learning framework where the frame reconstruction is devised at each levels of network. As the same time, we introduce a novel loss named local correlation distillation loss that supports explicitly learning of the correlation map at the region with high uncertainty, which is achieved by taking the finest local correlation map as pseudo labels.

Fine-grained correspondence. Dense correspondence between video frames has been widely applied for optical flow and motion estimation [31, 40, 29, 16], where the goal is to track individual pixels. Most deep neural networks [16, 40] are trained with the objective of regressing the groundtruth optical flow produced by synthetic datasets [4, 10]. In contrast to many classic methods [31, 29] that model dense correspondence as a matching problem, direct regression of pixel offsets has limited capability for frames containing dramatic appearance changes [3, 39], and suffers from problems related to domain shift when applied to real-world scenarios.

Self-supervised learning. Recently, numerous approaches have been developed for correspondence learning via various self-supervised signals, including image [17] or color transformation [44] and cycle-consistency [51, 45]. Self-supervised learning of correspondence in videos has been explored along the two different directions – for region-level localization [51, 45] and for fine-grained pixel level matching [44, 23]. In [45], a correlation filter is learned to track regions via a cycle-consistency constraint, and no pixel-level correspondence is determined. [51] develops patch-level tracking by modeling the similarity transformation of pixels within a fixed rectangular region. Conversely, several methods learn a matching network by transforming color/RGB information between adjacent frames [44, 24, 23]. As no region-level regularization is exploited, these approaches are less effective when color features are less distinctive (see Figure 1(b)). In contrast, our method learns object-level and pixel-level correspondence jointly across video frames in a self-supervised manner.

3 Approach

The basic idea of our method is to decouple video correspondence learning into two separate processes including spatial feature learning and temporal feature learning. We first train our model using contrastive loss with image data to learn the object appearance that is invariant to viewpoint. Then, we perform the temporal feature learning on a small video dataset to learn the fine-grained correspondence between frames. To deal with the problems including temporal discontinuity and catastrophic forgetting when training on video data, a pyramid learning framework is proposed with local and global correlation distillation loss.

3.1 Spatial Feature Learning

The spatial feature mainly describes appearance of objects involved in a image. Spatial feature learning is analogous to that of instance discrimination, and thus easily benefits from the recent advancements brought by contrastive learning. We firstly briefly reviewing instance discrimination objective in contrastive learning. Given an encoded query $\boldsymbol{q} \in \mathbb{R}^d$ and a set of encoded key vectors $\mathcal{K} = \left\{\boldsymbol{k}^+, \boldsymbol{k}_1^-, \boldsymbol{k}_2^-, \ldots, \boldsymbol{k}_K^-\right\}$ which consists of one positive key $\boldsymbol{k}^+ \in \mathbb{R}^d$ and K negative keys $\mathcal{K}^- = \left\{\boldsymbol{k}_j^-\right\}$, where d denotes the embedding dimension. The query and its positive key are generated from same instance with two different augmentations, while the negative keys ref to other instances. The objective of instance discrimination is to maximize the similarity between the query $\boldsymbol{q}$ and the positive key $\boldsymbol{k}^+$ while remains query distinct to all negative keys $\mathcal{K}^-$. Thus, a contrastive loss is presented in InfoNCE with a softmax formulation:

$$\mathcal{L}_{\text{nce}} = -\log \frac{\exp\left(\boldsymbol{q}^T\boldsymbol{k}^+/\tau_c\right)}{\exp\left(\boldsymbol{q}^T\boldsymbol{k}^+/\tau_c\right)+\sum_{i=1}^{K}\exp\left(\boldsymbol{q}^T\boldsymbol{k}_i^-/\tau_c\right)} \quad , \tag{1}$$

where the similarity is measured via dot product, and τ_c is the temperature hyper-parameter. MoCo [14] builds a dynamic memory bank to maintain a large number of negative samples with a moving-averaged encoder. DetCo [15] further introduces global and local contrastive learning to enhance local representation for dense prediction. In this paper, we adopt the same framework of DetCo [15] to learn a appearance model.

Global correlation distillation. After main training with contrastive loss, we get a encoder ϕ. Then we continuously train it on video data to learn the fine-grained correspondence (See section 3.2). However, directly fine-tuning the old model with only new data will lead to a well-known phenomenon of catastrophic forgetting, which degrades the performance. Thus we introduce a global correlation distillation loss in order to maintain the ability to capture object appearance. More specifically, we first fix the feature encoder ϕ as teacher denoted as ϕ_t. Given a pair of video frames consisting of target and reference frame $I_t, I_r \in \mathbb{R}^{H\times W\times 3}$, the ϕ maps them to a pair of feature embeddings $F_t^l, F_r^l \in \mathbb{R}^{h^l\times w^l\times d^l}$, where $l \in \{0, 1, \ldots, N\}$ is the index of each pyramid level and the smaller number represents the coarser level. Here l is set to N. For each query point $F_t^l(i)$ and key point $F_r^l(j)$, we have global correlation $a_{i,j}$ (See Figure 2) using a softmax over similarities $w.r.t.$ all keys in reference frame, $i.e$:

$$a_{i,j} = \frac{\exp\left(F_t^l(i)\cdot F_r^l(j)/\tau\right)}{\sum_n \exp\left(F_t^l(i)\cdot F_r^l(n)/\tau\right)}, i, j, n \in \{1, \ldots, h^l w^l\} \quad , \tag{2}$$

Where '$\cdot$' stands for the dot product. Each point in F_t^l and F_r^l covers a relatively large region since the output stride is set to 32 in our feature encoder. Thus we can form the correlation as object-level correspondence which is closely related to object appearance. We generate the pseudo labels of global correlation distillation by computing the global correlation $a_{i,j}^t$ for each query with teacher ϕ_t. The global correlation distillation loss is defined to minimize the mean squared error between a and a^t:

$$\mathcal{L}_{\text{gc}} = \left\|a - a^t\right\|_2^2 \quad , \tag{3}$$

3.2 Temporal Feature Learning

We then perform temporal feature learning right after spatial feature learning. Temporal feature learning is aiming to learn the fine-grained correspondence between video frames. Recently, a few studies [40, 84] introduce a reconstruction-based correspondence learning scheme, where each query pixel in the target frame can be reconstructed by leveraging the information of adjacent reference frames within a local region. More specifically, the target and reference frame I_t, I_r are projected into a fine-grained pixel embedding space. We denoted these embedding as $F_t, F_r \in \mathbb{R}^{hw\times d}$. For each query pixel i in I_t, we can calculate the local correlation $c_{i,j}$ with reference frame within a local region with a softmax formulation:

$$c_{i,j} = \frac{\exp\left(F_t(i)\cdot F_r(j)/\tau\right)}{\sum_n \exp\left(F_t(i)\cdot F_r(n)/\tau\right)}, i \in \{1, \ldots, hw\}, j, n \in \mathcal{N}(i) \quad , \tag{4}$$

Where $\mathcal{N}(i)$ is the index set with a limited range of r pixels for pixel i (see Figure 2). Then each query pixel i in target frame can be reconstructed by a weighted-sum of pixels in $\mathcal{N}(i)$, according the local correlation map $c \in \mathbb{R}^{hw\times(r)^2}$:

$$\hat{I}_t(i) = \sum_{j\in\mathcal{N}(i)} c_{i,j} I_r(j) \quad , \tag{5}$$

We regard the above process as a transformation function for all query pixels and denotes it as: $\hat{I}_t = \mathcal{T}\left(c, I_t\right)$. Then the reconstruction loss is defined as L1 distance between $\hat{I}_t$ and I_t:

$$\mathcal{L}_{\text{rec}} = \left\| I_t - \hat{I}_t \right\|_1 \ , \tag{6}$$

However, the Eq 5 should only be applied when the feature embedding has same size with video frame. Thus the stride of ϕ must be set to 1, which introduces large memory and computation cost. One possible solution is to apply down-sampling on target frame. MAST propose a image feature alignment module which sampling the pixel at center of strided convolution kernels. However, down-sampling with large rate would not only cause severe information lost of supervision but also result in more pixel occlusions between video frames, which obviously degrades the representation of temporal feature learning. To alleviate the problem, we design a pyramid learning framework consisting of pyramid frame reconstruction and local correlation distillation with entropy-based selection.

Pyramid frame reconstruction. As you can see in Figure 2, we obtain a pair of feature pyramids $\{F_t^l\}_{l=1}^{N-1}$,$\{F_r^l\}_{l=1}^{N-1}$. Then we get the pyramid local corelation map $\{c^l\}_{l=1}^{N-1}$ at each level by utilizing Eq 4 with different local range r^l. As the same time, we adopt a same down-sampling method as [1] to obtain a pair of frame pyramids $\{I_t^l\}_{l=1}^{N-1}$, $\{I_r^l\}_{l=1}^{N-1}$, which has same shape with the feature pyramids at each level. Given the c^l, I_t^l and I_r^l, we apply the pyramid reconstruction loss at each level:

$$\mathcal{L}_{\text{rec}}^p = \sum_l \left\| I_t^l - \mathcal{T}\left(c^l, I_r^l\right) \right\|_1 \ , \tag{7}$$

By doing this, we are able to exploit more free temporal supervision and get better temporal representation at intermediate level.

Local correlation distillation. The bottom level of the frame pyramid contains more rich information and serfer less occlusions for temporal feature learning due to relatively small down-sampling rate, which may result in more accurate local correlation map. Inspired by it, we design a novel local correlation distillation loss which explicitly make constraint on the final local correlation map $c^{N-1} \in \mathbb{R}^{h^{N-1}w^{N-1} \times (r^{N-1})^2}$. We first compute the local correlation map c^{N-2} at level $N-2$ and then apply correlation down-sampling [16] to get pseudo labels c^t with the same size as c^{N-1}. Then the local correlation distillation loss $\mathcal{L}_{lc}$ is adopt to minimize the mean squared error between c^{N-1} and c^t.

Entropy-based selection. The correlation of each query $w.r.t$ reference frame indicates more uncertainty when having smooth distribution, which should be paid more attention when applying distillation. Thus we calculate the entropy for each query i:

$$\mathcal{H}(i) = \sum_j -log c_{i,j}^{N-1} \ , \tag{8}$$

Then we obtain a mask $m \in \mathbb{R}^{h^{N-1}w^{N-1}}$ to filter out the region with lower entropy by setting a threshold T. The local correlation distillation loss with entropy selection is defined as:

$$\mathcal{L}_{\text{lc}}^e = \sum_i m_i \left\| c_{i,:}^{N-1} - c_{i,:}^t \right\|_2^2 \ , \tag{9}$$

Eventually, our training loss of temporal feature learning is defined as: $\mathcal{L}_{\text{t}}$ = $\mathcal{L}_{\text{rec}}^p + \alpha\mathcal{L}_{\text{lc}}$. The final loss is a weighted sum of $\mathcal{L}_{\text{t}}$ and a regularization term $\mathcal{L}_{\text{gc}}$ introduced in Section 3.1:

$$\mathcal{L} = \mathcal{L}_{\text{t}} + \beta\mathcal{L}_{\text{gc}} \ , \tag{10}$$

4 Experiments

We verify the merit of our method in a series of correspondence-related tasks, including semi-supervised video object segmentation, pose keypoints tracking, human parts segmentation propagation. In this section we will first introduce our experiments settings including details of implementation and evaluation. Then detailed ablation studies are performed to explain how each component of our method works. Last but not least, we finally report the performance comparision with state-of-the-art methods to further verify the effectiveness of our method.

#	$\mathcal{L}_{\text{nce}}$	$\mathcal{L}_{\text{rec}}$	$\mathcal{L}^{p}_{\text{rec}}$	$\mathcal{L}_{\text{lc}}$	$\mathcal{L}^{e}_{\text{lc}}$	$\mathcal{L}_{\text{gc}}$	Dataset	Arch	$\mathcal{J}\&\mathcal{F}$(Mean) ↑
1	✓						I	Res18	61.7
2		✓					YT	Res18	64.7
3		✓	✓				YT	Res18	65.4
4		✓	✓	✓			YT	Res18	68.1
5		✓	✓	✓	✓		YT	Res18	69.1
6	✓	✓	✓	✓	✓		I + YT	Res18	69.4
7	✓	✓	✓	✓	✓	✓	I + YT	Res18	**70.4**

(a) Ablation study of each component.

Method	Dataset	Arch	$\mathcal{J}\&\mathcal{F}$(Mean) ↑
DetCo [12]	I	Res50	66.5
+ Fine-tuning	I + YTV	Res50	68.3
+ EWC	I + YTV	Res50	67.9
+ LwF	I + YTV	Res50	68.6
+ $\mathcal{L}_{\text{gc}}$ (Ours)	I + YTV	Res50	**71.1**

(b) Ablation study of L_{gc}.

Table 5: Ablation study for each component in spatial and temporal feature learning. The "+" in (b) represents the method used in additional to $\mathcal{L}_{\text{t}}$ after trained with contrastive loss. I: ImageNet [11]. YTV: YouTube-VOS [12].

4.1 Implementation Details

Architectures. We exploit the encoder ϕ with both ResNet-18 and ResNet-50 [27, 37] for self-supervised training. Following prior works [12], we reduce the stride of convolutional layers in ϕ to increase the spatial resolution of feature maps on layer $res4$ by a factor of 4 or 8 ($i.e$, downsampling rate 8 or 4).

Training. We first train our model for 200 epochs on ImageNet [11] adopting the same hyperparameter as DetCo [15]. Then we perform temporal feature learning on YouTube-VOS train set [16] which consists of 3.5k videos. In this stage, the video frame is resized into 256×256 and channel-wise dropout in Lab color space [39] is adopted as the information bottleneck. We train the encoder for 100k iterations with mini-batch of 128 using Adam as our optimizer. The initial learning rate is set to 1e-4 with a cosine (half-period) learning rate schedule. The frame reconstruction is applied on both $res3$ and $res4$ layer while we realize global correlation distillation on $res5$ layer.

Evaluation. We follow the same evaluation protocol and downstream tasks in [33, 71]. We directly utilize unsupervised pre-trained model as the feature extractor without any fine-tuning. Given the input frame with spatial resolution of $H \times W$, the evaluation is realized on the $res4$ layer with a spatial resolution of $\frac{H}{8} \times \frac{W}{8}$ or $\frac{H}{4} \times \frac{W}{4}$. To propagate the semantic labels from the initial ground-truth annotation, the recurrent inference strategy is applied following recent works [12]. More specifically, the semantical label of first frame as well as previous predictions are propagated to the current frame with the help of affinity between video frames. We evaluate our method over three downstream tasks including semi-supervised video object segmentation in DAVIS-2017 [8], human part tracking in VIP [10] and human pose tracking in JHMDB [9].

4.2 Ablation Study

The ablation study is performed with semi-supervised video object segmentation on DAVIS-2017 validation set. We conduct a series of experiments to prove the effectiveness of each component.

Temporal feature learning. We first examine how each design in temporal feature learning impacts the overall performance, which are shown in Table 5 (a) $2 \sim 5$. To have a clear look, we train model from scratch on YouTube-VOS when examine the efficacy of our components for temporal feature learning. The baseline is to apply frame reconstruction $\mathcal{L}_{\text{rec}}$ on layer $res4$. The $\mathcal{L}^{p}_{\text{rec}}$, $\mathcal{L}_{\text{lc}}$ and $\mathcal{L}^{e}_{\text{lc}}$ represents pyramid frame reconstruction, local correlation distillation without or with entropy-based selection. From the table we can see $\mathcal{L}^{p}_{\text{rec}}$ leads to a improvements by leveraging more supervision of reconstruction at each level. With the guidance of more fine-grained local correlation map, $\mathcal{L}_{\text{lc}}$ boosts up the accuracy from 65.4% to 68.1%. Moreover, enforcing the local correlation distillation to focus on the region with higher entropy leads to a performance gain in the range of 1%. By fusing the above components, the performance of our method finally reaches 69.1%.

Spatial featuer learning. We investigate the effect of training with each component in spatial feature learning. The results are shown in Table 3 (a) $6 \sim 7$, with help of the pre-training on ImageNet using contrastive loss, the performance of our method reaches 69.4%. Moreover, the global correlation distillation loss $\mathcal{L}_{\text{gc}}$ boosts up the performance from 69.4% to 70.4% by keeping the ability of model to capture object-level correspondence which is closely related to object appearance modeling.

Further exploitation of $\mathcal{L}_{\text{gc}}$. As we mentioned above, directly finetuning the model pre-trained on image data will leads to a well-known phenomenon of catastrophic forgetting, which is closely related to continue learning. To further verify the effectiveness of $\mathcal{L}_{\text{gc}}$, we exploit several classic methods of continue learning including EWC [7] and LwF [23] apart from directly finetuning on video dataset. We adopt ResNet50 as our encoder which has a more stronger ability to capture object appearance

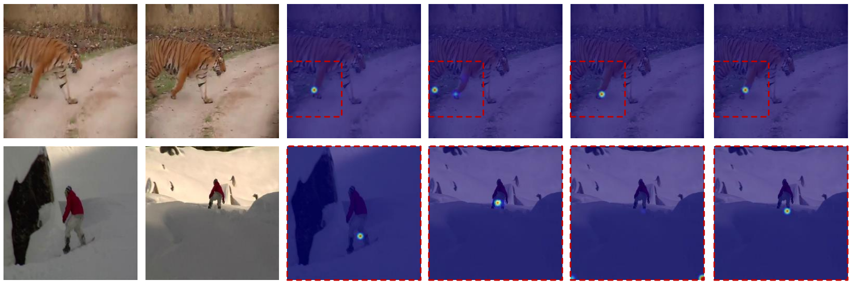
Target frame
Reference frame
Query
With $\mathcal{L}_{\mathrm{nce}}$
With $\mathcal{L}_{\mathrm{t}}$
With all

Figure 3: Visualization of the ablation study. Given a query point randomly sampled in target frame, we visualize the result of computing the local correlation and global correlation map $w.r.t.$ reference frame. Here "$\mathcal{L}_{\mathrm{nce}}$", "$\mathcal{L}_{\mathrm{t}}$" and "all" corresponds to contrastive loss of spatial feature learning, the loss of temporal feature learning and our full model.

and train model with all modules in temporal feature learning. The results are shown on Table 5 (b). All methods achieve better result attributed to the temporal feature learning on video data while our method gets the best performance. Interestingly, we found integrate EWC into our framework gets worse result compared fine-tuning with temporal feature learning. We speculate that this may be the result of inferior learning of temporal representation due to the influence of the regularization loss in EWC.

Discussion. We give a further discussion here base on the above analysis. On the one hand, the temporal feature learning helps to learn the fine-grained correspondence related to motion estimation between frames within a local range, which is unable to accomplish by training a appearance model. As you can see in the first row of Figure 3, the appearance model trained with $\mathcal{L}_{\mathrm{nce}}$ is misled by two patches at different location ($i.e.$ two feats of the tiger) with similar appearance while the model trained with $\mathcal{L}_{\mathrm{t}}$ tends to learn a better temporal representation for fine-grained correspondence. However, in the second row of Figure 3, the video contains large motion and deformation especially on the moving objects, which result in severe temporal discontinuity between frames. In such scenarios, the model trained with $\mathcal{L}_{\mathrm{t}}$ fails to capture temporal correspondence with a local correlation while the model trained with $\mathcal{L}_{\mathrm{nce}}$ is able to correct the mistakes by tracking the points base on the object appearance (see with $\mathcal{L}_{\mathrm{nce}}$ and with all). Besides, the performance comparison of ResNet18 and ResNet50 in Table 5, $i.e.$ 70.4% and 71.1%, indicates that our method is able to scale up to deeper models which is not achieved in the most of previous methods reported in [12].

4.3 Comparison with State-of-the-art

We compare fine-grained correspondence results of VFS against previous self-supervised methods. The results are all reported with the last block in res4 across all methods. Our method achieves state-of-the-art performance using ResNet-50. With ResNet-50, we observes UVC [42] does not benefit from using a deeper networks and the performance of CRW [33] becomes significantly worse. Learning with VFS, the deeper network with ResNet-50 improves 2.2% on DAVIS and 3.3% on VIP over ResNet-18, which is significant. We observe consistent results across the JHMDB [35] human pose and VIP [82] human part tracking tasks. With ResNet-18, VFS achieves comparable performance with CRW [33]. the last block in res4 may not achieve the optimal performance, thus we also report the result of the best block with gray color for reference. We compare fine-grained correspondence results of VFS against previous self-supervised methods. The results are all reported with the last block in res4 across all methods. Our method achieves state-of-the-art performance using ResNet-50. With ResNet-50, we observes UVC [42] does not benefit from using a deeper networks and the performance of CRW [33] becomes significantly worse. Learning with VFS, the deeper network with ResNet-50 improves 2.2% on DAVIS and 3.3% on VIP over ResNet-18, which is significant. We observe consistent results across the JHMDB [35] human pose and VIP [82] human part tracking tasks. With ResNet-18, VFS achieves comparable performance with CRW [33]. the last block in res4 may not achieve the optimal performance, thus we also report the result of the best block with gray color for reference. We compare fine-grained correspondence results of VFS against previous self-supervised methods. The results are all reported with the last block in res4 across all methods. Our method achieves state-of-the-art performance using ResNet-50. With ResNet-50, we observes UVC [42] does not benefit from using a deeper networks and the performance of CRW [33]

Method	Arch	Stride	Dataset (size)	$\mathcal{J}$ & $\mathcal{F}$(Mean) ↑	$\mathcal{J}$(Mean) ↑	$\mathcal{J}$(Recall) ↑	$\mathcal{F}$(Mean) ↑	$\mathcal{F}$(Recall) ↑
Supervised	ResNet-18	8	I (1.28M, -)	62.9	60.6	-	65.2	-
MoCo	ResNet-18	8	I (1.28M, -)	60.8	58.6	-	63.1	-
DetCo	ResNet-18	8	I (1.28M, -)	61.7	59.6	-	62.8	-
SimSiam	ResNet-18	8	I (1.28M, -)	62.0	60.0	-	64.0	-
Colorization	ResNet-18	8	Kinetics (- , 800 hours)	34.0	34.6	34.1	32.7	26.8
CorrFlow	ResNet-18	8	OxUvA (- , 14 hours)	50.3	48.4	53.2	52.2	56.0
MuG	ResNet-18	8	OxUvA (- , 14 hours)	54.3	52.6	57.4	56.1	58.1
CRW	ResNet-18	8	Kinetics (- , 800 hours)	68.3	65.5	78.6	71.0	82.9
ContrastCorr	ResNet-18	8	C + T (30k, 300 hours)	63.0	60.5	-	65.5	-
VFS	ResNet-18	8	Kinetics (- , 800 hours)	66.7	64.0	-	69.4	-
JSTG	ResNet-18	8	Kinetics (- , 800 hours)	68.7	65.8	77.7	71.6	84.3
MAST	ResNet-18	4	YTV (- , 5 hours)	65.5	63.3	73.2	67.6	77.7
MAMP	ResNet-18	4	YTV (- , 5 hours)	69.7	68.3	-	71.2	-
Ours	ResNet-18	8	YTV (- , 5 hours)	69.1	**-**	**-**	-	**-**
Ours	ResNet-18	8	I + YTV (1.28M, 5 hours)	70.4	-	-	-	-
Ours	ResNet-18	4	YTV (- , 5 hours)	71.3	-	-	-	-
Ours	ResNet-18	4	I + YTV (1.28M, 5 hours)	**73.2**	-	-	-	-
Supervised	ResNet-50	8	I (1.28M, -)	66.0	63.7	-	68.4	-
MoCo	ResNet-50	8	I (1.28M, -)	65.4	63.2	-	67.6	-
SimSiam	ResNet-50	8	I (1.28M, -)	66.3	64.5	-	68.2	-
DetCo	ResNet-50	8	I (1.28M, -)	66.5	64.7	-	68.4	-
VINCE	ResNet-50	8	Kinetics (- , 800 hours)	65.6	63.4	-	67.8	-
SeCo	ResNet-50	8	Kinetics (- , 800 hours)	60.6	60.4	-	62.8	-
VFS	ResNet-50	8	Kinetics (- , 800 hours)	68.9	-	-	-	-
Ours	ResNet-50	8	I + YTV (1.28M, 5 hours)	**71.1**	-	-	-	-

Table 2: **Quantitative results for video object segmentation**. I: ImageNet. YTV: YouTube-VOS. C: COCO. T: TrackingNet.

becomes significantly worse. Learning with VFS, the deeper network with ResNet-50 improves 2.2% on DAVIS and 3.3% on VIP over ResNet-18, which is significant. We observe consistent results across the JHMDB [35] human pose and VIP [82] human part tracking tasks. With ResNet-18, VFS achieves comparable performance with CRW [33]. the last block in res4 may not achieve the optimal performance, thus we also report the result of the best block with gray color for reference.

5 Conclusions

References

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes]
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? **[TODO]**
   (b) Did you describe the limitations of your work? **[TODO]**
   (c) Did you discuss any potential negative societal impacts of your work? **[TODO]**

    (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? **[TODO]**

2. If you are including theoretical results...
    (a) Did you state the full set of assumptions of all theoretical results? **[TODO]**
    (b) Did you include complete proofs of all theoretical results? **[TODO]**
3. If you ran experiments...
    (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? **[TODO]**
    (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? **[TODO]**
    (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? **[TODO]**
    (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? **[TODO]**
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
    (a) If your work uses existing assets, did you cite the creators? **[TODO]**
    (b) Did you mention the license of the assets? **[TODO]**
    (c) Did you include any new assets either in the supplemental material or as a URL? **[TODO]**
    (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? **[TODO]**
    (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? **[TODO]**
5. If you used crowdsourcing or conducted research with human subjects...
    (a) Did you include the full text of instructions given to participants and screenshots, if applicable? **[TODO]**
    (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? **[TODO]**
    (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? **[TODO]**

A Appendix

Optionally include extra information (complete proofs, additional experiments and plots) in the appendix. This section will often be part of the supplemental material.