Spatial-then-Temporal Self-Supervised Learning for Video Correspondence

Anonymous Author(s)
Affiliation
Address
email

Abstract

Learning temporal correspondence from unlabeled videos is of vital importance in computer vision, and has been tackled by different kinds of self-supervised pretext tasks. For the self-supervised learning, recent studies suggest using large-scale video datasets despite the training cost. We propose a spatial-then-temporal pretext task to address the training data cost problem. The task consists of two steps. First, we use contrastive learning from unlabeled still image data to obtain appearance sensitive features. Then we switch to unlabeled video data and learn motion sensitive features by reconstructing frames. In the second step, we propose a global correlation distillation loss to retain the appearance sensitivity learned in the first step, as well as a local correlation distillation loss in a pyramid structure to combat temporal discontinuity. Experimental results demonstrate that our method surpasses the state-of-the-art self-supervised methods on a series of correspondence-based tasks. The conducted ablation studies verify the effectiveness of the proposed method.

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 [7][14], video object segmentation [2][31], keypoint tracking [46], 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 large scale video datasets.

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 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 with a local range. Then a reconstruction loss is applied to minimize the photometric error between the raw frame and its reconstruction. However, the temporal discontinuity occurs frequently due to the occlusions, dramatic appearance changes, and deformations, especially for pixels in each frame with large down-sampling. In such scenarios, the frame reconstruction loss apparently becomes invalid, which results in inferior performance. To alleviate the problem, MAST [20] applies 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 free temporal supervision is by exploiting temporal cycle-consistency. [15][41] track objects forward and backward with the objective of maximizing the cycle-consistency using

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

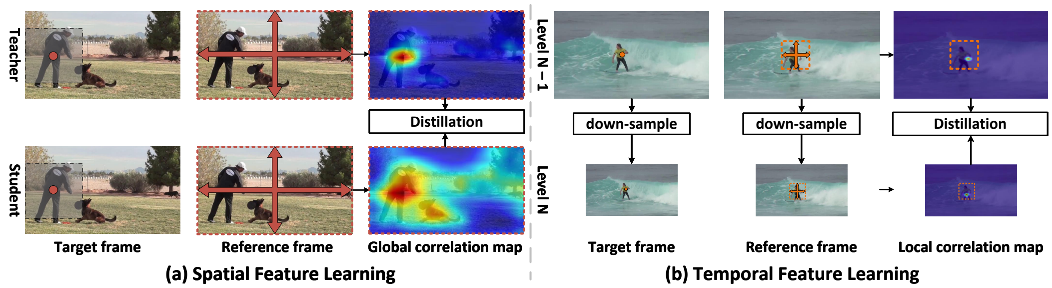

Figure 1: **Illustration of the main idea.** In (a), we first train a contrastive model on still image data and fixed it as teacher. Then the distillation on global correlation maps is proposed to retain the ability to capture object appearance. In (b), the distillation is performed between local correlation maps at different pyramid levels to facilitate fine-grained matching. The local correlation map computed at lower pyramid level is regarded as pseudo labels.

reconstruction and contrastive loss. However, compared to the correspondence learning realized at object-level, the frame reconstruction is conducted on raw image space, which provides more 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 appearance changes. [40] adopts a novel intra-inter consistency loss to learn discriminative spatial feature while [47] learns the space-time correspondence through a frame-wise contrastive loss. 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, as mentioned in [43], the contrastive model [13][45] pre-trained on image data shows competitive performance against dedicated methods for video correspondence due to its superior capability of learning spatial representation. However, such a model still fails to realize the fine-grained matching between video frames. This motivates us to design a framework that learns the spatial and temporal features independently with image and video data.

In this paper, we propose a spatial-then-temporal pretext task, which decouples self-supervised video correspondence learning into two separate steps, including spatial and temporal feature learning. To achieve this, we first train the model in a contrastive learning paradigm on ImageNet [6], which gives the model the ability to capture object appearance. Then, instead of training with a large-scale video dataset, $i.e.$, Kinetics [4] with 300k videos, we perform the temporal feature learning on YouTube-VOS [48], which consists of 3.5k videos. However, apart from the severe information loss and temporal discontinuity due to 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 [22]. To address the first problem, we propose a novel pyramid learning framework. First, the frame reconstruction is applied at different levels of the network to better exploit the free temporal supervision. As observed in Figure 1 (a), the pixels of the target and reference frame with higher resolution have a lower chance of occurring temporal discontinuity, which provides 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 in the region with high uncertainty, which is achieved by taking the finest local correlation map as pseudo labels. At the same time, as shown in Figure 1 (b), we freeze the model pre-trained on ImageNet as teacher. Then a global correlation distillation loss is proposed to retain the appearance sensitive features.

To sum up, our main contributions include: (a) We propose a novel spatial-then-temporal pretext task for self-supervised video correspondence, which addresses the training data cost problem by learning spatial and temporal features sequentially. (b) We propose a pyramid learning framework with local correlation distillation to combat the temporal discontinuity of frame reconstruction. (c) We introduce a global correlation distillation loss to retain the appearance sensitivity learned in the first step when training on a video dataset. (d) We verify our approach in a series of correspondence-related tasks, including video object segmentation, human parts propagation, and pose tracking. 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

Self-supervised learning for video correspondence. Recent approaches focus on learning correspondence from unlabeled videos in a self-supervised manner. The task requires the model to have the ability to capture object appearance and estimate the fine-grained correspondence between frames at the same time, which has proceeded along two different dimensions: reconstruction-based methods [19][20][21][39][40] and cycle-consistency-based methods [15][41][52]. In the first type, a query point is reconstructed from adjacent frames while the latter performs forward-backward tracking with the objective of minimizing the cycle inconsistency. Through getting promising results, most methods address the problem by considering only one perspective. VFS [47] learns the spatial and temporal representation through a frame-wise contrastive loss while [1][40] try to realize the spatial and temporal feature learning in a unified framework by exploiting the inter-video constraint, which may result in sub-optimal performance. In this paper, we learn a better representation by proposing a spatial-then-temporal pretext task, which performs spatial and temporal feature learning sequentially.

Self-supervised spatial feature learning. Self-supervised spatial feature learning aims to learn discriminative features of object appearance with unlabeled data, which recently got promising result with contrastive learning. In an early work [44], the contrastive learning is formulated as an instance discrimination task, which requires the model to return low values for similar pairs and high values for dissimilar pairs. Recently, the performance is further improved by creating a dynamic memory-bank [13], introducing online clustering [3] and avoiding the use of negative pairs [5][11]. Furthermore, [42][45][49] propose various pretext tasks to adapt the contrastive learning to dense prediction tasks. Even though showing superior performance for temporal correspondence [43], the contrastive model pre-trained on image data still struggles to model the motion between video frames.

Self-supervised temporal feature learning. Compared to spatial feature learning, temporal feature learning focus on learning the motion information of video, which is closely related to optical flow and motion estimation. Most methods [7][34] directly regress the ground-truth optical flow produced by synthetic datasets, thus suffering from severe domain shift. To deal with the problem, [28] tries to learn the dense correspondence on real video without any label by minimizing the photometric error between the raw frame and its reconstruction in the valid region. However, the video frames usually contain temporal discontinuity including dramatic appearance changes and occlusions, which seriously degrades the capability of the method. [18][24][25] alleviate the problem by utilizing the optical flow predictions from teacher model to guide the learning of student model in the region with occlusions. In this paper, we address the issue by proposing a local correlation distillation loss in a pyramid structure.

3 Approach

The basic idea of our method is to decouple video correspondence learning into two sequential steps, including spatial and temporal feature learning. We first train our model using contrastive loss with still image data to learn appearance sensitive features. Then, we perform the temporal feature learning on a small video dataset to learn the fine-grained correspondence between frames. In the second step, we propose a global correlation distillation loss to retain the ability to capture object appearance while address the problem of temporal discontinuity by introducing a local correlation distillation loss with a pyramid learning framework.

3.1 Spatial Feature Learning

The spatial feature mainly describes the appearance of objects involved in an image. Spatial feature learning is analogous to that of instance discrimination and thus easily benefits from the recent advancements brought by contrastive learning. We first briefly review the 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 the same instance with two different augmentations, while the negative keys refer 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 remaining query distinct to all negative keys $\mathcal{K}^-$. Thus, a

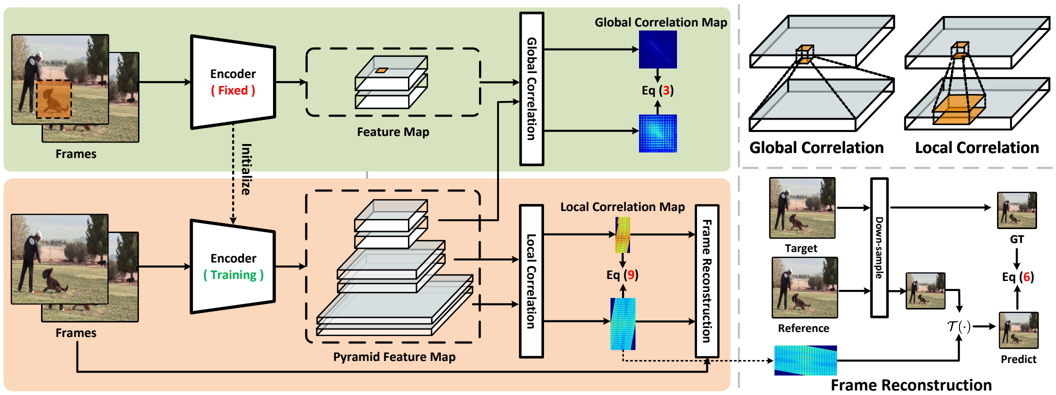

Figure 2: **Overview of the second step in our pretext task. The fixed encoder was trained in the first step (not shown).** We first exploits the contrastive loss to learn the appearance sensitive features with still image data (not shown). Then we perform the temporal feature learning with video data in the second step. To retain the appearance sensitivity, we fix the pre-trained network as teacher and a global correlation distillation loss is devised between global correlation maps. To address the issue of temporal discontinuity, we first apply frame reconstruction at each pyramid level of the network. Then the distillation is conducted between local correlation maps computed at different pyramid levels, which learns better motion sensitive features by taking fine-grained local correlation maps as pseudo labels.

contrastive loss is presented in InfoNCE [36] with a softmax formulation:

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

where the similarity is measured via dot product, and τ_c is the temperature hyper-parameter. MoCo [13] builds a dynamic memory bank to maintain a large number of negative samples with a moving-averaged encoder. DetCo [45] further improves the contrastive loss $\mathcal{L}_{\text{nce}}$ by introducing a global and local contrastive learning to enhance local representation for dense prediction. In this paper, we adopt the same framework as [13][45] to learn an appearance model for most of our experiments.

Global correlation distillation. After main training with contrastive loss, we get an 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 [22], 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, the ϕ maps them to a pair of feature embeddings $F_t^l, F_r^l \in \mathbb{R}^{h^l w^l \times d^l}$, where $l \in \{0, 1, \ldots, N\}$ is the index of each pyramid level and the smaller number represents the coarser pyramid level. Here l is set to N. For each query point $F_t^l(i)$ and key point $F_r^l(j)$, we compute the global correlation $a_{i,j}$ using a softmax over similarities $w.r.t.$ all keys in the reference frame (see the upper right of Figure 2), $i.e$:

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

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 $\mathcal{L}_{\text{gc}}$ is defined to minimize the mean squared error between a and a^t.

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

3.2 Temporal Feature Learning

We then perform temporal feature learning right after spatial feature learning. Temporal feature learning aims to learn the fine-grained correspondence between video frames. Recently, a few studies [20][39] 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 with a limited range. 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 $c_{i,j}$ $w.r.t$ the reference frame in a local range (see the upper right of Figure 2):

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

Where $\mathcal{N}(i)$ is the index set with a limited range of r pixels for pixel i. 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 $c \in \mathbb{R}^{hw\times(r)^2}$:

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

We regard the above process as a transformation function for all query pixels and denotes it as: $\hat{I}_t = \mathcal{T}\left(c, I_r\right)$. Then the reconstruction loss $\mathcal{L}_{\text{rec}}$ 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 the same size as 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 the target and reference frame. MAST [20] proposes an image feature alignment module that samples the pixel at the center of strided convolution kernels. However, down-sampling with a large rate would cause severe information loss and result in more pixel occlusions between video frames, which obviously degrades the representation of temporal feature learning. To address the issue, we design a pyramid learning framework consisting of pyramid frame reconstruction and local correlation distillation with entropy-based selection.

Pyramid frame reconstruction. As observed 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 pyramid level by utilizing Eq 4 with different range r^l. As the same time, we adopt a same down-sampling method as [20] to get 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 pyramid level. Given the c^l, I_t^l and I_r^l, we apply the pyramid reconstruction loss:

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

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

Local correlation distillation. The bottom level of the frame pyramid contains rich information and suffer 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 [34] 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 to 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: $\mathcal{L}_{\mathrm{t}} = \mathcal{L}^{p}_{\mathrm{rec}} + \alpha\mathcal{L}^{e}_{\mathrm{lc}}$. The final loss of training on video data is a weighted sum of $\mathcal{L}_{\mathrm{t}}$ and a regularization term $\mathcal{L}_{\mathrm{gc}}$ introduced in Section 3.1:

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

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, and human parts segmentation propagation. This section will first introduce our experiment settings, including implementation and evaluation details. Then detailed ablation studies are performed to explain how each component of our method works. Last but not least, we finally report the performance comparison with state-of-the-art methods to further verify the effectiveness of our method.

4.1 Implementation Details

Architectures. We exploit the encoder ϕ with both ResNet-18 and ResNet-50 [12] for self-supervised training. Following prior works [15][20][47], we reduce the stride of convolutional layers in ϕ to increase the spatial resolution of feature maps on layer res_4 by a factor of 4 or 8 ($i.e$, downsampling rate 8 or 4).

Training. We first train our model using contrastive loss for 200 epochs on ImageNet [6] following most hyper-parameters settings of [13]. Then we perform temporal feature learning on YouTube-VOS [48] training set 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 [19][20] is adopted as the information bottleneck. We train the encoder for 90k/45k iterations with a mini-batch of 128/64 for ResNet-18/ResNet-50, 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 res_3 and res_4 layer while we realize global correlation distillation on res_5 layer.

Evaluation. We directly utilize the 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 res_4 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 [15][20][47]. More specifically, the semantical label of the 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 [32], human part propagation in VIP [53], and human pose tracking in JHMDB [17].

4.2 Ablation Study

The ablation study is performed with semi-supervised video object segmentation [32] on DAVIS-2017 validation set. Following the official protocol [32], we use the mean of region similarity $\mathcal{J}_m$, mean of contour accuracy $\mathcal{F}_m$ and their average $\mathcal{J}\&\mathcal{F}_m$ as the evaluation metrics. We conduct a series of experiments to prove the effectiveness of each component. The stride of the encoder is all set to 8 for training and evaluation.

Temporal feature learning. We first examine how each design in temporal feature learning impacts the overall performance, which is shown in Table 1 (a). To have a clear look, we train the model from scratch on YouTube-VOS [48] when examining the efficacy of our components for temporal feature learning. The baseline is to apply frame reconstruction $\mathcal{L}_{\mathrm{rec}}$. The p, $\mathcal{L}_{\mathrm{lc}}$ and e represents pyramid frame reconstruction, local correlation distillation without and with entropy-based selection. From the table, we can see leveraging more supervision of frame reconstruction at each pyramid level leads to an improvement in the range of 0.8%. With the guidance of a more fine-grained local correlation map, $\mathcal{L}_{\mathrm{lc}}$ 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 0.9%. By fusing the above components, the performance finally reaches 69.0%.

Spatial feature learning. We investigate the effect of training with each component in spatial feature learning. The results are shown in the last two rows of Table 1 (a). With the help of the pre-training

$\mathcal{L}_{\rm nce}$	$\mathcal{L}_{\rm gc}$	$\mathcal{L}_{\rm rec}$	p	$\mathcal{L}_{\rm lc}$	e	Dataset	Backbone	$\mathcal{J}\&\mathcal{F}_m \uparrow$
		✓				YTV	Res18	64.6
		✓	✓			YTV	Res18	65.4
		✓	✓	✓		YTV	Res18	68.1
		✓	✓	✓	✓	YTV	Res18	69.0
✓		✓	✓	✓	✓	I + YTV	Res18	69.3
✓	✓	✓	✓	✓	✓	I + YTV	Res18	**70.5**

(a) Ablation study of each component.

Method	Dataset	Backbone	$\mathcal{J}\&\mathcal{F}_m \uparrow$
$\mathcal{L}_{\rm nce}$	I	Res50	66.5
w $\mathcal{L}_{\rm t}$	I + YTV	Res50	69.6
w $\mathcal{L}_{\rm t}$ + LwF [22]	I + YTV	Res50	69.9
w $\mathcal{L}_{\rm t}$ + $\mathcal{L}_{\rm gc}$	I + YTV	Res50	**71.3**

(b) Ablation study of $\mathcal{L}_{\rm gc}$.

Table 1: **Ablation study for each component in our framework**. The "p" and "e" in (a) correspond to pyramid frame reconstruction and entropy-based selection. Models in (b) are all pre-trained on ImageNet with contrastive loss and models with "w" are subsequently trained on YouTube-VOS using different methods. I: ImageNet [6]. YTV: YouTube-VOS [48].

Target frame | Reference frame | Query point | With $\mathcal{L}_{\rm nce}$ | With $\mathcal{L}_{\rm t}$ | With all

Figure 3: **Visualization of the ablation study.** Given a query point randomly sampled in the target frame, we visualize the result of computing the local correlation and global correlation map w.r.t. reference frame. The dashed line in red represents the range of computing correlation map w.r.t. query point. The reference frame is randomly sampled in the memory bank of inference strategy [15][20][47].

on ImageNet using contrastive loss, the performance of our method reaches 69.3%. Moreover, the global correlation distillation loss $\mathcal{L}_{\rm gc}$ boosts up the performance from 69.3% to 70.5% by keeping the ability of the model to capture object-level correspondence, which is closely related to object appearance modeling.

Further exploitation of $\mathcal{L}_{\rm gc}$. Directly fine-tuning the model pre-trained with contrastive loss $\mathcal{L}_{\rm nce}$ will lead to a well-known phenomenon of catastrophic forgetting [22], which is closely related to continual learning. To further verify the effectiveness of $\mathcal{L}_{\rm gc}$, we exploit a general continual model LwF [22] based on knowledge distillation apart from directly fine-tuning on video dataset. We modify the framework of LwF to adapt to the paradigm of self-supervised learning and adopt the framework of [45] with ResNet-50 when training at first step. The results are shown on Table 1 (b). All methods achieve better results attributed to the proposed temporal feature learning, while our method using $\mathcal{L}_{\rm gc}$ gets the best performance.

Further analysis. We give a further analysis here based on the above experiments. On the one hand, temporal feature learning helps to learn the fine-grained correspondence related to motion estimation between frames, which is unable to accomplish by training an appearance model. As you can see in the first row of Figure 3, the appearance model trained with $\mathcal{L}_{\rm nce}$ is misled by two patches at different locations ($i.e.$ two feats of the tiger) which has a similar appearance, while the model trained with $\mathcal{L}_{\rm t}$ tends to learn a better temporal representation for fine-grained correspondence. However, in the second row of Figure 3, the model trained with $\mathcal{L}_{\rm t}$ fails to capture temporal correspondence with a local correlation when facing severe temporal discontinuity while the model trained with $\mathcal{L}_{\rm nce}$ is able to correct the mistakes by tracking the points based on the object appearance (see with $\mathcal{L}_{\rm nce}$ and with all).

4.3 Comparison with State-of-the-art

Results for video object segmentation. We compare our method against previous self-supervised methods in Table 2. For a fair comparison, we report both results by setting the stride of the encoder to 4 and 8. The results are all reported with layer res_4 across all methods. Our method achieves state-of-the-art performance using both ResNet-18 and ResNet-50. For ResNet-18, our method with a stride of 8 achieves 70.5%, making an absolute performance improvement by 1.2% over all baselines

Method	Sup.	Backbone	Stride	Training Dataset		$\mathcal{J}\&\mathcal{F}_m \uparrow$	$\mathcal{J}_m \uparrow$	$\mathcal{F}_m \uparrow$
				Image	Video			
MoCo [13]		ResNet-18	8	ImageNet	-	60.8	58.6	63.1
SimSiam [5]		ResNet-18	8	ImageNet	-	62.0	60.0	64.0
Colorization [39]		ResNet-18	8	-	Kinetics	34.0	34.6	32.7
CorrFlow [19]		ResNet-18	8	-	OxUvA	50.3	48.4	52.2
MuG [26]		ResNet-18	8	-	OxUvA	54.3	52.6	56.1
UVC [21]		ResNet-18	8	COCO	Kinetics	59.5	57.7	61.3
ContrastCorr [40]		ResNet-18	8	COCO	TrackingNet	63.0	60.5	65.5
VFS [47]		ResNet-18	8	-	Kinetics	66.7	64.0	69.4
CRW [15]		ResNet-18	8	-	Kinetics	67.6	64.8	70.2
JSTG [52]		ResNet-18	8	-	Kinetics	68.7	65.8	71.6
DUL [1]		ResNet-18	8	-	YTV	69.3	67.1	71.6
Ours		ResNet-18	8	-	YTV	69.0	66.4	71.7
Ours		ResNet-18	8	ImageNet	YTV	**70.5**	**67.8**	**73.2**
MAST [20]		ResNet-18	4	-	YTV	65.5	63.3	67.6
MAMP [29]		ResNet-18	4	-	YTV	69.7	68.3	71.2
Ours		ResNet-18	4	-	YTV	71.2	68.9	73.8
Ours		ResNet-18	4	ImageNet	YTV	**73.1**	**70.4**	**75.9**
MoCo [13]		ResNet-50	8	ImageNet	-	65.4	63.2	67.6
SimSiam [5]		ResNet-50	8	ImageNet	-	66.3	64.5	68.2
TimeCycle [41]		ResNet-50	8	-	VLOG	48.7	46.4	50.0
UVC [21]		ResNet-50	8	COCO	Kinetics	56.3	54.5	58.1
SeCo [51]		ResNet-50	8	-	Kinetics	60.6	60.4	62.8
VINCE [10]		ResNet-50	8	-	Kinetics	65.6	63.4	67.8
VFS [47]		ResNet-50	8	-	Kinetics	68.9	66.5	71.3
Ours		ResNet-50	8	ImageNet	YTV	**71.3**	**68.5**	**74.0**
Supervised [12]	✓	ResNet-18	8	ImageNet	-	62.9	60.6	65.2
Supervised [12]	✓	ResNet-50	8	ImageNet	-	66.0	63.7	68.4
OnAVOS [37]	✓	ResNet-38	-	I + C + P	D	65.4	61.6	69.1
OSVOS-S [27]	✓	VGG-16	-	I + P	D	68.0	64.7	71.3
FEELVOS [38]	✓	Xception-65	-	I + C	D + YTV	71.5	69.1	74.0

Table 2: **Quantitative results for video object segmentation on validation set of DAVIS-2017** [32]. We show results of state-of-the-art self-supervised methods and some supervised methods for comparison. We report the data size for self-supervised methods (total number/duration of image/video dataset). I:ImageNet [6] (1.28m). C:COCO [23] (30k). O:OxUvA [35] (14h). T:TrackingNet [30] (300h). K:Kinetics [4] (800h). V:VLOG [9] (344h). YTV:YouTube-VOS [48] (5h). D:DAVIS-2017 [32] (-). P:PASCAL-VOC [8] (-).

using the same architecture. Benefiting from less information loss for temporal feature learning by setting the stride of the encoder to 4, the performance of our method reaches 73.1%, leading to a performance gain of 3.4% over MAMP [29], which consistently verify the idea of our methods. For ResNet-50, our method still outperforms VFS [47] by 2.4%. It is worth noting that [15][21][40][47][52] are all pre-trained on large-scale video datasets, *i.e.*, Kinetics [4], TrackingNet [30], while our method adopt a small video dataset plus an image dataset which has a much smaller data size than video. Our method pre-trained only on YouTube-VOS gets 69.0%/71.2%, which is impressive. More remarkably, Our method even outperforms some task-specific fully-supervised algorithms [27][37][38].

Methods	Sup.	VIP	JHMDB	
		mIoU ↑	PCK@0.1 ↑	PCK@0.2 ↑
TimeCycle [41]		28.9	57.3	78.1
UVC [21]		34.1	58.6	79.6
CRW [15]		38.6	59.3	80.3
ContrastCorr [40]		37.4	61.1	80.8
VFS [47]		39.9	60.5	79.5
CLTC [16]		37.8	60.5	82.3
JSTG [52]		40.2	61.4	**85.3**
Ours		**41.0**	**63.1**	82.9
ResNet-18 [12]	✓	31.9	53.8	74.6
ATEN [53]	✓	37.9	-	-
Thin-Slicing Net [33]	✓	-	68.7	92.1

Table 3: **Quantitative results for human part propagation and human pose tracking.** We show results of state-of-the-art self-supervised methods and some supervised methods for comparison.

Results for human part propagation. Next, we evaluate our method for human part tracking. Experiments are conducted on the validation set of VIP [53], which consists of 50 videos with 19 human semantic part classes, requiring more precise matching than DAVIS-2017 [32]. Following [53], we adopt mean intersection-over-union (mIoU) as our evaluation metric and resize the video frames to 560×560. All models except TimeCycle [41] are set to ResNet-18 with a stride of 8 for a fair comparison. The results are shown in Table 3. Our method achieves state-of-the-art performance, surpassing all previous state-of-the-art by 0.8%. Notably, our

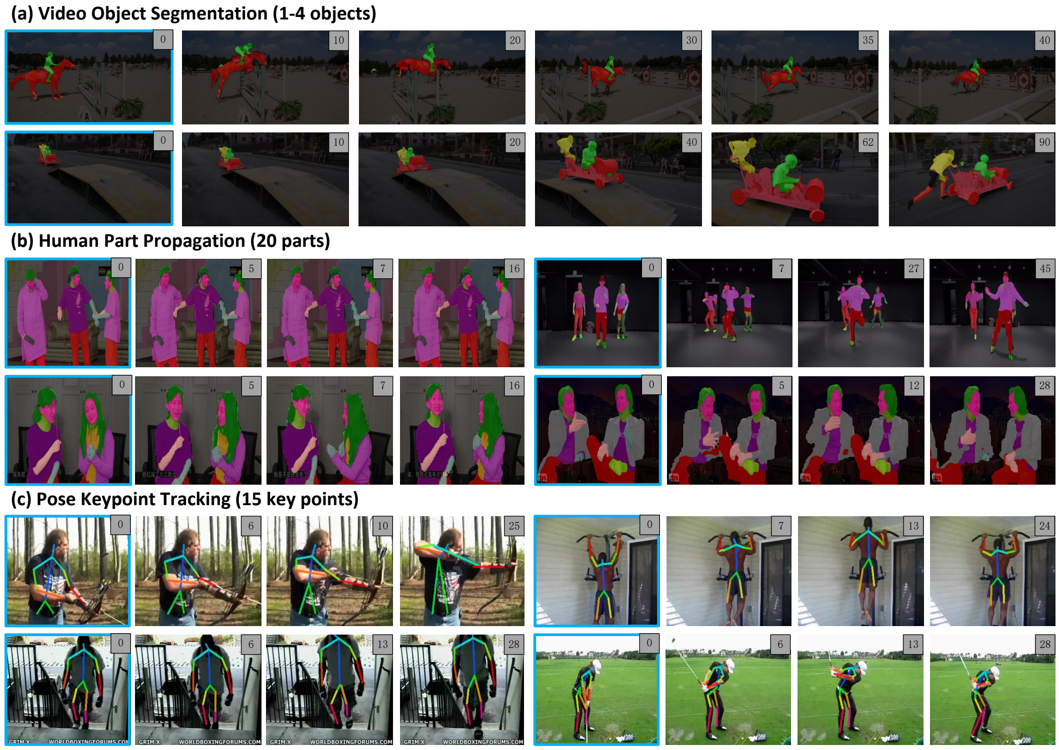

Figure 4: Qualitative results for label propagation. Given the first frame with different annotations highlighted with a blue outline, we propagate it to the current frame without fine-tuning. (a) Video object segmentation on DAVIS-2017 [32]. (b) Human part propagation on VIP [53]. (c) Pose keypoint tracking on JHMDB [17].

model outperforms ATEN [53] which is specifically designed for the task using human annotations. Figure 4 (b) depicts some visualization results on several representative videos.

Results for human pose tracking. We then make a performance comparison on the downstream task of human pose tracking. We conduct the experiments on the validation of JHMDB [17] which has 268 videos. The annotations consist of 15 body joints for each person. The probability of correct keypoint [50] is utilized here to examine the accuracy between result and ground truth with different thresholds. Follow the evaluation protocol of [15][21], we resize the video frames to 320×320. The results in Table 3 show a consistent performance gain over previous methods, which successfully demonstrates the transferability of our method to different downstream tasks. The visualization results in Figure 4 (c) show the robustness of our approach to various challenges.

5 Conclusions

In this work, we look into the self-supervised video correspondence learning from two perspectives: (1) Learning an appearance model invariant for appearance changes. (2) Learning a better temporal representation to capture fine-grained correspondence between video frames. To materialize our idea, we design a novel framework that learns the spatial and temporal feature sequentially. More specifically, we first train a model using contrastive loss on ImageNet. Then temporal feature learning is performed with the objective of frame reconstruction. To achieve the goal of realizing fine-grained matching and maintaining spatial representation at the same time, we devise a pyramid learning framework consisting of local and global correlation distillation. The distillation of local correlation is achieved between the different pyramid levels since the bottom level always contains richer information for temporal feature learning, while the global correlation distillation is conducted at the coarse pyramid level, which is closely related to finding the correspondence at the object-level. Furthermore, the entropy-based selection is proposed to pay more attention to the region with high uncertainty when applying the local correlation distillation. Extensive experiments on a variety of downstream tasks validate our method. More remarkably, self-supervised training shows superior performance to fully-supervised ImageNet pre-training and several dedicated methods with task-specific human annotations.

References

[1] N. Araslanov, S. Schaub-Meyer, and S. Roth. Dense unsupervised learning for video segmentation. In *NeurIPS*, 2021.

[2] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. One-shot video object segmentation. In *CVPR*, 2017.

[3] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

[4] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[5] X. Chen and K. He. Exploring simple siamese representation learning. In *CVPR*, 2021.

[6] J. Deng. A large-scale hierarchical image database. In *CVPR*, 2009.

[7] 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 *ICCV*, 2015.

[8] M. Everingham, S. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 2015.

[9] D. F. Fouhey, W.-c. Kuo, A. A. Efros, and J. Malik. From lifestyle vlogs to everyday interactions. In *CVPR*, 2018.

[10] D. Gordon, K. Ehsani, D. Fox, and A. Farhadi. Watching the world go by: Representation learning from unlabeled videos. *arXiv preprint arXiv:2003.07990*, 2020.

[11] 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. In *NeurIPS*, 2020.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[13] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[14] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial intelligence*, 1981.

[15] A. Jabri, A. Owens, and A. Efros. Space-time correspondence as a contrastive random walk. In *NeurIPS*, 2020.

[16] S. Jeon, D. Min, S. Kim, and K. Sohn. Mining better samples for contrastive learning of temporal correspondence. In *CVPR*, 2021.

[17] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *ICCV*, 2013.

[18] R. Jonschkowski, A. Stone, J. T. Barron, A. Gordon, K. Konolige, and A. Angelova. What matters in unsupervised optical flow. In *ECCV*, 2020.

[19] Z. Lai and W. Xie. Self-supervised learning for video correspondence flow. In *BMVC*, 2019.

[20] Z. Lai, E. Lu, and W. Xie. Mast: A memory-augmented self-supervised tracker. In *CVPR*, 2020.

[21] X. Li, S. Liu, S. De Mello, X. Wang, J. Kautz, and M.-H. Yang. Joint-task self-supervised learning for temporal correspondence. In *NeurIPS*, 2019.

[22] Z. Li and D. Hoiem. Learning without forgetting. *IEEE transactions on PAMI*, 2017.

[23] 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*, 2014.

[24] L. Liu, J. Zhang, R. He, Y. Liu, Y. Wang, Y. Tai, D. Luo, C. Wang, J. Li, and F. Huang. Learning by analogy: Reliable supervision from transformations for unsupervised optical flow estimation. In *CVPR*, 2020.

[25] P. Liu, I. King, M. R. Lyu, and J. Xu. Ddflow: Learning optical flow with unlabeled data distillation. In *AAAI*, 2019.

[26] X. Lu, W. Wang, J. Shen, Y.-W. Tai, D. J. Crandall, and S. C. Hoi. Learning video object segmentation from unlabeled videos. In *CVPR*, 2020.

[27] K.-K. Maninis, S. Caelles, Y. Chen, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. Video object segmentation without temporal information. *IEEE transactions on PAMI*, 2018.

[28] S. Meister, J. Hur, and S. Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, 2018.

[29] B. Miao, M. Bennamoun, Y. Gao, and A. Mian. Self-supervised video object segmentation by motion-aware mask propagation. *arXiv preprint arXiv:2107.12569*, 2021.

[30] M. Muller, A. Bibi, S. Giancola, S. Alsubaihi, and B. Ghanem. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In *ECCV*, 2018.

[31] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim. Video object segmentation using space-time memory networks. In *ICCV*, 2019.

[32] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[33] J. Song, L. Wang, L. Van Gool, and O. Hilliges. Thin-slicing network: A deep structured model for pose estimation in videos. In *CVPR*, 2017.

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

[35] J. Valmadre, L. Bertinetto, J. F. Henriques, R. Tao, A. Vedaldi, A. W. Smeulders, P. H. Torr, and E. Gavves. Long-term tracking in the wild: A benchmark. In *ECCV*, 2018.

[36] A. Van den Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv e-prints*, pages arXiv–1807, 2018.

[37] P. Voigtlaender and B. Leibe. Online adaptation of convolutional neural networks for video object segmentation. *arXiv preprint arXiv:1706.09364*, 2017.

[38] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, and L.-C. Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *CVPR*, 2019.

[39] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy. Tracking emerges by colorizing videos. In *ECCV*, 2018.

[40] N. Wang, W. Zhou, and H. Li. Contrastive transformation for self-supervised correspondence learning. In *AAAI*, 2020.

[41] X. Wang, A. Jabri, and A. A. Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, 2019.

[42] X. Wang, R. Zhang, C. Shen, T. Kong, and L. Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, 2021.

[43] Z. Wang, H. Zhao, Y.-L. Li, S. Wang, P. Torr, and L. Bertinetto. Do different tracking tasks require different appearance models? In *NeurIPS*, 2021.

[44] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.

[45] E. Xie, J. Ding, W. Wang, X. Zhan, H. Xu, P. Sun, Z. Li, and P. Luo. Detco: Unsupervised contrastive learning for object detection. In *ICCV*, 2021.

[46] Y. Xiu, J. Li, H. Wang, Y. Fang, and C. Lu. Pose flow: Efficient online pose tracking. In *BMVC*, 2018.

[47] J. Xu and X. Wang. Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. In *ICCV*, 2021.

[48] N. Xu, L. Yang, Y. Fan, D. Yue, Y. Liang, J. Yang, and T. Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018.

[49] C. Yang, Z. Wu, B. Zhou, and S. Lin. Instance localization for self-supervised detection pretraining. In *CVPR*, 2021.

[50] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE transactions on PAMI*, 2012.

[51] T. Yao, Y. Zhang, Z. Qiu, Y. Pan, and T. Mei. Seco: Exploring sequence supervision for unsupervised representation learning. In *AAAI*, 2021.

[52] Z. Zhao, Y. Jin, and P.-A. Heng. Modelling neighbor relation in joint space-time graph for video correspondence learning. In *ICCV*, 2021.

[53] Q. Zhou, X. Liang, K. Gong, and L. Lin. Adaptive temporal encoding network for video instance-level human parsing. In *ACM MM*, 2018.

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]**