Boosting Video Object Segmentation via Space-time Correspondence Learning

Yurong Zhang[1]*, Liulei Li[2]*, Wenguan Wang[2]†, Rong Xie[1], Li Song[1], Wenjun Zhang[1]
[1]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University [2]ReLER, CCAI, Zhejiang University
https://github.com/wenguanwang/VOS_Correspondence

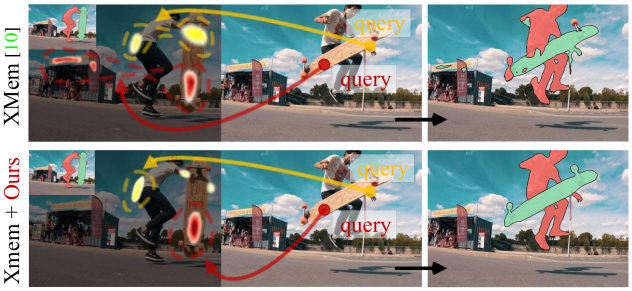
Abstract

*Current top-leading solutions for video object segmentation (VOS) typically follow a **matching-based** regime: for each query frame, the segmentation mask is inferred according to its correspondence to previously processed and the first annotated frames. They simply exploit the supervisory signals from the groundtruth masks for learning mask prediction only, without posing any constraint on the space-time correspondence matching, which, however, is the fundamental building block of such regime. To alleviate this crucial yet commonly ignored issue, we devise a correspondence-aware training framework, which boosts matching-based VOS solutions by explicitly encouraging robust correspondence matching during network learning. Through comprehensively exploring the intrinsic coherence in videos on pixel and object levels, our algorithm reinforces the standard, fully supervised training of mask segmentation with label-free, contrastive correspondence learning. Without neither requiring extra annotation cost during training, nor causing speed delay during deployment, nor incurring architectural modification, our algorithm provides solid performance gains on four widely used benchmarks,* i.e.*, DAVIS2016&2017, and YouTube-VOS2018&2019, on the top of famous matching-based VOS solutions.*

1. Introduction

In this work, we address the task of (one-shot) video object segmentation (VOS) [5,73,96]. Given an input video with groundtruth object masks in the first frame, VOS aims at accurately segmenting the annotated objects in the subsequent frames. As one of the most challenging tasks in computer vision, VOS benefits a wide range of applications including augmented reality and interactive video editing [72].

Modern VOS solutions are built upon fully supervised deep learning techniques and the top-performing ones [10, 12] largely follow a *matching-based* paradigm, where the object masks for a new coming frame (*i.e.*, query frame) are generated according to the correlations between the query frame and the previously segmented as well as first annotated frames (*i.e.*, reference frames), which are stored in an outside memory. It is thus apparent that the module for cross-frame matching (*i.e.*, space-time correspondence modeling) plays the central role in these advanced VOS systems. Nevertheless, these matching-based solutions are simply trained under the direct supervision of the groundtruth segmentation masks. In other words, during training, the whole VOS system is purely optimized towards accurate segmentation mask prediction, yet without taking into account any *explicit* constraint/regularization on the central component — space-time correspondence matching. This comes with a legitimate concern for sub-optimal performance, since there is no any solid guarantee of truly establishing reliable cross-frame correspondence during network learning. Fig. 1(a) offers a visual evidence for this viewpoint. XMem [10], the latest state-of-the-art matching-based VOS solution, tends to struggle at discovering valid space-time correspondence; indeed, some background pixels/patches are incorrectly recognized as highly correlated to the query foreground.



(a) *reference* frame (*w.* mask) (b) *query* frame (c) segmentation result of *query* frame

Figure 1. (a-b) shows some correspondences between a reference frame and a query frame. (c) gives mask prediction. XMem [10], even a top-leading matching-based VOS solution, still suffers from unreliable correspondence. In contrast, with our correspondence-aware training strategy, robust space-time correspondence can be established, hence leading to better mask-tracking results.

The aforementioned discussions motivate us to propose a new, space-time correspondence-aware training framework which addresses the weakness of existing matching-based

*The first two authors contribute equally to this work.
†Corresponding author.

VOS solutions in an elegant and targeted manner. The core idea is to empower the matching-based solutions with enhanced robustness of correspondence matching, through mining complementary yet *free* supervisory signals from the inherent nature of space-time continuity of training video sequences. In more detail, we comprehensively investigate the coherence nature of videos on both pixel and object levels: **i)** *pixel-level consistency*: spatiotemporally proximate pixels/patches tend to be consistent; and **ii)** *object-level coherence*: visual semantics of same object instances at different timesteps tend to retain unchanged. By accommodating these two properties to an unsupervised learning scheme, we give more explicit direction on the correspondence matching process, hence promoting the VOS model to learn dense discriminative and object-coherent visual representation for robust, matching-based mask tracking (see Fig. 1 (b-c)).

It is worth mentioning that, beyond boosting the segmentation performance, our space-time correspondence-aware training framework enjoys several compelling facets. **First**, our algorithm supplements the standard, fully supervised training paradigm of matching-based VOS with *self-training* of space-time correspondence. As a result, it does not cause any extra annotation burden. **Second**, our algorithm is fully compatible with current popular matching-based VOS solutions [10, 12], without particular adaption to the segmentation network architecture. This is because the learning of the correspondence matching only happens in the visual embedding space. **Third**, as a training framework, our algorithm does not produce additional computational budget to the applied VOS models during the deployment phase.

We make extensive experiments on various gold-standard VOS datasets, *i.e.*, DAVIS2016&2017 [52], and YouTube-VOS2018&2019 [86]. We empirically prove that, on the top of recent matching-based VOS models, *i.e.*, STCN [12] and XMem [10], our approach gains impressive results, surpassing all existing state-of-the-arts. Concretely, in multi-object scenarios, it improves STCN by **1.2**%, **2.3**%, and **2.3**%, and XMem by **1.5**%, **1.2**%, and **1.1**% on DAVIS2017$_{val}$, Youtube-VOS2018$_{val}$, as well as Youtube-VOS2019$_{val}$, respectively, in terms of $J\&F$. Besides, it respectively promotes STCN and XMem by **0.4**% and **0.7**% on single-object benchmark dataset DAVIS2016$_{val}$.

2. Related Work

(One-Shot) Video Object Segmentation. Recent VOS solutions can be roughly categorized into three groups: **i)** *Online learning* based methods adopt online fine-tuning [5, 40, 65] or adaption [3, 43, 48, 53] techniques to accommodate a pre-trained generic segmentation network to the test-time target objects. Though impressive, they are typically hyper-parameter sensitive and low efficient. **ii)** *Propagation-based* methods [2, 7, 13, 22, 23, 25, 42, 46, 50, 63, 71, 78, 94] formulate VOS as a frame-by-frame mask propagation process. Though compact, they heavily rely on the previous segmentation mask, hence easily trapping in occlusion cases and suffering from error accumulation. **iii)** *Matching-based methods* instead leverage the first annotated frame (and previous segmented frames) to build an explicit object model, according to which query pixels are matched and classified [8, 64, 88]. As a landmark in this line, STM [47] introduces an external memory for explicitly and persistently storing the representations and masks of past frames, allowing for long-term matching. Since then, matching-based solutions [11, 12, 21, 33, 38, 41, 44, 49, 55, 56, 67, 82] dominate this area due to the superior performance and high efficiency [96].

Recent studies for matching-based VOS mainly focus on improving network designs, through, for instance, building more efficient memory [10, 12, 36–38, 79], adopting local matching [21, 55, 91], and incorporating background context [88]. However, our contribution is orthogonal to these studies, as we advance the matching-based regime in the aspect of model learning. We devise a new training framework that improves the standard, supervised segmentation training protocol with self-constrained correspondence learning. We show our algorithm can be seamlessly incorporated into the latest arts [10, 12] with notable performance gains.

Self-supervised Space-time Correspondence Learning. Capturing cross-frame correlations is a long-standing task in the field of computer vision, due to its vital role in many video applications such as optical flow estimation, and object tracking. A line of recent work tackles this problem in a self-supervised learning fashion. The methods can be divided into three classes: **i)** *Reconstruction* based methods enforce the network to reconstruct a query frame from a neighboring frame [1, 3, 26, 29, 30, 32, 66, 70], so as to find accurate alignment between the query and reference frames. **ii)** *Cycle-consistency* based methods [24, 34, 39, 59, 69, 75, 95] conduct forward-backward tracking and learn correspondence by penalizing the disagreement between start and end points. **iii)** *Contrastive learning* based methods [1, 26, 27, 57, 85] emerged very recently, inspired by the astonishing success of contrastive learning in self-supervised image representation learning. Their core idea is to distinguish confident correspondences from a large set of unlikely ones.

Although a few correspondence learning methods also report performance on mask-tracking, they confine focus to the self-supervised setting and simply treat VOS as an exemplar application task without contributing neither dedicated model design nor specific insight to VOS. This work represents a very early (if not the first) effort towards boosting supervised learning of VOS with self-supervised correspondence learning, within a principled training framework. Hence our ultimate goal is supervised learning of VOS, yet space-time correspondence learning is a mean to this end.

Object-level Self-supervised Learning. Self-supervised visual representation learning aims to learn transferable fea-

tures with massive unlabeled data. Recently, contrastive learning based methods [6, 15–17, 60, 80, 81, 92, 97] made rapid progress. They are build upon an *instance discrimination* task that maximizes the agreement between different augmented views of the same image. Yet, it then became apparent that image-level pretraining is suboptimal to dense prediction tasks [76, 77], due to the discrepancy between holistic representation and fine-grained task nature. Hence a growing number of work investigate pixel-level pretraining [1, 4, 45, 74, 76, 84]. Though addressing local semantics, they fail to learn object-level visual properties. In view of the limitations of image-level and pixel-level self-supervised learning, some latest efforts [19, 20, 31, 54, 62, 77, 83, 90] turn to exploring object-level pretraining, with the aid of heuristic object proposals [19, 77, 83], saliency [54, 62], or clustering [20]. We assimilate the insight of object-level representation learning and perform adaption to self-supervised correspondence learning. This leads to a comprehensive solution for both dense and object-oriented correlation modeling, hence grasping the central properties of matching-based VOS.

3. Methodology

Fig. 3 depicts a diagram of our algorithm. Before elucidating our correspondence-aware training framework for VOS (*cf.* §3.2), we first formalize the task of interest and provide preliminaries on recent advanced matching-based VOS solutions (*cf.* §3.1). Finally, §3.3 gives implementation details.

3.1. Problem Statement and Preliminaries

Task Setup. In VOS, the target object is predefined by a reference mask in the first frame. Formally, given a video with T frames $\mathcal{I}=\{I_t\}_{t=1}^{T}$ and the first-frame reference mask Y_1, a robust VOS solution f should exploit Y_1 to produce accurate object masks $\{\hat{Y}_t\}_{t=2}^{T}$ for the rest $T-1$ frames $\{I_t\}_{t=2}^{T}$:

$$\{\hat{Y}_t\}_{t=2}^{T} = f(\{I_t\}_{t=1}^{T}, Y_1). \tag{1}$$

Modern VOS solutions often rely on supervised deep learning techniques. For a training video $\mathcal{I}=\{I_t\}_{t=1}^{T}$, let us denote its groundtruth mask sequence as $\mathcal{Y}=\{Y_t\}_{t=1}^{T}$. The optimal solution f^* is found by minimizing a supervised segmentation loss $\mathcal{L}_{\text{SEG}}$, on a set of N training pairs $\{\mathcal{I}_n, \mathcal{Y}_n\}_{n=1}^{N}$:

$$f^* = \arg\min_{f} \frac{1}{NT} \sum\nolimits_n \sum\nolimits_t \mathcal{L}_{\text{SEG}}(\hat{Y}_t, Y_t), \tag{2}$$

where $\mathcal{L}_{\text{SEG}}$ is typically the well-known cross-entropy loss.

Revisit Matching-based VOS Solutions. Since the seminal work of STM [47], VOS solutions largely adopt a matching-based regime, where the mask $\hat{Y}^q$ of current query frame I^q is predicted according to the correlations between I^q and past processed frames $\{I_k^r\}_{k=1}^{K}$. The reference frames $\{I_k^r\}_k$ and masks $\{\hat{Y}_k^r\}_k$ are stored in an external *memory*, easing the access of long-term historic context. We are particularly interested in the latest matching-based models, *i.e.*, STCN [12]

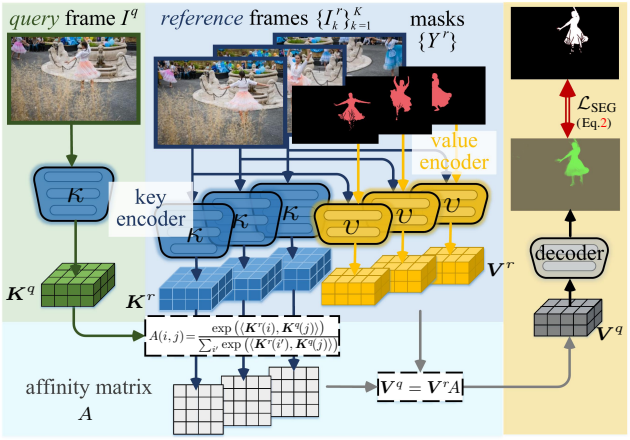


Figure 2. Illustration of recent advanced matching-based VOS solutions [10, 12], which are simply trained with the manually annotated segmentation masks, without posing any explicit supervisory signal on correspondence matching (*i.e.*, affinity estimation).

and XMem [10], due to their high performance and elegant model design. Specifically, they have two core components:

- *Key encoder* κ takes one single frame image as input and outputs a dense visual feature, *i.e.*, $\boldsymbol{K}=\kappa(I)\in\mathbb{R}^{C\times HW}$, with HW spatial dimension and C channels.
- *Value encoder* υ extracts mask-embedded representation for paired frame and mask, *i.e.*, $\boldsymbol{V}=\upsilon(I,\hat{Y})\in\mathbb{R}^{D\times HW}$.

Given a query frame I^q and K reference frame and mask pairs $\{(I_k^r, \hat{Y}_k^r)\}_{k=1}^{K}$ stored in the memory, we have: query key $\boldsymbol{K}^q\in\mathbb{R}^{C\times HW}$, memory key $\boldsymbol{K}^r\in\mathbb{R}^{C\times KHW}$, and memory value $\boldsymbol{V}^r\in\mathbb{R}^{D\times KHW}$. Then we compute the (augmented) affinity matrix $A\in[0,1]^{KHW\times HW}$ between $\boldsymbol{K}^r$ and $\boldsymbol{K}^q$:

$$A(i,j)=\frac{\exp\left(\langle \boldsymbol{K}^r(i), \boldsymbol{K}^q(j)\rangle\right)}{\sum_{i'}\exp\left(\langle \boldsymbol{K}^r(i'), \boldsymbol{K}^q(j)\rangle\right)}, \tag{3}$$

where $\boldsymbol{K}(i)\in\mathbb{R}^C$ denotes the feature vector of i-th position in $\boldsymbol{K}$, and $\langle\cdot,\cdot\rangle$ is a similarity measure, *e.g.*, ℓ_2 distance. In this way, $A(i,j)\in[0,1]$ – the (i,j)-th element in the normalized affinity A – signifies the proximity between i-th pixel in the reference $\{I_k^r\}_k$ and j-th pixel in the query I^q.

Next, a supportive feature $\boldsymbol{V}^q\in\mathbb{R}^{D\times HW}$ for the query can be created by aggregating memory value features using A:

$$\boldsymbol{V}^q = \boldsymbol{V}^r A. \tag{4}$$

$\boldsymbol{V}^q$ is fed into a *decoder* to output the mask $\hat{Y}^q$. $(I^q, \hat{Y}^q)$ can be further added into the memory as new reference, and the query key is reused as the memory key. As the memory and decoder are not our focus, we refer to [10, 12] for details.

3.2. Space-time Correspondence-aware Training

Core Idea. From Eq. 3 we can find that, the affinity A gives the strength of all the pixel pairwise correlations between the query frame I_q and the memorized reference frames I_r. Thus Eq. 3 essentially performs correspondence matching

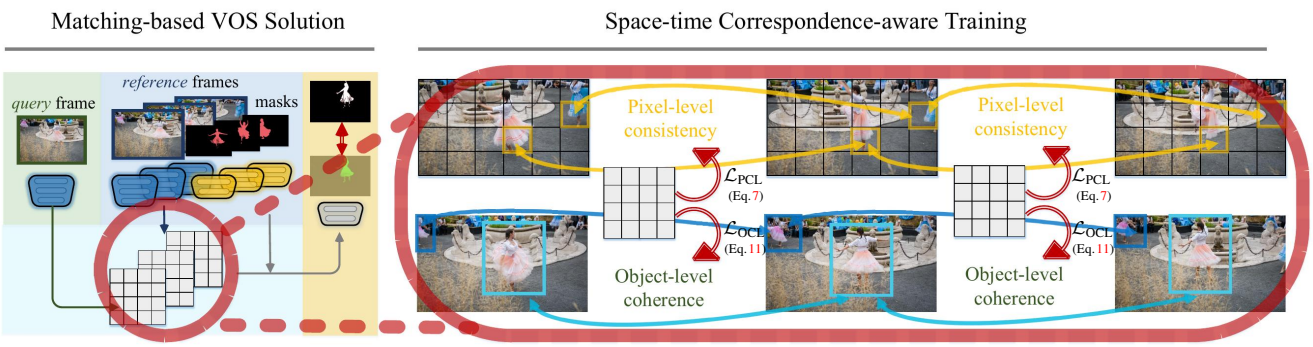


Figure 3. Diagram of our proposed space-time correspondence-aware training framework for matching-based VOS.

between the query and the memory (despite the normalization over all the reference frames), and the computed affinity A serves as the basis for the final mask decoding in Eq. 4. Nevertheless, most existing matching-based VOS models are simply trained by minimizing the standard, supervised segmentation loss $\mathcal{L}_{\text{SEG}}$ (*cf.* Eq.2). As a result, during training, the correspondence matching component (*cf.* Eq. 2) can only access the *implicit*, segmentation-oriented supervision signals, yet lacking *explicit* constraint/regulizarition over the cross-frame correlation estimation – A. This may result in unreliable pixel association, hence suffering from sub-optimal performance eventually.

Noticing the crucial role of space-time correspondence and the deficiency of standard training strategy in the context of matching-based VOS, we hence seek to complement the segmentation-specified learning objective $\mathcal{L}_{\text{SEG}}$ (*cf.* Eq.2) with certain correspondence-aware training target. However, obtaining annotations of space-time correspondence for real videos is almost prohibitive, due to occlusions and free-form object deformations. This further motivates us to explore the intrinsic coherence of videos as a source of free supervision for correspondence matching. The delivered outcome is a powerful training framework that reinforces matching-based VOS models with *annotation-free* correspondence learning.

Basically speaking, we holistically explore the coherence nature of video sequences on *pixel* and *object* granularities, within a contrastive correspondence learning scheme.

Correspondence learning based on Pixel-level Consistency. We first address local continuity residing in videos, *i.e.*, spatiotemporally adjacent pixels/patches typically yield consistent patterns. Specifically, given a training video $\mathcal{I}$, we sample two successive frames I_t, I_{t+1} as well as an *anchor* frame I_τ, where $\tau \neq t$ and $\tau \neq t+1$. Based on a contrastive formulation, our approach estimates pixel pairwise correlations of I_t and I_{t+1} *w.r.t.* the anchor I_τ, and learns correspondence matching by enforcing these correlations to be spatially consistent across I_t and I_{t+1}. More precisely, with the key encoder κ, we have the dense visual representations, *i.e.*, $\boldsymbol{K}_t, \boldsymbol{K}_{t+1}, \boldsymbol{K}_\tau \in \mathbb{R}^{C\times HW}$, of I_t, I_{t+1}, and I_τ, respectively. Then, we compute the affinity of the pixel/patch feature vector at i-th position of $\boldsymbol{K}_t$ *w.r.t.* the anchor feature tensor $\boldsymbol{K}_\tau$:

$$A^{t,\tau}(i,j) = \frac{\exp\left(\langle \boldsymbol{K}_t(i), \boldsymbol{K}_\tau(j)\rangle\right)}{\sum_{j'} \exp\left(\langle \boldsymbol{K}_t(i), \boldsymbol{K}_\tau(j')\rangle\right)}. \tag{5}$$

We acquire the pixel/patch j^* in $\boldsymbol{K}_\tau$ that best matches the pixel/patch i in $\boldsymbol{K}_t$ (see Fig. 4):

$$j^* = \underset{j\in\{1,\cdots,HW\}}{\arg\max}\ A^{t,\tau}(i,j). \tag{6}$$

The index of the best alignment j^* of $\boldsymbol{K}_t(i)$ is subtly used as the pseudo label for correspondence matching between the i-th feature vector of $\boldsymbol{K}_{t+1}$ and the anchor feature tensor $\boldsymbol{K}_\tau$, hence addressing local consistency and enabling self-supervised learning of pixel-level correspondence matching:

$$\mathcal{L}_{\text{PCL}} = -\log \sum_i \frac{\exp\left(\langle \boldsymbol{K}_{t+1}(i), \boldsymbol{K}_\tau(j^*)\rangle\right)}{\sum_j \exp\left(\langle \boldsymbol{K}_{t+1}(i), \boldsymbol{K}_\tau(j)\rangle\right)}. \tag{7}$$

Such self-supervised loss trains the model to distinguish the aligned pair, *i.e.*, $(\boldsymbol{K}_{t+1}(i), \boldsymbol{K}_\tau(j^*))$, from the set of non-corresponding ones, *i.e.*, $\{(\boldsymbol{K}_{t+1}(i), \boldsymbol{K}_\tau(j))\}_{j\neq j^*}$, based on the assignment of $\boldsymbol{K}_t(i)$, which is located at the same spatial position of $\boldsymbol{K}_{t+1}(i)$, to the anchor $\boldsymbol{K}_\tau$. Through this self-training mechanism, the model learns to assign the features

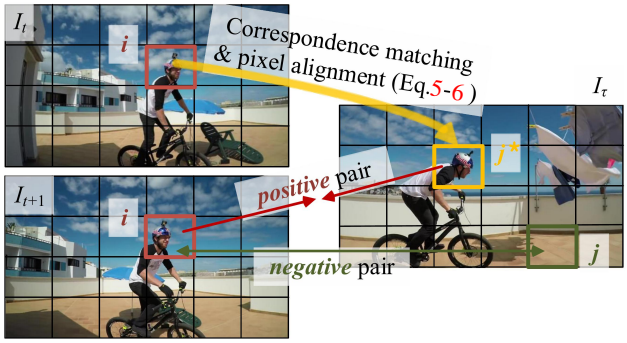


Figure 4. Given two successive frames I_t, I_{t+1} and an *anchor* frame I_τ, sampled from a same training video $\mathcal{I}$, we first make pixel-level correspondence matching between I_t and I_τ, through Eq. 5-6. Then the alignment results are used as the pseudo label for the contrastive correspondence learning (*i.e.*, Eq. 7) between I_{t+1} and I_τ, based on the local continuity assumption, *i.e.*, I_t and I_{t+1} yield consistent patterns at spatially adjacent locations. For clarity, negative pixel/patch samples from other training videos are omitted.

in frame I_t consistently with the temporally proximate frame I_{t+1} *w.r.t.* I_τ, so as to impose the desired property of local consistency on the visual embedding space κ and encourage reliable correspondence matching explicitly. Moreover, following the common practice of contrastive learning [6, 17], we randomly sample pixel/patch features from other videos in the training batch as negative examples during the computation of $\mathcal{L}_{\text{PCL}}$, teaching the model to efficiently disambiguate correspondence on both inter- and intra-video levels.

Correspondence Learning based on Object-level Coherence. Through implementing the pixel-level consistency property in our framework, we inspire matching-based VOS models to learn locally distinctive features, hence constructing reliable, dense correspondence between video frames. For the sake of full-scale robust matching, we further investigate the content continuity of videos on the object-level – representations of a same object instance should remain stable across frames. By enforcing the key encoder κ to learn object-level compact and discriminative representations, we are able to boost the robustness of correspondence matching against local disturbance caused by deformation and occlusion, and better address the object-aware nature of the VOS task. Put simply, we apply contrastive correspondence learning on both automatically discovered and pre-labeled video objects; object-level, space-time correspondence is learnt by maximizing the similarity of the representations of the same object instance at different timesteps.

VOS training videos often involve complex visual scenes with multiple objects, while only a small portion of the object instances are labeled [52, 86] (*i.e.*, the masked objects in Fig. 4 (a-b)). We thus adopt Selective Search [61], an unsupervised, object proposal algorithm, to obtain an exhaustive set of potential objects for each training frame I (plotted in solid boxes in Fig. 5 (a-b)). The automatically discovered object proposals substantially provide diverse training samples to aid object-level correspondence learning, without adding extra annotation cost. Formally, let $\mathcal{P}=\{P_i\}_i$ denote the full set of automatically discovered objects and pre-labeled ones in frame I. Each object $P=(x, y, w, h)$ is represented as a bounding box, whose center is (x, y) and the size is of $w\times h$, and its object-level representation $\boldsymbol{p}\in\mathbb{R}^D$ is given by:

$$\boldsymbol{p} = \texttt{AVGPool}(\texttt{RoIAlign}(\boldsymbol{K}, P)). \tag{8}$$

Given two distant frames, I_t and $I_{t'}$, sampled from training video $\mathcal{I}$, as well as their corresponding object sets, *i.e.*, $\mathcal{P}$ and $\mathcal{P}'$, a small set of objects are first drawn from $\mathcal{P}$, *i.e.*, $\mathcal{Q}\subset\mathcal{P}$; the objects are sampled as the spatial distribution of the centers are sparse, so as to ensure the objects are from different instances. Note that we make $|\mathcal{P}'|\gg|\mathcal{Q}|$ to ensure $\mathcal{P}'$ covers all the instances appeared in $\mathcal{Q}$. Given $\mathcal{Q}$, we are next to find the corresponding objects in $\mathcal{P}'$, which can be formulated as a bipartite matching problem* (see Fig. 5 (c)):

$$\begin{aligned} &\max_{A^{t,t'}} \sum\nolimits_i \sum\nolimits_j \langle \boldsymbol{p}_i, \boldsymbol{p}'_j \rangle \cdot A^{t,t'}(i,j), \\ \textit{s.t.}\quad &\forall p_i \in \mathcal{Q}, \quad \sum\nolimits_{j=1} A^{t,t'}(i,j) \le 1, \\ &\forall p'_j \in \mathcal{P}', \quad \sum\nolimits_{i=1} A^{t,t'}(i,j) \le 1, \\ &\forall (p_i, p'_j), \quad A^{t,t'}(i,j) \in \{0,1\}, \end{aligned} \tag{9}$$

where $A^{t,t'} \in \{0,1\}^{|\mathcal{Q}|\times|\mathcal{P}'|}$ refers to the assignment of $\mathcal{Q}$ *w.r.t.* $\mathcal{P}'$, and the constraints ensure exclusive assignment. The global optimal solution of Eq. 9 can be acquired by the Hungarian algorithm [28], and is cleverly leveraged as the pseudo label for our object-level correspondence learning.

Concretely, for each object $p_i \in \mathcal{Q}$, the index of its aligned counterpart in $\mathcal{P}'$ can be directly derived from $A^{t,t'}$:

$$j^* = \underset{j\in\{1,\cdots,|\mathcal{P}'|\}}{\arg\max}\ A^{t,t'}(i,j). \tag{10}$$

Given a query object $p_i \in \mathcal{Q}$, we view $p'_{j^*} \in \mathcal{P}'$ as a positive sample while objects from other videos in the training batch as negative samples (denoted as $\mathcal{O}=\{O_1, O_2, \cdots\}$). Hence the contrastive learning objective of our object-level correspondence can be formulated as:

$$\mathcal{L}_{\text{OCL}} = -\log \sum_{p_i\in\mathcal{Q}} \frac{\exp\left(\langle \boldsymbol{p}_i, \boldsymbol{p}'_{j^*} \rangle\right)}{\exp\left(\langle \boldsymbol{p}_i, \boldsymbol{p}'_{j^*} \rangle\right) + \sum_{o\in\mathcal{O}} \exp\left(\langle \boldsymbol{p}_i, \boldsymbol{o} \rangle\right)}. \tag{11}$$

By contrasting the positive object pair, *i.e.*, (p_i, p'_{j^*}), against negative ones, *i.e.*, $\{(p_i, o)\}_{o\in\mathcal{O}}$, features of same object instances in different video frames are forced to be aligned. In this way, our training framework introduces the object-level consistency property into the visual embedding space κ of matching-based segmentation models, facilitating the disco-

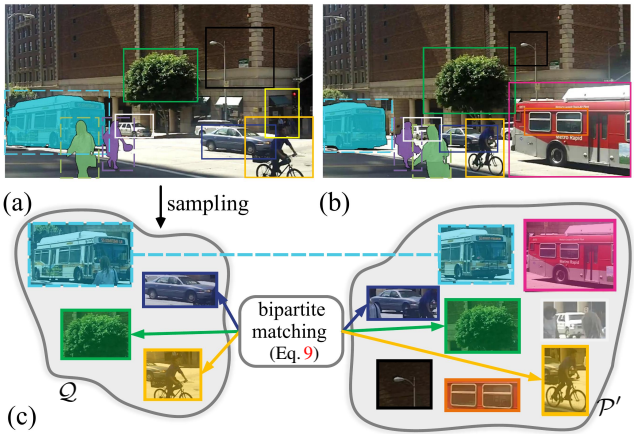


Figure 5. (a-b) Frames I_t and $I_{t'}$ with their corresponding object sets, *i.e.*, $\mathcal{P}$ and $\mathcal{P}'$, where the manually annotated object instances are plotted in dashed boxes and the object proposals discovered by Selective Search [61] are plotted in solid boxes. (c) Bipartite matching (Eq. 9) is made between a subset of $\mathcal{P}$, *i.e.*, $\mathcal{Q}\subset\mathcal{P}$, and the full object set $\mathcal{P}'$, so as to find paired objects in I_t and $I_{t'}$. The object pairs are used as positive samples for our object-level coherence based contrastive correspondence learning (*i.e.*, Eq. 11). For clarity, negative object samples from other training videos are omitted.

*We only make bipartite matching for the automatically discovered objects, as the correspondence between the annotated ones is already known.

very of robust, object-oriented correspondence. In practice, we find that our pixel-level and object-level correspondence learning strategies (*i.e.*, Eq. 7 and Eq. 11) boost the performance *collaboratively* (see related experiments in §4.3).

3.3. Implementation Details

Network Configuration. We apply our training algorithm to two top-leading matching-based VOS models: STCN[12] and XMem [10], without architecture change. Specifically, as in [10, 12] the key encoder κ and value encoder υ are constructed with the first four residual blocks of ResNet50 [18] and ResNet18, respectively. A 3×3 convolutional layer is used to project the `res4` feature with stride 16 to either the key feature $\boldsymbol{K}$ of $C=64$ channels or the value $\boldsymbol{V}$ of $D=512$ channels. For the decoder, it first compresses the memory output $\boldsymbol{V}^q$ (*cf.* Eq. 4) to 512 channels with a residual block, and then makes gradual upsampling by $2\times$ until stride 4 with higher-resolution features from κ incorporated using skip-connections. The memory stores features of past frames for long-term modeling; details can be referred to [10, 12].

Pixel-level Correspondence Learning. For training efficiency and robustness, we consider a sparse set of features of the anchor frame I_τ, which are obtained through random sampling over I_τ with a spatially uniform grid of size 8×8, instead of using all the point features of I_τ, during the computation of Eq. 5. The sampled pixels from the same batch are utilized for contrastive learning (the ratio of positive and negative pair is $1/5\times10^4$). In addition to sampling I_τ from $\mathcal{I}$, we apply random multi-scale crop and horizontal flipping to I_t and treat the transformed frame as an anchor frame. This allows us to approach cross-frame and cross-view contrastive correspondence learning within a unified framework.

Object-level Correspondence Learning. Considering the redundancy of the proposals generated by Selective Search [61] (typically thousands of proposals per frame image), we follow the common practice in object-level self-supervised representation learning [77, 83] to keep only the proposals $P=(x,y,w,h)$ that satisfy: i) the aspect ratio w/h is between 1/3 and 3/1; and ii) the scale $w\times h$ occupies between 0.3^2 and 0.8^2 of the entire image area. Moreover, we split the lattice of the frame image into multiple 32×32 grids, and cluster proposals into the grids which the center positions (x,y) fall in. The proposals in $\mathcal{Q}$ are sampled from the object clusters in I_t (up to sampling one proposal for each cluster), to omit duplicated proposals. We empirically set $|\mathcal{Q}|=3$.

Training Objective. The final learning target is the combination of the standard VOS training objective $\mathcal{L}_{\text{SEG}}$ (*cf.* Eq. 2) and our proposed two self-supervised correspondence learning loss functions, *i.e.*, $\mathcal{L}_{\text{PCL}}$ (pixel-level; Eq. 7) and $\mathcal{L}_{\text{OCL}}$ (object-level; Eq. 11):

$$\mathcal{L}=\mathcal{L}_{\text{SEG}}+\alpha(\mathcal{L}_{\text{PCL}}+\beta\mathcal{L}_{\text{OCL}}), \tag{12}$$

where the coefficient $\alpha\in[0,0.2]$ is scheduled following a linear warmup policy and β is fixed as 0.5.

Method	S	DAVIS2017$_{val}$			DAVIS2017$_{test-dev}$		
		$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$
FEELVOS[64]	✗	71.6	69.1	74.0	57.8	55.2	60.5
SSTVOS[14]	✗	82.5	79.9	85.1	-	-	-
CFBI+[88]	✗	82.9	80.1	85.7	75.6	71.6	79.6
Joint[41]	✗	83.5	80.8	86.2	-	-	-
STCN [12]	✗	82.5	79.3	85.7	73.9	69.9	77.9
STCN+**Ours**	✗	**84.7**	**81.6**	**87.8**	**77.3**	**73.5**	**81.1**
XMem[10]	✗	84.5	-	-	79.8	-	-
XMem+**Ours**	✗	**86.1**	**82.7**	**89.5**	**81.0**	**77.3**	**84.7**
STM[47]	✓	81.8	79.2	84.3	72.2	69.3	75.2
EGMN[38]	✓	82.8	80.2	85.2	-	-	-
KMN[55]	✓	82.8	80.0	85.6	77.2	74.1	80.3
RMNet[82]	✓	83.5	81.0	86.0	75.0	71.9	78.1
LCM[21]	✓	83.5	80.5	86.5	78.1	74.4	81.8
HMMN[56]	✓	84.7	81.9	87.5	78.6	74.7	82.5
AOT[87]	✓	84.9	82.3	87.5	79.6	75.9	83.3
RDE[33]	✓	84.2	80.8	87.5	77.4	73.6	81.2
PCVOS[49]	✓	86.1	83.0	89.2	-	-	-
DeAOT[89]	✓	86.2	83.1	89.3	77.5	74.0	80.9
STCN[12]	✓	85.4	82.2	88.6	76.1	73.1	80.0
STCN+**Ours**	✓	**86.6**	**83.0**	**90.1**	**79.3**	**75.8**	**82.8**
XMem[10]	✓	86.2	82.9	89.5	81.0	77.4	84.5
XMem+**Ours**	✓	**87.7**	**84.1**	**91.2**	**82.0**	**78.3**	**85.6**

Table 1. Results on DAVIS2017$_{val}$ and DAVIS2017$_{test-dev}$ [52] (§4.2). S: if synthetic data is used for pre-training.

4. Experiment

4.1. Experimental Setup

Datasets. We give extensive experiments on three datasets.

- **DAVIS2016** [51] has 50 single-object videos that are finely labeled at 24 FPS and split into 30/20 for `train`/`val`.
- **DAVIS2017** [52] contains 60/30 multi-object videos for `train`/`val`. It also provides a `test-dev` set consisting of 30 videos with more challenging scenarios.
- **YouTube-VOS** [86] includes 3,471 videos for training and 474/507 videos for validation in the 2018/2019 split, respectively. The videos are sampled at 30 FPS and annotated per 5 frame with single or multiple objects.

Training. For fair comparison, we adopt the standard training protocol [12, 38, 47, 87], which has two phases: **First**, we pre-train the network on synthetic videos generated from static, segmentation images [9, 35, 58, 68, 93]. **Second**, the main training is made on DAVIS2017$_{train}$ and YouTube-VOS2019$_{train}$. At each training step, we sample 3 frames per video to create mini-sequences, as in [10, 12]. More training details can be found in the supplementary.

Testing. All the configurations in the testing phase are kept exactly the same as the baseline. Note that our algorithm is only applied at the training time; it renders no redundant computation load and speed delay to deployment process, equally efficient as the baseline models.

Evaluation. We follow the official evaluation protocol [51] to adopt region similarity ($\mathcal{J}$), contour accuracy ($\mathcal{F}$), and their average score ($\mathcal{J}\&\mathcal{F}_m$) for evaluation. Performance on

Method	Synthetic	DAVIS2016$_{val}$		
		$\mathcal{J}\&\mathcal{F}_m\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$
RMNet[82]	✓	88.8	88.9	88.7
STM[47]	✓	89.3	88.7	89.9
LCM[21]	✓	90.7	89.9	91.4
HMMN[56]	✓	90.9	89.6	92.0
AOT[87]	✓	91.1	90.1	92.1
RDE[33]	✓	91.1	89.7	92.5
PCVOS[49]	✓	91.9	90.8	93.0
STCN[12]	✓	91.6	90.8	92.5
STCN+**Ours**		**92.0**	**91.0**	**92.9**
XMem[10]	✓	91.5	90.4	92.7
XMem+**Ours**		**92.2**	**91.1**	**93.3**

Table 2. Results on DAVIS2016$_{val}$ [51] (§4.2).

DAVIS2017$_{test\text{-}dev}$ and YouTube-VOS2018$_{val}$ & 2019$_{val}$ is obtained by submitting the results to the official servers; the latter two sets are further reported at *seen* and *unseen* classes.

4.2. Comparison to State-of-the-Arts

DAVIS2016 [51]. As demonstrated in Table 2, our approach makes stable performance gains over STCN (91.6% → **92.0%**) and XMem (91.5%→**92.2%**) on DAVIS2016$_{val}$, and outperforms all the previous state-of-the-arts. Such results are particularly impressive, considering DAVIS2016 is a simple yet extensively studied dataset.

DAVIS2017 [52]. Table 1 reports the comparison results on DAVIS2017$_{val}$&2017$_{test-dev}$. Our approach yields impressive results. Specifically, without synthetic video pre-training, our approach boosts the performance of STCN [12] by a solid margin (*i.e.*, 82.5% → **84.7%** on `val`, 73.9% → **77.3%** on `test-dev`), in terms of $\mathcal{J}\&\mathcal{F}_m$. Similarly, our approach improves the $\mathcal{J}\&\mathcal{F}_m$ of XMem [10] by **1.6%** and **1.2%**, on `val` and `test-dev`, respectively. With the aid of synthetic video data, our approach based on STCN defeats all the competitors. Notably, on the top of XMem, our approach further pushes the state-of-the-arts forward, *i.e.*, **87.7%** $\mathcal{J}\&\mathcal{F}_m$ on `val` and **82.0%** $\mathcal{J}\&\mathcal{F}_m$ on `test-dev`.

YouTube-VOS [86]. Table 3 compares our method against several top-leading approaches on YouTube-VOS2018$_{val}$ & 2019$_{val}$.As seen, our method greatly outperforms base models, *i.e.*, STCN: **83.6%** *vs.* 81.2%, and XMem: **85.6%** *vs.* 84.3%, on 2018$_{val}$ without synthetic data. With synthetic video pre-training, our approach respectively brings STCN and XMem to **85.3%** and **86.9%** on 2018$_{val}$, as well as **85.0%** and **86.6%** on 2019$_{val}$, setting new state-of-the-arts.

Qualitative Results. Fig. 6 displays qualitative comparison results on YouTube-VOS2018$_{val}$. We can observe that, compared with the original STCN, our approach generates more stable and accurate mask-tracking results, even on challenging scenarios with fast motion or occlusion.

4.3. Diagnostic Experiment

For thorough evaluation, we conduct ablation studies on DAVIS2017$_{val}$ [52] and YouTube-VOS2018$_{val}$ [86].

Method	Synthetic	Overall	Seen		Unseen	
			$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$
YouTube-VOS2018 validation split						
SSTVOS[14]	✗	81.7	81.2	-	76.0	-
CFBI+[88]	✗	82.8	81.8	86.6	77.1	85.6
Joint[41]	✗	83.1	81.5	85.9	78.7	86.5
STCN[12]	✗	81.2	81.0	85.6	74.8	83.7
STCN+**Ours**	✗	**83.6**	**82.1**	**87.0**	**78.5**	**86.7**
XMem[10]	✗	84.3	83.9	88.8	77.7	86.7
XMem+**Ours**	✗	**85.6**	**84.9**	**89.7**	**79.0**	**87.8**
STM[47]	✓	79.4	79.7	84.2	72.8	80.9
EGMN[38]	✓	80.2	80.7	85.1	74.0	80.9
RMNet[82]	✓	81.5	82.1	85.7	75.7	82.4
LCM[21]	✓	82.0	82.2	86.7	75.7	83.4
HMMN[56]	✓	82.6	82.1	87.0	76.8	84.6
AOT[87]	✓	84.1	83.7	88.5	78.1	86.1
PCVOS[49]	✓	84.6	83.0	88.0	79.6	87.9
STCN[12]	✓	83.0	81.9	86.5	77.9	85.7
STCN+**Ours**		**85.3**	**83.9**	**88.9**	**79.9**	**88.3**
XMem[10]	✓	85.7	84.6	89.3	80.2	88.7
XMem+**Ours**		**86.9**	**85.5**	**90.2**	**81.6**	**90.4**
YouTube-VOS2019 validation split						
KMN[55]	✓	80.0	80.4	84.5	73.8	81.4
LWL[3]	✓	81.0	79.6	83.8	76.4	84.2
SSTVOS[14]	✓	81.8	80.9	-	76.6	-
RDE[33]	✓	81.9	81.1	85.5	76.2	84.8
HMMN[56]	✓	82.5	81.7	86.1	77.3	85.0
AOT[87]	✓	84.1	83.5	88.1	78.4	86.3
PCVOS[49]	✓	84.6	82.6	87.3	80.0	88.3
STCN[12]	✓	82.7	81.1	85.4	78.2	85.9
STCN+**Ours**		**85.0**	**83.2**	**87.7**	**80.6**	**88.6**
XMem[10]	✓	85.5	84.3	88.6	80.3	88.6
XMem+**Ours**		**86.6**	**85.3**	**89.8**	**81.4**	**89.8**

Table 3. Results on YouTube-VOS2018$_{val}$ & 2019$_{val}$ [86] (§4.2).

Component-wise Analysis. We first ablate the effects of our core algorithm components, *i.e.*, pixel-level correspondence learning ($\mathcal{L}_{\text{PCL}}$, Eq. 7) and object-level correspondence learning ($\mathcal{L}_{\text{OCL}}$, Eq. 11). In Table 4, #1 row gives the results of the baseline models, *i.e.*, STCN [12] and XMem [10]; #2 and #3 rows list the performance obtained by additionally considering $\mathcal{L}_{\text{PCL}}$ and $\mathcal{L}_{\text{OCL}}$ individually; #4 row provides the scores of our full algorithm. Comparisons between baselines (#1 row) and variants with single component bonus (#2 row and #3 row) verify the efficacy of each module design. When comprehensively comparing the four rows, we can find that the best performance is acquired after combining the two components (*i.e.*, #4 row). This suggests that the two components can cooperate harmoniously and confirms the joint effectiveness of our overall algorithmic design.

Training Speed. Training speeds are also compared in Table 4. As seen, our algorithm only introduces negligible delay in the training speed (about 7%∼8%), while leveraging such a performance leap. More essentially, the adaption of our training mechanism does not affect the original inference process in both complexity and efficiency.

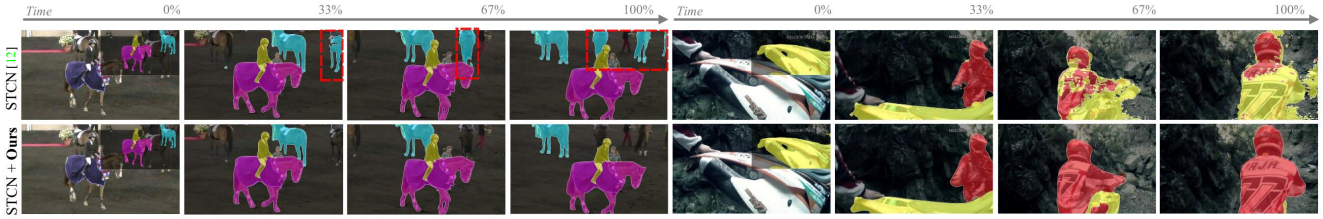


Figure 6. Qualitative results on YouTube-VOS2018$_{val}$ [86] (§4.2). The initial mask is presented in the upper right corner of the first frame.

#	$\mathcal{L}_{\text{PCL}}$ (Eq. 7)	$\mathcal{L}_{\text{OCL}}$ (Eq. 11)	STCN [12] D_{17}	STCN [12] Y_{18}	STCN [12] min/epoch	XMem [10] D_{17}	XMem [10] Y_{18}	XMem [10] min/epoch
1			85.4	83.0	2.29	86.2	85.7	3.42
2	✓		86.3	84.8	2.43	87.2	86.6	3.59
3		✓	85.9	84.0	2.39	86.9	86.3	3.53
4	✓	✓	86.6	85.3	2.50	87.7	86.9	3.66

Table 4. Analysis of essential components of our training algorithm on DAVIS2017$_{val}$ (D_{17}) [52] and YouTube-VOS2018$_{val}$ (Y_{18}) [86] (§4.3). Training speed is reported in min/epoch.

#	Negative Sample *inter*-video	Negative Sample *intra*-video	STCN [12] D_{17}	STCN [12] Y_{18}	XMem [10] D_{17}	XMem [10] Y_{18}
1			85.4	83.0	86.2	85.7
2	✓		86.1	84.5	86.9	86.4
3		✓	85.8	83.9	86.6	86.2
4	✓	✓	86.3	84.8	87.2	86.6

Table 5. Comparison of different strategies (§4.3) for sampling negative pairs during pixel-level correspondence learning ($\mathcal{L}_{\text{PCL}}$, Eq. 7).

Negative Pair Sampling for Pixel-level Correspondence Learning. During the computation of our pixel-level consistency based contrastive correspondence learning objective $\mathcal{L}_{\text{PCL}}$ (*cf.* Eq. 7), we sample non-corresponding pairs from both the current training video as well as other videos within the same training batch as negative examples. Next, we investigate the impact of such negative example sampling strategy. As indicated by Table 5, exploring both inter- and inter-video negative correspondence leads to the best performance (*i.e.*, #4 row). This is because the performance of contrastive correspondence learning heavily relies on the diversity (or quality) of the negative samples.

Object Source for Object-level Correspondence Learning. During the computation of our object-level consistency based contrastive correspondence learning objective $\mathcal{L}_{\text{OCL}}$ (*cf.* Eq. 11), we adopt Selective Search [61] to automatically generate a large set of potential object candidates, instead of only using a few annotated object instances – only occupying a small ratio of the objects in the training videos. Table 6 studies the influence of different sources of object proposals. It can be found that the best performance is achieved by exploring both manually-labeled object instances as well as massive automatically-mined object proposals as training samples. This is because the considerable number of object proposals can improve the richness of training samples, enabling robust correspondence matching.

Can VOS Benefit from Existing Self-supervised Correspondence Learning Techniques? One may be interested in if VOS can be boosted by existing correspondence learning techniques. We select three representative top-leading correspondence algorithms: reconstruction based [32], cycle-consistency based [24], and contrastive learning based [1], and apply them to STCN [12]. The results are summarized in Table 7. As seen, little or even negative performance gain is obtained. It is possibly because: **i)** none of them adopts object-level matching, but VOS is *object-aware*; **ii)** their training strategy is relatively simple (*e.g.*, color space reconstruction [32], *within*-video cycle-tracking [24]), tending to overemphasize low-level cues; **iii)** [85] learns correspondence based on frame-level similarity, which may be sub-optimal for the scenarios involving multiple objects. In contrast, our method incorporates pixel- and object-level correspondence learning simultaneously, facilitating more VOS-aligned training.

#	Object Source *manu.* annotated	Object Source *auto.* discovered	STCN [12] D_{17}	STCN [12] Y_{18}	XMem [10] D_{17}	XMem [10] Y_{18}
1			85.4	83.0	86.2	85.7
2	✓		85.7	83.5	86.5	86.1
3		✓	85.8	83.7	86.7	86.0
4	✓	✓	85.9	84.0	86.9	86.3

Table 6. Comparison of different sources of objects (§4.3), for object-level correspondence learning ($\mathcal{L}_{\text{OCL}}$, Eq. 11).

Dataset	VOS model [12]	LIIR [32]	CRW [24]	VFS [85]	**Ours**
DAVIS2017$_{val}$	85.4	85.6	85.5	85.8	**86.6**
YouTube2018$_{val}$	83.0	83.4	83.3	83.9	**85.3**

Table 7. Comparison with self-supervised corresponding learning methods on DAVIS2017$_{val}$ [52] and YouTube-VOS2018$_{val}$ [86].

5. Conclusion

Recent efforts in matching-based VOS are dedicated to more powerful model designs, while neglecting the absence of explicit supervision signals for space-time correspondence matching, which yet is the heart of the whole system. Noticing this, we take the lead in incorporating self-constrained correspondence training target with matching-based VOS models, begetting a new training mechanism for fully-supervised VOS and boosting excellent performance in a portable manner. It enjoys several charms: **i)** no modification on network architecture, **ii)** no extra annotation budget, and **iii)** no inference time delay and efficiency burden.

Acknowledgements This work was supported by "the Fundamental Research Funds for the Central Universities", 111 Project, China under Grants B07022 and (Sheitc) 150633, and Shanghai Key Laboratory of Digital Media Processing and Transmissions, China.

References

[1] Nikita Araslanov, Simone Schaub-Meyer, and Stefan Roth. Dense unsupervised learning for video segmentation. In *NeurIPS*, 2021. 2, 3, 8

[2] Linchao Bao, Baoyuan Wu, and Wei Liu. Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf. In *CVPR*, 2018. 2

[3] Goutam Bhat, Felix Järemo Lawin, Martin Danelljan, Andreas Robinson, Michael Felsberg, Luc Van Gool, and Radu Timofte. Learning what to learn for video object segmentation. In *ECCV*, 2020. 2, 7

[4] Zhangxing Bian, Allan Jabri, Alexei A Efros, and Andrew Owens. Learning pixel trajectories with multiscale contrastive random walks. In *CVPR*, 2022. 3

[5] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *CVPR*, 2017. 1, 2

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 3, 5

[7] Xi Chen, Zuoxin Li, Ye Yuan, Gang Yu, Jianxin Shen, and Donglian Qi. State-aware tracker for real-time video object segmentation. In *CVPR*, 2020. 2

[8] Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *CVPR*, 2018. 2

[9] Ho Kei Cheng, Jihoon Chung, Yu-Wing Tai, and Chi-Keung Tang. Cascadepsp: Toward class-agnostic and very high-resolution segmentation via global and local refinement. In *CVPR*, 2020. 6

[10] Ho Kei Cheng and Alexander G. Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *ECCV*, 2022. 1, 2, 3, 6, 7, 8

[11] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. In *CVPR*, 2021. 2

[12] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. In *NeurIPS*, 2021. 1, 2, 3, 6, 7, 8

[13] Jingchun Cheng, Yi-Hsuan Tsai, Wei-Chih Hung, Shengjin Wang, and Ming-Hsuan Yang. Fast and accurate online video object segmentation via tracking parts. In *CVPR*, 2018. 2

[14] Brendan Duke, Abdalla Ahmed, Christian Wolf, Parham Aarabi, and Graham W Taylor. Sstvos: Sparse spatiotemporal transformers for video object segmentation. In *CVPR*, 2021. 6, 7

[15] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *ICCV*, 2021. 3

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. In *NeurIPS*, 2020. 3

[17] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 3, 5

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[19] Olivier J Hénaff, Skanda Koppula, Jean-Baptiste Alayrac, Aaron Van den Oord, Oriol Vinyals, and João Carreira. Efficient visual pretraining with contrastive detection. In *CVPR*, 2021. 3

[20] Olivier J Hénaff, Skanda Koppula, Evan Shelhamer, Daniel Zoran, Andrew Jaegle, Andrew Zisserman, João Carreira, and Relja Arandjelović. Object discovery and representation networks. In *ECCV*, 2022. 3

[21] Li Hu, Peng Zhang, Bang Zhang, Pan Pan, Yinghui Xu, and Rong Jin. Learning position and target consistency for memory-based video object segmentation. In *CVPR*, 2021. 2, 6, 7

[22] Ping Hu, Gang Wang, Xiangfei Kong, Jason Kuen, and Yap-Peng Tan. Motion-guided cascaded refinement network for video object segmentation. In *CVPR*, 2018. 2

[23] Yuan-Ting Hu, Jia-Bin Huang, and Alexander Schwing. Maskrnn: Instance level video object segmentation. In *NeurIPS*, 2017. 2

[24] Allan Jabri, Andrew Owens, and Alexei A Efros. Space-time correspondence as a contrastive random walk. In *NeurIPS*, 2020. 2, 8

[25] Won-Dong Jang and Chang-Su Kim. Online video object segmentation via convolutional trident network. In *CVPR*, 2017. 2

[26] Sangryul Jeon, Dongbo Min, Seungryong Kim, and Kwanghoon Sohn. Mining better samples for contrastive learning of temporal correspondence. In *CVPR*, 2021. 2

[27] Minseon Kim, Jihoon Tack, and Sung Ju Hwang. Adversarial self-supervised contrastive learning. In *NeurIPS*, 2020. 2

[28] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955. 5

[29] Zihang Lai, Erika Lu, and Weidi Xie. Mast: A memory-augmented self-supervised tracker. In *CVPR*, 2020. 2

[30] Zihang Lai and Weidi Xie. Self-supervised learning for video correspondence flow. In *BMVC*, 2019. 2

[31] Jiacheng Li, Chang Chen, and Zhiwei Xiong. Contextual outpainting with object-level contrastive learning. In *CVPR*, 2022. 3

[32] Liulei Li, Tianfei Zhou, Wenguan Wang, Lu Yang, Jianwu Li, and Yi Yang. Locality-aware inter-and intra-video reconstruction for self-supervised correspondence learning. In *CVPR*, 2022. 2, 8

[33] Mingxing Li, Li Hu, Zhiwei Xiong, Bang Zhang, Pan Pan, and Dong Liu. Recurrent dynamic embedding for video object segmentation. In *CVPR*, 2022. 2, 6, 7

[34] Xueting Li, Sifei Liu, Shalini De Mello, Xiaolong Wang, Jan Kautz, and Ming-Hsuan Yang. Joint-task self-supervised learning for temporal correspondence. In *NeurIPS*, 2019. 2

[35] Xiang Li, Tianhan Wei, Yau Pun Chen, Yu-Wing Tai, and Chi-Keung Tang. Fss-1000: A 1000-class dataset for few-shot segmentation. In *CVPR*, 2020. 6

[36] Yu Li, Zhuoran Shen, and Ying Shan. Fast video object segmentation using the global context module. In *ECCV*, 2020. 2

[37] Yongqing Liang, Xin Li, Navid Jafari, and Qin Chen. Video object segmentation with adaptive feature bank and uncertain-region refinement. In *NeurIPS*, 2020. 2

[38] Xiankai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. Video object segmentation with episodic graph memory networks. In *ECCV*, 2020. 2, 6, 7

[39] Xiankai Lu, Wenguan Wang, Jianbing Shen, Yu-Wing Tai, David J Crandall, and Steven CH Hoi. Learning video object segmentation from unlabeled videos. In *CVPR*, 2020. 2

[40] K-K Maninis, Sergi Caelles, Yuhua Chen, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. Video object segmentation without temporal information. *IEEE TPAMI*, 41(6):1515–1530, 2018. 2

[41] Yunyao Mao, Ning Wang, Wengang Zhou, and Houqiang Li. Joint inductive and transductive learning for video object segmentation. In *ICCV*, 2021. 2, 6, 7

[42] Nicolas Märki, Federico Perazzi, Oliver Wang, and Alexander Sorkine-Hornung. Bilateral space video segmentation. In *CVPR*, 2016. 2

[43] Tim Meinhardt and Laura Leal-Taixé. Make one-shot video object segmentation efficient again. In *NeurIPS*, 2020. 2

[44] Jiaxu Miao, Yunchao Wei, and Yi Yang. Memory aggregation networks for efficient interactive video object segmentation. In *CVPR*, 2020. 2

[45] Pedro O O Pinheiro, Amjad Almahairi, Ryan Benmalek, Florian Golemo, and Aaron C Courville. Unsupervised learning of dense visual representations. In *NeurIPS*, 2020. 3

[46] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *CVPR*, 2018. 2

[47] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *ICCV*, 2019. 2, 3, 6, 7

[48] Hyojin Park, Jayeon Yoo, Seohyeong Jeong, Ganesh Venkatesh, and Nojun Kwak. Learning dynamic network using a reuse gate function in semi-supervised video object segmentation. In *CVPR*, 2021. 2

[49] Kwanyong Park, Sanghyun Woo, Seoung Wug Oh, In So Kweon, and Joon-Young Lee. Per-clip video object segmentation. In *CVPR*, 2022. 2, 6, 7

[50] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *CVPR*, 2017. 2

[51] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016. 6, 7

[52] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 2, 5, 6, 7, 8

[53] Andreas Robinson, Felix Jaremo Lawin, Martin Danelljan, Fahad Shahbaz Khan, and Michael Felsberg. Learning fast and robust target models for video object segmentation. In *CVPR*, 2020. 2

[54] Ramprasaath R Selvaraju, Karan Desai, Justin Johnson, and Nikhil Naik. Casting your model: Learning to localize improves self-supervised representations. In *CVPR*, 2021. 3

[55] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *ECCV*, 2020. 2, 6, 7

[56] Hongje Seong, Seoung Wug Oh, Joon-Young Lee, Seongwon Lee, Suhyeon Lee, and Euntai Kim. Hierarchical memory matching network for video object segmentation. In *CVPR*, 2021. 2, 6, 7

[57] Gopal Sharma, Kangxue Yin, Subhransu Maji, Evangelos Kalogerakis, Or Litany, and Sanja Fidler. Mvdecor: Multi-view dense correspondence learning for fine-grained 3d segmentation. In *ECCV*, 2022. 2

[58] Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *IEEE TPAMI*, 38(4):717–729, 2015. 6

[59] Jeany Son. Contrastive learning for space-time correspondence via self-cycle consistency. In *CVPR*, 2022. 2

[60] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *NeurIPS*, 2020. 3

[61] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013. 5, 6, 8

[62] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *ICCV*, 2021. 3

[63] Carles Ventura, Miriam Bellver, Andreu Girbau, Amaia Salvador, Ferran Marques, and Xavier Giro-i Nieto. Rvos: End-to-end recurrent network for video object segmentation. In *CVPR*, 2019. 2

[64] Paul Voigtlaender, Yuning Chai, Florian Schroff, Hartwig Adam, Bastian Leibe, and Liang-Chieh Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *CVPR*, 2019. 2, 6

[65] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for video object segmentation. In *BMVC*, 2017. 2

[66] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *ECCV*, 2018. 2

[67] Haochen Wang, Xiaolong Jiang, Haibing Ren, Yao Hu, and Song Bai. Swiftnet: Real-time video object segmentation. In *CVPR*, 2021. 2

[68] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, 2017. 6

[69] Ning Wang, Yibing Song, Chao Ma, Wengang Zhou, Wei Liu, and Houqiang Li. Unsupervised deep tracking. In *CVPR*, 2019. 2

[70] Ning Wang, Wengang Zhou, and Houqiang Li. Contrastive transformation for self-supervised correspondence learning. In *AAAI*, 2021. 2

[71] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast online object tracking and segmentation: A unifying approach. In *CVPR*, 2019. 2

[72] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Selective video object cutout. *IEEE TIP*, 26(12):5645–5655, 2017. 1

[73] Wenguan Wang, Jianbing Shen, Fatih Porikli, and Ruigang Yang. Semi-supervised video object segmentation with super-trajectories. *IEEE TPAMI*, 41(4):985–998, 2018. 1

[74] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. In *ICCV*, 2021. 3

[75] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, 2019. 2

[76] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, 2021. 3

[77] Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning pretraining for detection via object-level contrastive learning. In *NeurIPS*, 2021. 3, 6

[78] Longyin Wen, Dawei Du, Zhen Lei, Stan Z Li, and Ming-Hsuan Yang. Jots: Joint online tracking and segmentation. In *CVPR*, 2015. 2

[79] Ruizheng Wu, Huaijia Lin, Xiaojuan Qi, and Jiaya Jia. Memory selection network for video propagation. In *ECCV*, 2020. 2

[80] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 3

[81] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. In *ICCV*, 2021. 3

[82] Haozhe Xie, Hongxun Yao, Shangchen Zhou, Shengping Zhang, and Wenxiu Sun. Efficient regional memory network for video object segmentation. In *CVPR*, 2021. 2, 6, 7

[83] Jiahao Xie, Xiaohang Zhan, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Unsupervised object-level representation learning from scene images. In *NeurIPS*, 2021. 3, 6

[84] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *CVPR*, 2021. 3

[85] Jiarui Xu and Xiaolong Wang. Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. In *ICCV*, 2021. 2, 8

[86] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *ECCV*, 2018. 2, 5, 6, 7, 8

[87] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. In *NeurIPS*, 2021. 6, 7

[88] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by multi-scale foreground-background integration. *IEEE TPAMI*, 44:4701–4712, 2021. 2, 6, 7

[89] Zongxin Yang and Yi Yang. Decoupling features in hierarchical propagation for video object segmentation. In *NeurIPS*, 2022. 6

[90] Junbo Yin, Dingfu Zhou, Liangjun Zhang, Jin Fang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. Proposalcontrast: Unsupervised pre-training for lidar-based 3d object detection. In *ECCV*, 2022. 3

[91] Ye Yu, Jialin Yuan, Gaurav Mittal, Li Fuxin, and Mei Chen. Batman: Bilateral attention transformer in motion-appearance neighboring space for video object segmentation. In *ECCV*, 2022. 2

[92] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021. 3

[93] Yi Zeng, Pingping Zhang, Jianming Zhang, Zhe Lin, and Huchuan Lu. Towards high-resolution salient object detection. In *ICCV*, 2019. 6

[94] Lu Zhang, Zhe Lin, Jianming Zhang, Huchuan Lu, and You He. Fast video object segmentation via dynamic targeting network. In *ICCV*, 2019. 2

[95] Zixu Zhao, Yueming Jin, and Pheng-Ann Heng. Modelling neighbor relation in joint space-time graph for video correspondence learning. In *CVPR*, 2021. 2

[96] Tianfei Zhou, Fatih Porikli, David J Crandall, Luc Van Gool, and Wenguan Wang. A survey on deep learning technique for video segmentation. *IEEE TPAMI*, 2022. 1, 2

[97] Rui Zhu, Bingchen Zhao, Jingen Liu, Zhenglong Sun, and Chang Wen Chen. Improving contrastive learning by visualizing feature transformation. In *ICCV*, 2021. 3