Learning Video Object Segmentation from Unlabeled Videos

Xiankai Lu[1], Wenguan Wang[2]*, Jianbing Shen[1], Yu-Wing Tai[3], David Crandall[4], Steven C. H. Hoi[5,6]
[1] Inception Institute of Artificial Intelligence, UAE [2] ETH Zurich, Switzerland
[3] Tencent [4] Indiana University, USA [5] Salesforce Research Asia, Singapore [6] Singapore Management University, Singapore
carrierlxk@gmail.com, wenguanwang.ai@gmail.com
https://github.com/carrierlxk/MuG

Abstract

We propose a new method for video object segmentation (VOS) that addresses object pattern learning from unlabeled videos, unlike most existing methods which rely heavily on extensive annotated data. We introduce a unified unsupervised/weakly supervised learning framework, called MuG, that comprehensively captures intrinsic properties of VOS at multiple granularities. Our approach can help advance understanding of visual patterns in VOS and significantly reduce annotation burden. With a carefully-designed architecture and strong representation learning ability, our learned model can be applied to diverse VOS settings, including object-level zero-shot VOS, instance-level zero-shot VOS, and one-shot VOS. Experiments demonstrate promising performance in these settings, as well as the potential of MuG in leveraging unlabeled data to further improve the segmentation accuracy.

1. Introduction

Video object segmentation (VOS) has two common settings, zero-shot and one-shot. Zero-shot VOS (Z-VOS)[1] is to automatically segment out the primary foreground objects, without any test-time human supervision, whereas one-shot VOS (O-VOS) focuses on extracting the human determined foreground objects, typically assuming the first-frame annotations are given ahead inference[1]. Current leading methods for both Z-VOS and O-VOS are *supervised* deep learning models that require extensive amounts of elaborately annotated data to improve the performance and avoid over-fitting. However, obtaining pixel-wise segmentation labels is labor-intensive and expensive (Fig. 1 (a)).

It is thus attractive to design VOS models that can learn from unlabeled videos. With this aim in mind, we develop a unified, *unsupervised/weakly supervised* VOS method that mines *multi-granularity* cues to facilitate video object pattern learning (Fig. 1 (b)). This allows us to take advantage of nearly infinite amounts of video data. Below we give a more formal description of our problem setup and main idea.

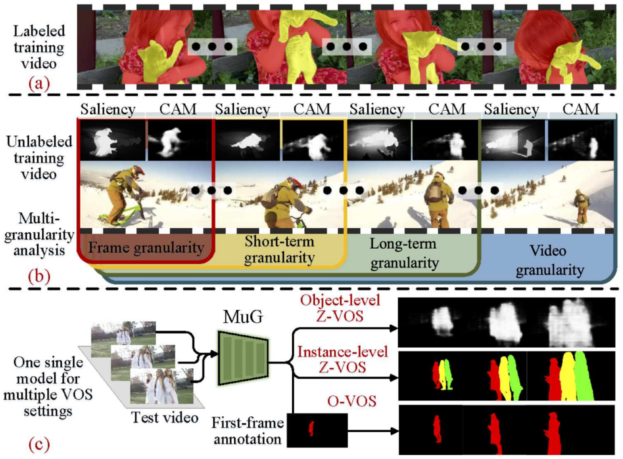

Figure 1: (a) Current leading VOS methods are learned in a supervised manner, requiring large-scale elaborately labeled data. (b) Our model, MuG, provides an unsupervised/weakly-supervised framework that learns video object patterns from unlabeled videos. (c) Once trained, MuG can be applied to diverse VOS settings, with strong modeling ability and high generability.

Problem Setup and Main Idea. Let $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ denote the input video space and output VOS space, respectively. Deep learning based VOS solutions seek to learn a differentiable, *ideal* video-to-segment mapping $g^*\colon\boldsymbol{\mathcal{X}}\mapsto\boldsymbol{\mathcal{Y}}$. To approximate g^*, recent leading VOS models typically work in a *supervised learning* manner, requiring N input samples and their desired outputs $y_n := g^*(x_n)$, where $\{(x_n, y_n)\}_n \subset \boldsymbol{\mathcal{X}} \times \boldsymbol{\mathcal{Y}}$. In contrast, we address the problem in settings with much less supervision: (1) the *unsupervised* case, when we only have samples drawn from $\boldsymbol{\mathcal{X}}$, $\{x_n\}_n \subset \boldsymbol{\mathcal{X}}$, and want to approximate g^*, and (2) the *weakly supervised learning* setting, in which we have annotations for $\boldsymbol{\mathcal{K}}$, which is a related output domain for which obtaining annotations is easier

*Corresponding author: *Wenguan Wang*.

[1] Some conventions [36, 59] also use 'unsupervised VOS' and 'semi-supervised VOS' to name the Z-VOS and O-VOS settings [3]. In this work, for notational clarity, the terms 'supervised', 'weakly supervised' and 'unsupervised' are only used to address the different learning paradigms.

than $\mathcal{Y}$, and we approximate g^* using samples from $\mathcal{X}\times\mathcal{K}$.

The standard way of evaluating learning outcomes follows an *empirical* risk/loss minimization formulation [43]:

$$\tilde{g} \in \arg\min_{g\in\mathcal{G}} \frac{1}{N}\sum\nolimits_n \varepsilon(g(x_n), z(x_n)), \tag{1}$$

where $\mathcal{G}$ denotes the hypothesis (solution) space, and $\varepsilon:\mathcal{X}\times\mathcal{Y}\mapsto\mathbb{R}$ is an error function that evaluates the estimate $g(x_n)$ against VOS-related prior knowledge $z(x_n)\in\mathcal{Z}$. To make $\tilde{g}$ a good approximation of g^*, current supervised VOS methods directly use the desired output y_n, *i.e.*, $z(x_n) := g^*(x_n)$, as the prior knowledge, with the price of vast amounts of well-annotated data.

In our method, the prior knowledge $\mathcal{Z}$, in the unsupervised learning setting, is built upon several heuristics and intrinsic properties of VOS itself, while in the weakly supervised learning setting, it additionally considers a related, easily-annotated output domain $\mathcal{K}$. For example, part of the fore-background knowledge could be from a saliency model [70] (Fig. 1 (b)), or in a form of CAM maps [73, 76] from a pre-trained image classifier [14] (*i.e.*, a related image classification domain $\mathcal{K}$)[2]. Exploring VOS in an unsupervised or weakly supervised setting is appealing not only because it alleviates the annotation burden of $\mathcal{Y}$, but also because it inspires an in-depth understanding of the nature of VOS by exploring $\mathcal{Z}$. Specifically, we analyze several different types of cues at multiple granularities, which are crucial for video object pattern modeling:

- At the *frame* granularity, we leverage information from an unsupervised saliency method [70] or CAM [73, 76] activation maps to enhance the foreground and background discriminability of our intra-frame representation.
- At the *short-term* granularity, we impose local consistency within the representations of short video clips, to describe the continuous and coherent visual patterns within a few seconds.
- At the *long-range* granularity, we address semantic correspondence among distant frames, which makes the cross-frame representations robust to local occlusions, appearance variations and shape deformations.
- At the *whole-video* granularity, we encourage the video representation to capture global and compact video content, by learning to aggregate multi-frame information and be discriminative to other videos' representations.

All these constraints are formulated under a unified, multi-granularity VOS (MuG) framework, which is fully differentiable and allows unsupervised/weakly supervised video object pattern learning, from unlabeled videos. Our extensive experiments over various VOS settings, *i.e.*, object-level Z-VOS, instance-level Z-VOS, and O-VOS, show that MuG outperforms other unsupervised and weakly supervised methods by a large margin, and continuously improves its performance with more unlabeled data.

2. Related Work

2.1. Video Object Segmentation

Z-VOS. As there is no indication for objects to be segmented, conventional ZVOS methods resorted to certain heuristics, such as saliency [59, 62, 61, 7], object proposals [19, 37, 24], and discriminative motion patterns [31, 10, 33]. Recent advances have been driven by deep learning techniques, from early, relatively simple architectures, such as recurrent network [45, 32, 63], and two-stream network [6, 49, 77], to recent, more powerful designs, such as teacher-student adaption [44], neural co-attention [26] and graph neural network [58, 68].

O-VOS. As the annotations for the first frame are assumed available at the test phase, O-VOS focuses on how to accurately propagate the initial labels to subsequent frames. Traditional methods typically used optical flow based propagation strategy [29, 9, 60, 28]. Now, deep learning based solutions become the main stream, which can be broadly classified into three categories, *i.e.*, *online learning*, *propagation* and *matching* based methods. Online learning based methods [3, 55, 35] fine-tune the segmentation network for each test video on the first-frame annotations. Propagation based methods [18, 67, 71] rely on the segments of the previous frames and work in a frame-by-frame manner. Matching based methods [66, 54, 27] segment each frame according to its correspondence/matching relation to the first frame.

Typically, current deep learning based VOS solutions (both Z-VOS and O-VOS) are trained using a large amount of elaborately-annotated data for supervised learning. In contrast, the proposed method trains a VOS network from scratch using unlabeled videos. This is essential for understanding how visual recognition works in VOS and for narrowing down the annotation budget.

2.2. VOS with Unlabeled Training Videos

Learning VOS from unlabeled videos is important but under-explored. Among a few efforts, Pathak *et al.* [34] present an early attempt in this direction, which uses a modified, purely unsupervised version of [7] to generate proxy masks as pseudo annotations. In a similar spirit, some methods use heuristic segmentation masks [17] or weakly supervised location maps [23] as supervisory signals. With a broader view, some works [47, 11, 74] capitalized on untrimmed videos tagged with semantic labels. In addition to increased annotation efforts, they are hard to handle such a class-agnostic VOS setting. Recently, self-supervised video learning has been applied for O-VOS [56, 65], which imposes the learned features to capture certain constraints on local coherence, such as cross-frame color consistency [56] and temporal cycle-correspondence [65].

[2] Note that any unsupervised or weakly supervised object segmentation/saliency model can be used; saliency [70], and CAM [73, 76] are just chosen due to their popularity and relatively high performance.

Our method is distinctive in two aspects. First, it explores various intrinsic properties of videos as well as class-agnostic fore-background knowledge in a unified, multi-granularity framework, bringing a more comprehensive understanding of visual patterns in VOS. Second, it shows strong video object representation learning ability and, for the first time, it is applied to diverse VOS settings after only being trained once. This gives a new glimpse into the connections between the two most influential VOS settings.

3. Proposed Algorithm

3.1. Multi-Granularity VOS Network

For a training video $\mathcal{X} \in \boldsymbol{\mathcal{X}}$ containing T frames: $\mathcal{X} = \{X_t\}_{t=1}^{T}$, its features are specified as $\{\boldsymbol{x}_t\}_{t=1}^{T}$, obtained from a fully convolutional feature extractor φ: $\boldsymbol{x}_t = \varphi(X_t) \in \mathbb{R}^{W \times H \times C}$. Characteristics at four-granularity are explored to guide the learning of φ (Fig. 2), as follows.

Frame Granularity Analysis: Fore-background Knowledge Understanding. As φ is VOS-aware, basic fore-background knowledge is desired to be encoded. In our method, this knowledge (Fig. 1 (b)) is initially from a background prior based saliency model [70] (in an unsupervised learning setting), or in a form of CAM maps [73, 76] (in a weakly supervised learning setting).

Formally, for each frame X_t, let us denote its corresponding initial fore-background mask as $Q_t \in \{0,1\}^{W \times H}$ (*i.e.*, a binarized saliency or CAM activation map). In our frame granularity analysis, the learning of φ is guided by the supervision signals of $\{Q_t\}_{t=1}^{T}$, *i.e.*, utilizing the intra-frame information $\boldsymbol{x}_t = \varphi(X_t)$ to regress Q_t:

$$\mathcal{L}_{\text{frame}} = \mathcal{L}_{\text{CE}}(P_t, Q_t). \tag{2}$$

Here $\mathcal{L}_{\text{CE}}$ is the *cross-entropy* loss, and $P_t = \rho(\boldsymbol{x}_t)$ where ρ: $\mathbb{R}^{W \times H \times C} \mapsto [0,1]^{W \times H}$ maps the input single-frame feature $\boldsymbol{x}_t$ into a fore-background prediction map P_t. ρ is implemented by a 1×1 convolutional layer with *sigmoid* activation.

Short-Term Granularity Analysis: Intra-Clip Coherence Modeling. Short-term coherence is an essential property in videos, as temporally-close frames typically exhibit continuous visual content changes [15]. To capture this property, we apply a forward-backward patch tracking mechanism [57] which learns φ by tracking a sampled patch forwards in a few successive frames and then backwards until the start frame, and penalizing the distance between the initial and final backwards tracked positions of that patch.

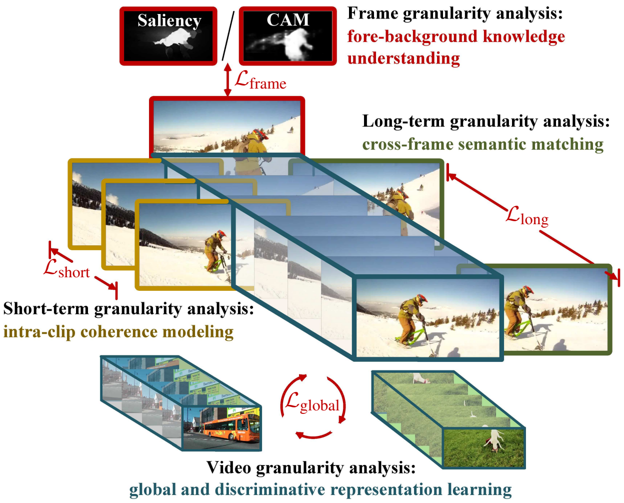
Saliency
CAM
Frame granularity analysis:
fore-background knowledge understanding
$\mathcal{L}_{\text{frame}}$
Long-term granularity analysis:
cross-frame semantic matching
$\mathcal{L}_{\text{long}}$
$\mathcal{L}_{\text{short}}$
Short-term granularity analysis:
intra-clip coherence modeling
$\mathcal{L}_{\text{global}}$
Video granularity analysis:
global and discriminative representation learning

Figure 2: Overview of our approach. Intrinsic properties over **frame**, **short-term**, **long-term** and **whole video** granularities are explored to guide the video object pattern learning.

Formally, given two consecutive frames X_t and X_{t+1}, we first crop a patch p from X_t and apply φ on p and X_{t+1}, separately. Then we obtain two feature embeddings: $\varphi(p) \in \mathbb{R}^{w \times h \times C}$ and $\boldsymbol{x}_{t+1} = \varphi(X_{t+1}) \in \mathbb{R}^{W \times H \times C}$. With a design similar to the classic Siamese tracker [2], we forward track the patch p on the next frame X_{t+1} by conducting a cross-correlation operation '$\star$' on $\varphi(p)$ and $\varphi(X_{t+1})$:

$$S_{\Rightarrow} = \varphi(p) \star \varphi(X_{t+1}) \in [0,1]^{W \times H}, \tag{3}$$

where $S_{\Rightarrow}$ is a *sigmoid*-normalized response map whose size is rescaled into (H, W). The new location of p in X_{t+1} is then inferred according to the peak value on $S_{\Rightarrow}$. After obtaining the forward tracked patch p' in X_{t+1}, we backward track p' to X_t and get a backward tracking response map $S_{\Leftarrow}$:

$$S_{\Leftarrow} = \varphi(p') \star \varphi(X_t) \in [0,1]^{W \times H}. \tag{4}$$

Ideally, the peak of $S_{\Leftarrow}$ should correspond to the location of p in the initial frame X_t. Thus we build a consistency loss that measures the alignment error between the initial and forward-backward tracked positions of p:

$$\mathcal{L}_{\text{short}} = \|S_{\Leftarrow} - G_p\|_2^2, \tag{5}$$

where $G_p \in [0,1]^{W \times H}$ is a (H, W)-dimensional Gaussian-shape map with the same center of p and variance proportional to the size of p. As in [57], the above forward-backward tracking mechanism is extended to a multi-frame setting (Fig. 3). Specifically, after obtaining the forward

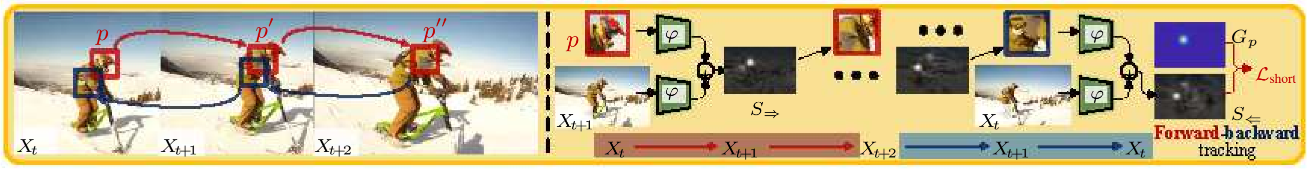

Figure 3: Left: Main idea of short-term granularity analysis. Right: Training details for intra-clip coherence modeling.

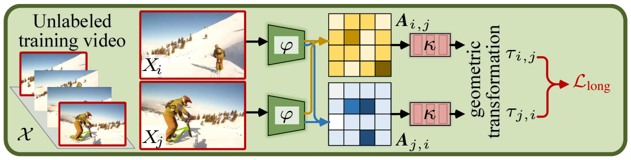

Figure 4: Illustration of our long-term granularity analysis.

tracked patch p' in X_{t+1}, p' is further tracked to the next frame X_{t+2}, and a new tracked patch p'' is obtained. Then p'' is reversely tracked to X_{t+1} and further to the initial frame X_t, and the local consistency loss in Eq. 5 is computed. Moreover, during training, we first random sample a short video clip consisting of six successive frames. Then we perform above forward-backward tracking based learning strategy over three frames random drawn from the six-frame video clip. With above designs, φ captures the spatiotemporally local correspondence and is content-discriminative (due to its cross-frame target re-identification nature).

Long-Term Granularity Analysis: Cross-Frame Semantic Matching. In addition to the local consistency among adjacent frames, there also exist strong semantic correlations among distant frames, as frames from the same video typically contain similar content [30, 69]. Capturing this property is essential for φ, as it makes φ robust to many challenges, such as appearance variation, shape deformations, object occlusions, *etc*. To address this issue, we conduct a long-term granularity analysis, which casts cross-frame correspondence learning as a dual-frame semantic matching problem (Fig. 4). Specifically, given a training pair of two *disordered* frames (X_i, X_j) randomly sampled from $\mathcal{X}$, we compute a similarity affinity $\boldsymbol{A}_{i,j}$ between their embeddings: $(\varphi(X_i), \varphi(X_j))$ by a co-attention [52]:

$$\boldsymbol{A}_{i,j} = \text{softmax}(\boldsymbol{x}_i^\top \boldsymbol{x}_j) \in [0,1]^{(WH)\times(WH)}, \tag{6}$$

where $\boldsymbol{x}_i \in \mathbb{R}^{C\times(WH)}$ and $\boldsymbol{x}_j \in \mathbb{R}^{C\times(WH)}$ are flat matrix formats of $\varphi(X_i)$ and $\varphi(X_j)$, respectively. 'softmax' indicates *column-wise softmax* normalization. Given the normalized cross-correlation $\boldsymbol{A}_{i,j}$, in line with [41], we use a small neural network $\kappa : \mathbb{R}^{(W\times H)\times(W\times H)} \mapsto \mathbb{R}^6$ to regress the parameters of a geometric transformation $\tau_{i,j}$, *i.e.*, six-degree of freedom (translation, rotation and scale). $\tau_{i,j} : \mathbb{R}^2 \mapsto \mathbb{R}^2$ gives the relations between the spatial coordinates in X_i and X_j considering the corresponding semantic similarity:

$$\boldsymbol{m}_i = \tau_{i,j}(\boldsymbol{m}_j), \tag{7}$$

where $\boldsymbol{m}_i$ is a 2-D spatial coordinate of X_i, and $\boldsymbol{m}_j$ the corresponding sampling coordinates in X_j. Using $\tau_{i,j}$, we can warp X_i to X_j. Similarly, we can also compute $\tau_{j,i}$, *i.e.*, a 2-D warping from X_j to X_i. Let us consider two sampling coordinates $\boldsymbol{m}_i$ and $\boldsymbol{n}_j$ in X_j and X_i, respectively, we introduce a semantic matching loss [41]:

$$\begin{aligned}\mathcal{L}_{\text{long}} = -\Big(&\sum\nolimits_{\boldsymbol{m}_i\in\Omega}\sum\nolimits_{\boldsymbol{o}_j\in\Omega}\boldsymbol{A}_{i,j}(\boldsymbol{m}_i,\boldsymbol{o}_j)\iota(\boldsymbol{m}_i,\boldsymbol{o}_j)+\\ &\sum\nolimits_{\boldsymbol{n}_j\in\Omega}\sum\nolimits_{\boldsymbol{o}_i\in\Omega}\boldsymbol{A}_{j,i}(\boldsymbol{n}_j,\boldsymbol{o}_i)\iota(\boldsymbol{m}_i,\boldsymbol{o}_i)\Big),\end{aligned} \tag{8}$$

where Ω refers to the image lattice, $\boldsymbol{A}_{i,j}(\boldsymbol{m}_i, \boldsymbol{o}_j) \in [0,1]$ gives the similarity value between the positions $\boldsymbol{m}_i$ and $\boldsymbol{o}_j$ in X_i and X_j, and $\iota(\boldsymbol{m}_i, \boldsymbol{o}_j)$ determines if the correspondence between $\boldsymbol{m}_i$ and $\boldsymbol{o}_j$ is geometrically consistent. If $||\boldsymbol{m}_i, \tau_{i,j}(\boldsymbol{o}_j)|| \leq 1$, $\iota{=}1$; otherwise $\iota{=}0$.

Video Granularity Analysis: Global and Discriminative Representation Learning. So far, we have used the pairwise cross-frame information in local and long terms to boost the learning of φ. φ is also desired to learn a compact and globally discriminative video representation. To achieve this, we use a *global information aggregation* module which performs video granularity analysis within an unsupervised video embedding learning framework [1] to leverage supervision signals from different videos.

Starting with our global information aggregation module, we split $\mathcal{X}{=}\{X_t\}_{t=1}^T$ into K segments of equal durations: $\mathcal{X}{=}\cup_{k=1}^K \mathcal{X}_k$. For each segment $\mathcal{X}_k$, we randomly sample a single frame, resulting in a K-frame abstract $\mathcal{X}'{=}\{X_{t_k}\}_{k=1}^K$ of $\mathcal{X}$. $\mathcal{X}'$ reduces the redundancy among successive frames while preserving global information.

With a similar spirit of *key-value* retrieval networks [46], for each $X_{t_k} \in \mathcal{X}'$, we set it as a *query* and the remaining frames $\mathcal{X}'/X_{t_k}$ as *reference*. Then we compute the normalized cross-correlation between the query and reference:

$$\boldsymbol{A}_{t_k}{=}\text{softmax}(\boldsymbol{x}_{t_k}^\top[\{\boldsymbol{x}_{t_{k'}}\}_{t_{k'}}]) \in [0,1]^{(WH)\times(WH(K-1))}, \tag{9}$$

where $k' \in \{1, \cdots, K\}/k$, and '[·]' denotes the concatenation operation. $\boldsymbol{x}_{t_k} \in \mathbb{R}^{C\times(WH)}$ and $[\{\boldsymbol{x}_{t_{k'}}\}_{t_{k'}\in\{1,\cdots,K\}/k}] \in \mathbb{R}^{C\times(WH(K-1))}$ are flat feature matrices of the query and reference, respectively. Subsequently, $\boldsymbol{A}_{t_k}$ is used as a weight matrix for global information summarization:

$$\boldsymbol{x}'_{t_k} = [\{\boldsymbol{x}_{t_{k'}}\}_{t_{k'}}]\,\boldsymbol{A}_{t_k}^\top \in \mathbb{R}^{(WH)\times C}, \quad \text{where } k' \in \{1, \cdots, K\}/k. \tag{10}$$

Our global information aggregation module gathers information from the reference set with a correlation-based feature summarization procedure. For query frame X_{t_k}, we obtain a global information augmented representation:

$$\boldsymbol{r}_{t_k} = [\boldsymbol{x}'_{t_k}, \boldsymbol{x}_{t_k}] \in \mathbb{R}^{W\times H\times 2C}. \tag{11}$$

During training, the video granularity analysis essentially discriminates between a set of surrogate video classes [1]. Specifically, given N training videos, we randomly sample a single frame from each video, leading to N training *instances*: $\{X^n\}_{n=1}^N$. The core idea is that, for a query frame $X_{t_k}^n$ in the n-th video, its global feature embedding is close to the instance X^n from the same n-th video, and far from other unrelated instances $\{X^{n'}\}_{n'\neq n}$ (from the other $N{-}1$ videos). We solve this as a binary classification problem via maximum likelihood estimation (MLE). In particular, for $X_{t_k}^n$, instance X^n should be classified into n, while other instances $\{X^{n'}\}_{n'\neq n}$ shouldn't be. The probability of X^n being recognized as instance n is:

$$P(n|X^n) = \frac{\exp(\text{GAP}(\boldsymbol{r}_{t_k}^{n\top}\boldsymbol{r}^n))}{\sum_{i=1}^{N}\exp(\text{GAP}(\boldsymbol{r}_{t_k}^{n\top}\boldsymbol{r}^i))}. \tag{12}$$

where 'GAP' stands for *global average pooling*. Similarly, given $X_{t_k}^n$, the probability of other instances $X^{n'}$ be recognized as instance n is:

$$P(n|X^{n'}) = \frac{\exp(\text{GAP}(\boldsymbol{r}_{t_k}^{n\top}\boldsymbol{r}^{n'}))}{\sum_{i=1}^{N}\exp(\text{GAP}(\boldsymbol{r}_{t_k}^{n\top}\boldsymbol{r}^i))}. \tag{13}$$

Correspondingly, the probability of $X^{n'}$ not being recognized as instance n is $1-P(n|X^{n'})$. The joint probability of X^n being recognized as instance n and $X^{n'}$ not being is: $P(n|X^n)\prod_{n'\neq n}(1-P(n|X^{n'}))$, under the assumption that different instances being recognized as n are independent.

Then the loss function is defined as the negative log likelihood over N query frames from N videos:

$$\mathcal{L}_{\text{global}} = -\sum_{n}\log P(n|X^n) - \sum_{n}\sum_{n'\neq n}\log(1-P(n|X^{n'})). \tag{14}$$

3.2. One Training Phase for both Z-VOS and O-VOS

We now describe the network architecture during the training and inference phases. An appealing advantage of our multi-granularity VOS network is that, after being trained in a unified mode, it can be directly applied to both Z-VOS and O-VOS settings with only slight adaption.

Network Architecture. Our whole module is end-to-end trainable. The video representation space φ is learned by a fully convolutional network, whose design is inspired by ResNet-50[13]. In particular, the first four groups of convolutional layers in ResNet are preserved and dilated convolutional layer[72] is used to maintain enough spatial details as well as ensure a large receptive field, resulting in a 512-channel feature representation $\boldsymbol{x}$ whose spatial dimensions are $1/4$ of an input video frame X.

During training, we use a mini-batch of $N=16$ videos and scale all the training frames into 256×256 pixels. For frame granularity analysis, all the frames access to the supervision signal from the loss $\mathcal{L}_{\text{frame}}$ in Eq. 2.

For short-term granularity analysis, six successive video frames are first randomly sampled from each training video, resulting in a six-frame video clip. For each video clip, we further sample three video frames orderly and randomly crop a 64×64 patch as p. With the feature embedding $\varphi(p)\in\mathbb{R}^{16\times16\times64}$ of p, we forward-backward track p and get its final backward tracking response map $S_{\Leftarrow}\in[0,1]^{64\times64}$ via Eq. 4. For computing the loss in Eq. 5, the Gaussian-shape map $G_p\in[0,1]^{64\times64}$ is obtained by convolving the center position of p with a two-dimension Gaussian map with a kernel width proportional (0.1) to the patch size.

For long-term granularity analysis, after randomly sampling two *disordered* frames (X_i, X_j) $(|i-j|\geq6)$ from a training video $\mathcal{X}$, we compute the correlation map $\boldsymbol{A}_{i,j}\in[0,1]^{(64\times64)\times(64\times64)}$ by the normalized inner production operation in Eq. 6. For the geometric transformation parameter estimator $\kappa:\mathbb{R}^{(64\times64)\times(64\times64)}\mapsto\mathbb{R}^6$, it is achieved by two convolutional layers and one linear layer, as in [41]. Then the semantic matching loss in Eq. 8 is computed.

For video granularity analysis, we split each training video $\mathcal{X}$ into $K=8$ segments, and get the global information augmented representation $\boldsymbol{r}_{t_k}\in\mathbb{R}^{64\times64\times256}$ for each query frame X_{t_k} by Eq. 11. Then, we compute the softmax embedding learning loss using Eq. 14, which leverages supervision signals from the N training videos.

Iterative Training by Bootstrapping. As seen in Fig. 1(b), the fore-background knowledge from the saliency [70] or CAM [73, 76] is ambiguous and noisy. Inspired by Bootstrapping [40], we apply an iterative training strategy: after training with the initial fore-background maps, we use our trained model to re-label the training data. With each iteration, the learner bootstraps itself by mining better fore-background knowledge and then leading a better model. Specifically, for each training frame X, given the initial fore-background mask $Q\in\{0,1\}^{64\times64}$ and current prediction $\bar{P}^i\in\{0,1\}^{64\times64}$ of the model in i-th training iteration, the loss in Eq. 2 in $(i+1)$-th iteration is formulated in a bootstrapping format:

$$\begin{aligned}\mathcal{L}_{\text{frame}}^{(i+1)} = \sum\nolimits_{\boldsymbol{m}\in\Omega}&[\alpha Q_{\boldsymbol{m}}+(1-\alpha)\bar{P}_{\boldsymbol{m}}^i]\log(P_{\boldsymbol{m}}^{i+1})+\\ &[\alpha(1-Q_{\boldsymbol{m}})+(1-\alpha)(1-\bar{P}_{\boldsymbol{m}}^i)]\log(1-P_{\boldsymbol{m}}^{i+1}),\end{aligned} \tag{15}$$

where $\alpha=0.05$ and $Q_{\boldsymbol{m}}$ gives the value in position $\boldsymbol{m}$. In such a design, the 'confident' fore-background knowledge is generated as a convex combination of the initial fore-background information Q and model prediction P.

In the i-th training iteration, the overall loss to optimize the whole network parameters is the combination of the losses in Eq. 15, 4, 8 and 14:

$$\mathcal{L}^{(h)} = \mathcal{L}_{\text{frame}}^{(h)} + \beta_1\mathcal{L}_{\text{short}} + \beta_2\mathcal{L}_{\text{long}} + \beta_3\mathcal{L}_{\text{global}}, \tag{16}$$

where βs are coefficients: $\beta_1=0.1$, $\beta_2=0.02$ and $\beta_3=0.5$.

The above designs enable a unified un-/weakly supervised feature learning framework. Once the model is trained, the learned representations φ can be used for Z-VOS and O-VOS, with slight modifications. In practice, we find that our model can perform well after being trained with 2 iterations; please see §4.2 for related experiments.

3.3. Inference for Z-VOS and O-VOS

Now we detail our inference modes for object-level Z-VOS, instance-level Z-VOS, and O-VOS settings.

Object-Level Z-VOS Setting. For each test frame, object-level Z-VOS aims to predict a binary segmentation mask where the primary foreground objects are separated from the background while the identities of different foreground objects are not distinguished. In the classic VOS setting, since there is no any test-time human intervention, how to discover the primary video objects is the central problem. Considering the fact that interested objects frequently appear throughout the video sequence, we readout the segmentation results from the global information augmented feature $\boldsymbol{r}$, instead of directly using intra-frame information to predict the fore-background mask (*i.e.*, $\rho(\boldsymbol{x})$). This is achieved by an extra segmentation readout layer $\upsilon: \mathbb{R}^{64\times64\times256} \mapsto [0,1]^{64\times64}$, which takes the global frame embedding $\boldsymbol{r}$ as the input and produces the final object-level segmentation prediction. υ is also trained by the *cross-entropy* loss, as in Eq. 15. For notation clarity, we omit this term in the overall training loss in Eq. 16. Note that υ is only used in Z-VOS (not O-VOS, see below).

Instance-Level Z-VOS Setting. Our model can also be adapted for the instance-level Z-VOS setting, in which different object instances must be discriminated, in addition to separating the primary video objects from the background without test-time human supervision. For each test frame, we first apply mask-RCNN [12] to produce a set of category agnostic object proposals.Then we apply our trained model for producing a binary foreground-background mask per frame. After combining object bounding-box proposals with binary object-level segmentation masks, we can filter out the background proposals and obtain pixel-wise, instance-level object candidates for each frame. Finally, to link those object candidates across different frames, similar to [27], we use overlap ratio and optical flow as the cross-frame candidate-association metric. Note that, mask-RCNN can be replaced with non-learning Edgebox [78] and GrabCut, resulting a purely unsupervised/weakly-supervised protocol.

O-VOS Setting. In O-VOS, for each test video sequence, instance-level annotations regarding multiple general foreground objects in the first frame are given. In such a setting, our trained network works in a per-frame matching based mask propagation fashion. Concretely, assume there are a total of L object instances (including the background) in the first-frame annotation, each spatial position $\boldsymbol{n} \in \Omega$ will be associated with a one-hot class vector $\hat{\boldsymbol{y}}_{\boldsymbol{n}} \in \{0,1\}^L$, whose element $\hat{y}^l_{\boldsymbol{n}}$ indicates whether pixel $\boldsymbol{n}$ belong to l-th object instance. Starting from the second frame, we use both the last segmented frame X_{t-1} as well as current under-segmented frame X_t to build an input pair for our model. Then we compute their similarity affinity $\boldsymbol{A}_{t-1,t} \in [0,1]^{(64\times64)\times(64\times64)}$ in the feature space: $\boldsymbol{A}_{t-1,t} = \text{softmax}(\boldsymbol{x}_{t-1}^{\top}\boldsymbol{x}_t)$. After that, for each pixel $\boldsymbol{m}$ in X_t, we compute its probability distribution $\boldsymbol{v}_{\boldsymbol{m}} \in [0,1]^L$ over the L object instances as:

$$\boldsymbol{v}_{\boldsymbol{m}} = \sum\nolimits_{\boldsymbol{n}\in\Omega} \boldsymbol{A}_{t-1,t}(\boldsymbol{n},\boldsymbol{m}) \; \hat{\boldsymbol{y}}_{\boldsymbol{m}}, \tag{17}$$

where $\boldsymbol{A}_{t-1,t}(\boldsymbol{n},\boldsymbol{m}) \in [0,1]$ is the affinity value between pixel $\boldsymbol{n}$ in X_{t-1} and $\boldsymbol{m}$ in X_t. For $\boldsymbol{m}$, it is assigned to l^*-th instance: $l^* = \arg\max_l(\{v^l_{\boldsymbol{m}}\}^L_{l=1})$, where $\boldsymbol{v}_{\boldsymbol{m}} = [v^l_{\boldsymbol{m}}]^L_{l=1}$. Then we get its label vector $\hat{\boldsymbol{y}}_{\boldsymbol{m}}$. In this way, from the segmented frame X_t, we move to the next input frame pair (X_t, X_{t+1}) and get the segmentation result for X_{t+1}. As our method does not use any first-frame fine-tuning [6, 35] or online learning [55] technique, it is fast for inference.

Aspects	Module	Unsuper.		Weakly-super.	
		mean $\mathcal{J}$	$\Delta\mathcal{J}$	mean $\mathcal{J}$	$\Delta\mathcal{J}$
Reference	**Full model** (2 iterations)	58.0	-	61.2	-
Initial Fore-/Background Knowledge	Heuristic Saliency [70]	37.2	-20.8	-	-
	CAM [73]	-	-	45.3	-15.9
Multi-Granularity Analysis	*w/o.* Frame Granularity	40.2	-17.8	40.2	-21.0
	w/o. Short-term Granularity	51.3	-6.7	57.1	-4.1
	w/o. Long-term Granularity	52.8	-5.2	56.0	-5.2
	w/o. Video Granularity	56.4	-1.6	60.4	-0.8
Iterative Training via Bootstrapping	1 iteration	50.8	-7.2	54.9	-6.3
	3 iterations	58.0	0.0	61.2	0.0
	4 iterations	58.0	0.0	61.2	0.0
More Data	+ LaSOT dataset [8]	59.5	+1.5	62.3	+1.1
Post-Process	*w/o.* CRF	55.3	-2.7	58.7	-2.5

Table 1: **Ablation study on DAVIS$_{16}$ [36] `val` set**, under the object-level Z-VOS setting. Please see §4.2 for details.

4. Experiment

4.1. Common Setup

Implementation Details. We train the whole network from scratch on the OxUvA [51] tracking dataset, as in [22]. OxUvA comprises 366 video sequences with more than 1.5 million frames in total. We train our model with SGD optimizer. For our bootstrapping based iterative training, two iterations are used and each takes about 8 hours.

Configuration and Reproducibility. MuG is implemented on PyTorch. All experiments are conducted on an Nvidia TITAN Xp GPU and an Intel (R) Xeon E5 CPU. All our implementations, trained models, and segmentation results will be released to provide the full details of our approach.

4.2. Diagnostic Experiments

A series of ablation studies are performed for assessing the effectiveness of each essential component of MuG.

Initial Fore-Background Knowledge. Baselines *Heuristic Saliency* and *CAM* give the scores of initial fore-background knowledge, based on their CRF-binarized outputs. As seen, with the low-quality initial knowledge, our MuG gains huge performance improvements (+20.8% and +15.9% promotions), showing the significance of our multi-granularity video object pattern learning scheme.

Multi-Granularity Analysis. Next we investigate the contributions of multi-granularity cues in depth. As shown in Table 1, the intrinsic, multi-granularity properties are indeed meaningful, as disabling any granularity analysis component causes performance to erode. For instance, removing the frame granularity analysis during learning hurts performance (mean $\mathcal{J}$: $58.0 \rightarrow 40.2, 61.2 \rightarrow 40.2$), due to the lack of fore-/background information. Similarly, performance drops when excluding short- or long-term granularity analysis, suggesting the importance of capturing local consistency and semantic correspondence. Moreover, considering video granularity information also improves the final performance, proving the meaning of comprehensive video content understanding in video object pattern modeling.

Iterative Training Strategy. From Table 1, we can see that

Supervision		Non Learning						Unsupervised Learning			Weakly-supervised	
Method		TRC[10]	CVOS[48]	KEY[24]	MSG[31]	NLC[7]	FST[33]	Motion Masks[34]	TSN[17]	**Ours**	COSEG[50]	**Ours**
$\mathcal{J}$	Mean ↑	47.3	48.2	49.8	53.3	55.1	**55.8**	48.9	31.2	**58.0**	52.8	**61.2**
	Recall ↑	49.3	54.0	59.1	61.6	55.8	**64.7**	44.7	18.7	**65.3**	50.0	**74.5**
	Decay ↓	8.3	10.5	14.1	2.4	12.6	**0.0**	19.2	**-0.4**	2.0	**10.7**	11.6
$\mathcal{F}$	Mean ↑	44.1	44.7	42.7	50.8	**52.3**	51.1	39.1	18.4	**51.5**	49.3	**56.1**
	Recall ↑	43.6	52.6	37.5	60.0	**51.9**	51.6	28.6	5.6	**53.2**	52.7	**62.1**
	Decay ↓	12.9	11.7	10.6	5.1	11.4	**2.9**	17.9	**1.9**	2.1	10.5	**3.55**
$\mathcal{T}$	Mean ↓	39.1	**25.0**	26.9	30.1	42.5	36.6	36.4	37.5	**30.1**	**28.2**	58.6

Table 2: **Evaluation of object-level Z-VOS on DAVIS$_{16}$ val set[36]** (§4.3), with region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and time stability $\mathcal{T}$. (The best scores in each supervision setting are marked in **bold**. These notes are the same to other tables.)

Supervision	Non Learning				Unsupervised Learning			Weakly-supervised Learning			
Method	CRANE[47]	NLC[7]	FST[33]	ARP[19]	Motion Masks[34]	TSN[17]	**Ours**	SOSD[75]	BBF[42]	COSEG[50]	**Ours**
$\mathcal{J}$ Mean ↑	23.9	27.7	**53.8**	46.2	32.1	52.2	**57.7**	54.1	53.3	58.1	**62.4**

Table 3: **Evaluation of object-level Z-VOS on Youtube-Objects[39]** (§4.3), with mean $\mathcal{J}$. See the supplementary for more details.

with more iterations of our bootstrapping training strategy (1 → 2), better performance can be obtained. However, further iterations (2 → 4) give only marginal performance change. We thus use two iterations in all the experiments.

More Training Data. To show the potential of our unsupervised/weakly supervised VOS learning scheme, we probe the upper bound by training on additional videos. With more training data (1400 videos) from LaSOT dataset [8], performance boosts can be observed in both two settings.

4.3. Performance for Object-Level Z-VOS

Datasets. Experiments are conducted on two famous Z-VOS datasets: DAVIS[36] and Youtube-Objects[39], which have pixel-wise, object-level annotations. DAVIS$_{16}$ has 50 videos (3,455 frames), covering a wide range of challenges, such as fast motion, occlusion, dynamic background, *etc*. It is split into a train set (30 videos) and a val set (20 videos). Youtube-Objects contains 126 video sequences that belong to 10 categories (such as *cat*, *dog*, *etc*.) and has 25,673 frames in total. The val set of DAVIS$_{16}$ and whole Youtube-Objects are used for evaluation.

Evaluation Criteria. For fair comparison, we follow the official evaluation protocols of each dataset. For DAVIS$_{16}$, we report region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and time stability $\mathcal{T}$. For Youtube-Objects, the performance is evaluated in terms of region similarity $\mathcal{J}$.

Post-processing. Following the common protocol in this area[49, 45, 6], the final segmentation results are optimized by CRF[21] (about 0.3s per frame).

Quantitative Results. Table 2 presents the comparison results with several non-learning, unsupervised or weakly supervised learning competitors in DAVIS$_{16}$ dataset. MuG exceeds current leading unsupervised learning-based methods (*i.e*., Motion Masks[34] and TSN[17]) in large margins (58.0 *vs* 48.9 and 58.0 *vs* 31.2). MuG also outperforms classical weakly-supervised Z-VOS method COSEG [50], and all the previous heuristic methods. Table3 summarizes comparison results on Youtube-Objects dataset, showing again our superior performance in both unsupervised and weakly supervised learning settings.

Supervision		Fully Supervised			Unsupervised		Weakly-super.	
Method		AGS [63]	PDB [45]	RVOS [53]	Ours*	Ours	Ours*	Ours
$\mathcal{J}$&$\mathcal{F}$	Mean ↑	45.6	40.4	22.5	36.5	37.3	40.6	41.7
$\mathcal{J}$	Mean ↑	42.1	37.7	17.7	33.8	35.0	37.7	38.9
	Recall ↑	48.5	42.6	16.2	38.2	39.3	42.5	44.3
	Decay ↓	2.6	4.0	1.6	2.1	3.8	1.9	2.7
$\mathcal{F}$	Mean ↑	49.0	43.0	27.3	38.0	39.6	43.5	44.5
	Recall ↑	51.5	44.6	24.8	38.6	41.1	44.9	46.6
	Decay ↓	2.6	3.7	1.8	3.2	4.6	1.0	1.7

Table 4: **Evaluation of instance-level Z-VOS on DAVIS$_{17}$ test-dev set[4]** (§4.4), $*$ denotes purely unsupervised/weakly-supervised protocol with non-learning Edgebox [78] and GrabCut.

Runtime Comparison. The inference time of MuG is about 0.6s per frame, which is faster than most deep learning based competitors (*e.g*., MotionMask [34] (1.1s), TSN [17] (0.9s)). This is because, except CRF [21], there is no other pre-/post-processing step (*e.g*., superpixel [50], optical flow[33], *etc*.) and online fine-tuning[19].

4.4. Performance for Instance-Level Z-VOS

Datasets. We test the performance for instance-level Z-VOS on DAVIS$_{17}$ [4] dataset, which has 120 videos and 8,502 frames in total. It has three subsets, namely, train, val, and test-dev, containing 60, 30, and 30 video sequences, respectively. We use the ground-truth masks provided by the newest DAVIS challenge [4], as the original annotations are biased towards the O-VOS scenario.

Evaluation Criteria. Three standard evaluation metrics, provided by DAVIS$_{17}$, are used, *i.e*., region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and the average value of $\mathcal{J}$&$\mathcal{F}$.

Quantitative Results. Three top-performing ZVOS methods from the DAVIS$_{17}$ benchmark are included. As shown in Table 4, our model achieves comparable performance with the fully supervised methods (*i.e*., AGS [63] and PDB [45]). Notably, it significantly outperforms recent RVOS[53] (mean $\mathcal{J}$&$\mathcal{F}$: +14.8% and +19.2% in unsupervised and weakly-supervised learning setting, respectively).

Runtime Comparison. The processing time for each frame is about 0.7s which is comparable to AGS [63] and

Supervision		Non Learning					Unsupervised Learning					Weakly-supervised	
Method		HVS [29]	JMP [9]	FCP [37]	SIFT Flow [25]	BVS [28]	Vondrick *et al.* [56]	mgPFF [20]	TimeCycle [65]	CorrFlow [22]	**Ours**	FlowNet2 [16]	**Ours**
$\mathcal{J}$	Mean ↑	54.6	57.0	58.4	51.1	**60.0**	38.9	40.5	55.8	48.9	**63.1**	41.6	**65.7**
	Recall ↑	61.4	62.6	**71.5**	58.6	66.9	37.1	34.9	64.9	44.7	**71.9**	45.7	**77.6**
	Decay ↓	23.6	39.4	**-2.0**	18.8	28.9	22.4	18.8	**0.0**	19.2	28.1	**19.9**	26.4
$\mathcal{F}$	Mean ↑	52.9	53.1	49.2	44.0	**58.8**	30.8	34.0	51.1	39.1	**61.8**	40.1	**63.5**
	Recall ↑	61.0	54.2	49.5	50.3	**67.9**	21.7	24.2	51.6	28.6	**64.2**	38.3	**67.7**
	Decay ↓	22.7	38.4	**-1.1**	20.0	21.3	16.7	13.8	**2.9**	17.9	30.5	**26.6**	27.2
$\mathcal{T}$	Mean ↓	36.0	**15.9**	30.6	16.4	34.7	45.9	53.1	36.6	**36.4**	43.0	**29.8**	44.4

Table 5: **Evaluation of O-VOS on DAVIS$_{16}$ `val` set [36]** (§4.5), with region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and time stability $\mathcal{T}$.

Supervision		Non Learning		Unsupervised Learning							Weakly-supervised	
Method		SIFT Flow [25]	BVS [28]	DeepCluster [5]	Transitive Inv [64]	Vondrick *et al.* [56]	mgPFF [20]	TimeCycle [65]	CorrFlow [22]	**Ours**	FlowNet2 [16]	**Ours**
$\mathcal{J}\&\mathcal{F}$	Mean ↑	34.0	**37.3**	35.4	29.4	34.0	44.6	42.8	50.3	**54.3**	26.0	**56.1**
$\mathcal{J}$	Mean ↑	**33.0**	32.9	37.5	32.0	34.6	42.2	43.0	48.4	**52.6**	26.7	**54.0**
	Recall ↑	-	31.8	-	-	34.1	41.8	43.7	53.2	**57.4**	23.9	**60.7**
$\mathcal{F}$	Mean ↑	35.0	**41.7**	33.2	26.8	32.7	46.9	42.6	52.2	**56.1**	25.2	**58.2**
	Recall ↑	-	41.4	-	-	26.8	44.4	41.3	56.0	**58.1**	24.6	**62.2**

Table 6: **Evaluation of O-VOS on DAVIS$_{17}$ `val` set [38]** (§4.5), with region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and average of $\mathcal{J}\&\mathcal{F}$.

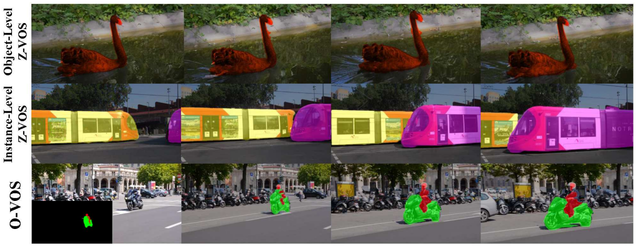

Figure 5: Visual results on three videos (top: *blackswan*, middle: *tram*, bottom: *scooter-black*) under object-level Z-VOS, instance-level Z-VOS and O-VOS setting, respectively (see §4.6). For *scooter-black*, its first-frame annotation is also depicted.

PDB [45], and slightly slower than RVOS [53] (0.3s).

4.5. Performance for O-VOS

Datasets. DAVIS$_{16}$ [36] and DAVIS$_{17}$ [38] datasets are used for performance evaluation under the O-VOS setting.

Evaluation Criteria. Three standard evaluation criteria are reported: region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and the average value of $\mathcal{J}\&\mathcal{F}$. For DAVIS$_{16}$ dataset, we further report the time stability $\mathcal{T}$.

Quantitative Results. Table 5 and Table 6 give evaluation results on DAVIS$_{16}$ and DAVIS$_{17}$, respectively. Table 5 shows that our unsupervised method exceeds representative self-supervised methods (*i.e.*, TimeCyle [65] and CorrFlow [65]) and the best non-learning method (*i.e.*, BVS [28]) across most metrics. In particular, with the learned CAM as supervision, our weakly supervised method further improves the performance, *e.g.*, mean $\mathcal{J}$ of 65.7. Table 6 verifies again our method performs favorably against the current best unsupervised method, CorrFlow, according to mean $\mathcal{J}\&\mathcal{F}$ (54.3 *vs* 50.3). Note that CorrFlow and our method use the same training data. This demonstrates our MuG is able to learn more powerful video object patterns, compared to previous self-learning counterparts.

Runtime Comparison. In O-VOS setting, MuG runs about 0.4s per frame. This is faster than matching based methods (*e.g.*, SIFT Flow [25] (5.1s) and mgPFF [20] (1.3s)), and favorably against self-supervised learning methods, *e.g.*, TimeCycle [65] and CorrFlow [22].

4.6. Qualitative Results

Fig. 5 presents some visual results for object-level ZVOS (top row), instance-level Z-VOS (middle row) and O-VOS (bottom row). For *blackswan* in DAVIS$_{16}$ [36], the primary objects undergo view changes and background clutter, but our MuG still generates accurate foreground segments. The effectiveness of instance-level Z-VOS can be observed in *tram* of DAVIS$_{17}$ [4]. In addition, MuG can produce high-quality results with the given first-frame annotations in O-VOS setting (see the results on the last row for *scooter-black* in DAVIS$_{17}$ [38]), although the different instances suffer from fast motion and scale variation. More results can be found in supplementary materials.

5. Conclusion

We proposed MuG – an end-to-end trainable, unsupervised/weakly supervised learning approach for segmenting objects from the videos. In contrast to current popular supervised VOS solutions requiring extensive amounts of elaborately annotated training samples, our MuG models video object patterns by comprehensively exploring supervision signals from different granularities of unlabeled videos. Our model sets new state-of-the-arts over diverse VOS settings, including object-level Z-VOS, instance-level Z-VOS, and O-VOS. Our model opens up the probability of learning VOS from nearly infinite amount of unlabeled videos and unifying different VOS settings from a single view of video object pattern understanding.

Acknowledgements This work was sponsored by Zhejiang Lab's Open Fund (No. 2019KD0AB04), Zhejiang Lab's International Talent Fund for Young Professionals, CCF-Tencent Open Fund and ARO grant W911NF-18-1-0296.

References

[1] Dosovitskiy Alexey, Springenberg Jost Tobias, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *NIPS*, 2014. 4

[2] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *ECCV*, 2016. 3

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

[4] Sergi Caelles, Jordi Pont-Tuset, Federico Perazzi, Alberto Montes, Kevis-Kokitsi Maninis, and Luc Van Gool. The 2019 davis challenge on vos: Unsupervised multi-object segmentation. *arXiv:1905.00737*, 2019. 7, 8

[5] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018. 8

[6] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *ICCV*, 2017. 2, 6, 7

[7] Alon Faktor and Michal Irani. Video segmentation by non-local consensus voting. In *BMVC*, 2014. 2, 7

[8] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *CVPR*, 2019. 6, 7

[9] Qingnan Fan, Fan Zhong, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Jumpcut: non-successive mask transfer and interpolation for video cutout. *ACM Trans. Graph.*, 34(6):195:1–195:10, 2015. 2, 8

[10] Katerina Fragkiadaki, Geng Zhang, and Jianbo Shi. Video segmentation by tracing discontinuities in a trajectory embedding. In *CVPR*, 2012. 2, 7

[11] Glenn Hartmann, Matthias Grundmann, Judy Hoffman, David Tsai, Vivek Kwatra, Omid Madani, Sudheendra Vijayanarasimhan, Irfan Essa, James Rehg, and Rahul Sukthankar. Weakly supervised learning of object segmentations from web-scale video. In *ECCV*, 2012. 2

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 6

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[14] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 2

[15] Jarmo Hurri and Aapo Hyvärinen. Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural Computation*, 15(3):663–691, 2003. 3

[16] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, 2017. 8

[17] Croitoru Ioana, Bogolin Simion-Vlad, and Leordeanu Marius. Unsupervised learning from video to detect foreground objects in single images. In *ICCV*, 2017. 2, 7

[18] Varun Jampani, Raghudeep Gadde, and Peter V. Gehler. Video propagation networks. In *CVPR*, 2017. 2

[19] Yeong Jun Koh and Chang-Su Kim. Primary object segmentation in videos based on region augmentation and reduction. In *CVPR*, 2017. 2, 7

[20] Shu Kong and Charless Fowlkes. Multigrid predictive filter flow for unsupervised learning on videos. *arXiv preprint arXiv:1904.01693*, 2019. 8

[21] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, 2011. 7

[22] Zihang Lai and Weidi Xie. Self-supervised learning for video correspondence flow. In *BMVC*, 2019. 6, 8

[23] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Frame-to-frame aggregation of active regions in web videos for weakly supervised semantic segmentation. In *ICCV*, 2019. 2

[24] Yong Jae Lee, Jaechul Kim, and Kristen Grauman. Key-segments for video object segmentation. In *ICCV*, 2011. 2, 7

[25] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE TPAMI*, 33(5):978–994, 2010. 8

[26] Xiankai Lu, Wenguan Wang, Chao Ma, Jianbing Shen, Ling Shao, and Fatih Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In *CVPR*, 2019. 2

[27] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *ACCV*, 2018. 2, 6

[28] Nicolas Marki, Federico Perazzi, Oliver Wang, and Alexander Sorkine-Hornung. Bilateral space video segmentation. In *CVPR*, 2016. 2, 8

[29] Grundmann Matthias, Kwatra Vivek, Han Mei, and Essa Irfan. Efficient hierarchical graph-based video segmentation. In *CVPR*, 2010. 2, 8

[30] Hossein Mobahi, Collobert Ronan, and Weston Jason. Deep learning from temporal coherence in video. In *ICML*, 2009. 4

[31] Peter Ochs and Thomas Brox. Object segmentation in video: A hierarchical variational approach for turning point trajectories into dense regions. In *ICCV*, 2011. 2, 7

[32] Bo Pang, Kaiwen Zha, Hanwen Cao, Chen Shi, and Cewu Lu. Deep rnn framework for visual sequential applications. In *CVPR*, 2019. 2

[33] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *ICCV*, 2013. 2, 7

[34] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, 2017. 2, 7

[35] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *CVPR*, 2017. 2, 6

[36] 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. 1, 6, 7, 8

[37] Federico Perazzi, Oliver Wang, Markus H. Gross, and Alexander Sorkine-Hornung. Fully connected object proposals for video segmentation. In *ICCV*, 2015. 2, 8

[38] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Ar-

beláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 8

[39] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 7

[40] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014. 5

[41] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. End-to-end weakly-supervised semantic alignment. In *CVPR*, 2018. 4, 5

[42] Fatemeh Sadat Saleh, Mohammad Sadegh Aliakbarian, Mathieu Salzmann, Lars Petersson, and Jose M. Alvarez. Bringing background into the foreground: Making all classes equal in weakly-supervised video semantic segmentation. In *ICCV*, 2017. 7

[43] Hao Shen. Towards a mathematical understanding of the difficulty in learning with feedforward neural networks. In *CVPR*, 2018. 2

[44] Mennatullah Siam, Chen Jiang, Steven Lu, Laura Petrich, Mahmoud Gamal, Mohamed Elhoseiny, and Martin Jagersand. Video segmentation using teacher-student adaptation in a human robot interaction (hri) setting. In *ICRA*, 2019. 2

[45] Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *ECCV*, 2018. 2, 7, 8

[46] Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *NIPS*, 2015. 4

[47] Kevin Tang, Rahul Sukthankar, Jay Yagnik, and Li Fei-Fei. Discriminative segment annotation in weakly labeled video. In *CVPR*, 2013. 2, 7

[48] Brian Taylor, Vasiliy Karasev, and Stefano Soatto. Causal video object segmentation from persistence of occlusions. In *CVPR*, 2015. 7

[49] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning video object segmentation with visual memory. In *ICCV*, 2017. 2, 7

[50] Yi-Hsuan Tsai, Guangyu Zhong, and Ming-Hsuan Yang. Semantic co-segmentation in videos. In *ECCV*, 2016. 7

[51] Jack Valmadre, Luca Bertinetto, Joao F Henriques, Ran Tao, Andrea Vedaldi, Arnold WM Smeulders, Philip HS Torr, and Efstratios Gavves. Long-term tracking in the wild: A benchmark. In *ECCV*, 2018. 6

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 4

[53] 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. 7, 8

[54] 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

[55] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for video object segmentation. In *BMVC*, 2017. 2, 6

[56] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *ECCV*, 2018. 2, 8

[57] Ning Wang, Yibing Song, Chao Ma, Wengang Zhou, Wei Liu, and Houqiang Li. Unsupervised deep tracking. In *CVPR*, 2019. 3

[58] Wenguan Wang, Xiankai Lu, Jianbing Shen, David J Crandall, and Ling Shao. Zero-shot video object segmentation via attentive graph neural networks. In *ICCV*, 2019. 2

[59] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *CVPR*, 2015. 1, 2

[60] Wenguan Wang, Jianbing Shen, Fatih Porikli, and Ruigang Yang. Semi-supervised video object segmentation with super-trajectories. *IEEE TPAMI*, 41(4):985–998, 2018. 2

[61] Wenguan Wang, Jianbing Shen, and Ling Shao. Consistent video saliency using local gradient flow optimization and global refinement. *IEEE TIP*, 24(11):4185–4196, 2015. 2

[62] Wenguan Wang, Jianbing Shen, and Ling Shao. Video salient object detection via fully convolutional networks. *IEEE TIP*, 27(1):38–49, 2017. 2

[63] Wenguan Wang, Hongmei Song, Shuyang Zhao, Jianbing Shen, Sanyuan Zhao, Steven CH Hoi, and Haibin Ling. Learning unsupervised video object segmentation through visual attention. In *CVPR*, 2019. 2, 7

[64] Xiaolong Wang, Kaiming He, and Abhinav Gupta. Transitive invariance for self-supervised visual representation learning. In *CVPR*, 2017. 8

[65] Xiaolong Wang, Allan Jabri, and Alexei A. Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, 2019. 2, 8

[66] Ziqin Wang, Jun Xu, Li Liu, Fan Zhu, and Ling Shao. Ranet: Ranking attention network for fast video object segmentation. In *ICCV*, 2019. 2

[67] 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

[68] Yichao Yan, Qiang Zhang, Bingbing Ni, Wendong Zhang, Minghao Xu, and Xiaokang Yang. Learning context graph for person search. In *CVPR*, 2019. 2

[69] Yichao Yan, Ning Zhuang, Bingbing Ni, Jian Zhang, Minghao Xu, Qiang Zhang, Zhang Zheng, Shuo Cheng, Qi Tian, Xiaokang Yang, Wenjun Zhang, et al. Fine-grained video captioning via graph-based multi-granularity interaction learning. *IEEE TPAMI*, 2019. 4

[70] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, 2013. 2, 3, 5, 6

[71] Linjie Yang, Yanran Wang, Xuehan Xiong, Jianchao Yang, and Aggelos K Katsaggelos. Efficient video object segmentation via network modulation. In *CVPR*, 2018. 2

[72] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 5

[73] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, Lihe Zhang, Mingyang Qian, and Yizhou Yu. Multi-source weak supervision for saliency detection. In *CVPR*, 2019. 2, 3, 5, 6

[74] Dingwen Zhang, Le Yang, Deyu Meng, Dong Xu, and Junwei Han. Spftn: A self-paced fine-tuning network for segmenting objects in weakly labelled videos. In *CVPR*, 2017. 2

[75] Yu Zhang, Xiaowu Chen, Jia Li, Chen Wang, and Changqun

Xia. Semantic object segmentation via detection in weakly labeled video. In *CVPR*, 2015. 7

[76] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 2, 3, 5

[77] Tianfei Zhou, Shunzhou Wang, Yi Zhou, Yazhou Yao, Jianwu Li, and Ling Shao. Motion-attentive transition for zero-shot video object segmentation. In *AAAI*, 2020. 2

[78] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 6, 7