Bootstrapping Objectness from Videos by Relaxed Common Fate and Visual Grouping

Long Lian[1]
[1]UC Berkeley
longlian@berkeley.edu

Zhirong Wu[2]
[2]Microsoft Research Asia
wuzhiron@microsoft.com

Stella X. Yu[1,3]
[3]University of Michigan
stellayu@umich.edu

Abstract

We study learning object segmentation from unlabeled videos. Humans can easily segment moving objects without knowing what they are. The Gestalt law of common fate, i.e., what move at the same speed belong together, has inspired unsupervised object discovery based on motion segmentation. However, common fate is not a reliable indicator of objectness: Parts of an articulated / deformable object may not move at the same speed, whereas shadows / reflections of an object always move with it but are not part of it.

Our insight is to bootstrap objectness by first learning image features from relaxed common fate and then refining them based on visual appearance grouping within the image itself and across images statistically. Specifically, we learn an image segmenter first in the loop of approximating optical flow with constant segment flow plus small within-segment residual flow, and then by refining it for more coherent appearance and statistical figure-ground relevance.

On unsupervised video object segmentation, using only ResNet and convolutional heads, our model surpasses the state-of-the-art by absolute gains of $7/9/5\%$ on DAVIS16 / STv2 / FBMS59 respectively, demonstrating the effectiveness of our ideas. Our code is publicly available.

1. Introduction

Object segmentation from videos is useful to many vision and robotics tasks [1, 19, 30, 32]. However, most methods rely on pixel-wise human annotations [4, 5, 13, 20, 23, 25, 26, 29, 33, 35, 46, 47], limiting their practical applications.

We focus on learning object segmentation from entirely *unlabeled* videos (Fig. 1). The Gestalt law of *common fate*, i.e., *what move at the same speed belong together*, has inspired a large body of unsupervised object discovery based on motion segmentation [6, 18, 22, 28, 41, 43, 45].

There are three main types of *unsupervised* video object segmentation (UVOS) methods. **1) Motion segmentation** methods [18, 28, 41, 43] use motion signals from a pretrained optical flow estimator to segment an image into foreground objects and background (Fig. 1). OCLR [41] achieves state-of-the-art performance by first synthesizing a dataset with arbitrary objects moving and then training a motion segmentation model with known object masks. **2) Motion-guided image segmentation** methods such as GWM [6] use motion segmentation loss to guide appearance-based segmentation. Motion between video frames is only required during training, not during testing. **3) Joint appearance segmentation and motion estimation** methods such as AMD [22] learn motion and segmentation simultaneously in a self-supervised fashion by reconstructing the next frame based on how segments of the current frame move.

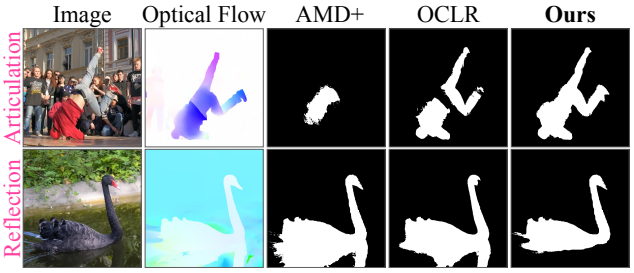


Figure 1. **We study how to discover objectness from unlabeled videos based on common motion and appearance.** AMD [22] and OCLR [41] rely on *common fate*, *i.e.*, what move at the same speed belong together, which is not always a reliable indicator of objectness. **Top**: Articulation of a human body means that object parts may not move at the same speed; common fate thus leads to *partial objectness*. **Bottom**: Reflection of a swan in water always moves with it but is not part of it; common fate thus leads to *excessive objectness*. Our method discovers full objectness by relaxed common fate and visual grouping. AMD+ refers to AMD with RAFT flows as motion supervision for fair comparison.

However, while *common fate* is effective at binding parts of heterogeneous appearances into one whole moving object, it is not a reliable indicator of objectness (Fig. 1).

1. **Articulation**: Parts of an articulated or deformable object may not move at the same speed; common fate thus leads to *partial* objectness containing the major moving part only. In Fig.1 top, AMD+ discovers only the mid-

Unsupervised object segmentation	MG	AMD	GWM	**Ours**
Sources of supervision	M	M*	M	M+A
Segment stationary objects?	✗	✓	✓	✓
Handle articulated objects?	-	✗	✗	✓
Label-free hyperparameter tuning?	✗	✗	✗	✓

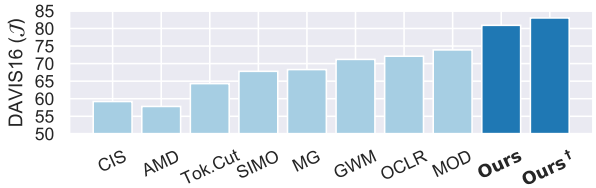


Figure 2. Advantages over leading unsupervised object segmentation methods MG [43]/AMD [22]/GWM [6]: **1)** With motion supervision instead of motion input, we can segment stationary objects. **2)** With both motion (M) and appearance (A) as supervision, we can discover full objectness from noisy motion cues. M* refers to implicit motion via image warping. **3)** By modeling relative motion within an object, we can handle articulated objects. **4)** By comparing motion-based segmentation with appearance-based segmentation, we can tune hyperparameters without labels. Our performance gain is substantial, more with post-processing (†).

dle torso of the street dancer since it moves the most, whereas OCLR misses the exposed belly which is very different from the red hoodie and the gray jogger.

2. **Reflection**: Shadows or reflections of an object always move with the object but are not part of the object; common fate thus leads to *excessive* objectness that covers more than the object. In Fig.1 bottom, AMD+ or OCLR cannot separate the swan *from* its reflection in water.

We have two insights to bootstrap full objectness from common fate in unlabeled videos. **1)** To detect an articulated object, we allow various parts of the same object to assume different speeds that deviate slightly from the object's overall speed. **2)** To detect an object from its reflections, we rely on visual appearance grouping within the image itself and statistical figure-ground relevance. For example, swans tend to have distinctive appearances from the water around them, and reflections may be absent in some swan images.

Specifically, we learn unsupervised object segmentation in two stages: Stage 1 learns to discover objects from motion supervision with relaxed common fate, whereas Stage 2 refines the segmentation model based on image appearance.

At Stage 1, we discover objectness by computing the optical flow and learning an image segmenter in the loop of approximating the optical flow with *constant segment flow* plus *small within-segment residual flow*, relaxing *common fate* from the strict same-speed assumption. **At Stage 2**, we refine our model by image appearance based on low-level visual coherence within the image itself and usual figure-ground distinction learned statistically across images.

Existing UVOS methods have hyperparameters that significantly impact the quality of segmentation. For example, the number of segmentation channels is a critical parameter for AMD [22], and it is usually chosen according to an annotated validation set in the downstream task, defeating the claim of *unsupervised* objectness discovery.

We propose **unsupervised hyperparameter tuning** that does not require any annotations. We examine how well our motion-based segmentation aligns with appearance-based affinity on DINO [2] features self-supervisedly learned on ImageNet [34], which is known to capture semantic objectness. Our idea is also *model-agnostic* and applicable to other UVOS methods.

Built on the novel concept of Relaxed Common Fate (RCF), our method has several advantages over leading UVOS methods (Fig. 2): It is the only one that uses both motion and appearance to supervise learning; it can segment stationary and articulated objects in single images, and it can tune hyperparameters without external annotations.

On UVOS benchmarks, using only standard ResNet [12] backbone and convolutional heads, our RCF surpasses the state-of-the-art by absolute gains of $7.0\%/9.1\%/4.5\%$ ($6.3\%/12.0\%/5.8\%$) without (with) post-processing on DAVIS16 [32] / STv2 [19] / FBMS59 [30] respectively, validating the effectiveness of our ideas.

2. Related Work

Unsupervised video object segmentation (UVOS) requires segmenting prominent objects from videos without human annotations. Mainstream benchmarks [1, 19, 30, 32] define the task as a binary figure-ground segmentation problem, where salient objects are the foreground. Despite the name, several previous UVOS methods require *supervised* (pre-)training on *other* data such as large-scale images or videos *with* manual annotations [10, 16, 20, 25, 33, 44, 46, 47]. In contrast, we focus on UVOS methods which do not rely on any labels at either *training or inference* time.

Motion segmentation separates figure from ground based on motion, which is typically optical flow computed from a pre-trained model. FTS [31] utilizes motion boundaries for segmentation. SAGE [39] additionally considers edges and saliency priors jointly with motion. CIS [45] uses independence between foreground and background motion as the goal for foreground segmentation. However, this assumption does not always hold in real-world motion patterns. MG [43] leverages attention mechanisms to group pixels with similar motion patterns. SIMO [18] and OCLR [41] generate synthetic data for segmentation supervision, with the latter supporting individual segmentation of multiple objects. Nevertheless, both rely on human-annotated sprites for realistic shapes in artificial data synthesis. Motion segmentation fails when objects do not move.

Motion-guided image segmentation treats motion computed by a pre-trained optical flow model such as RAFT [38] as ground-truth and uses it to supervise appearance-

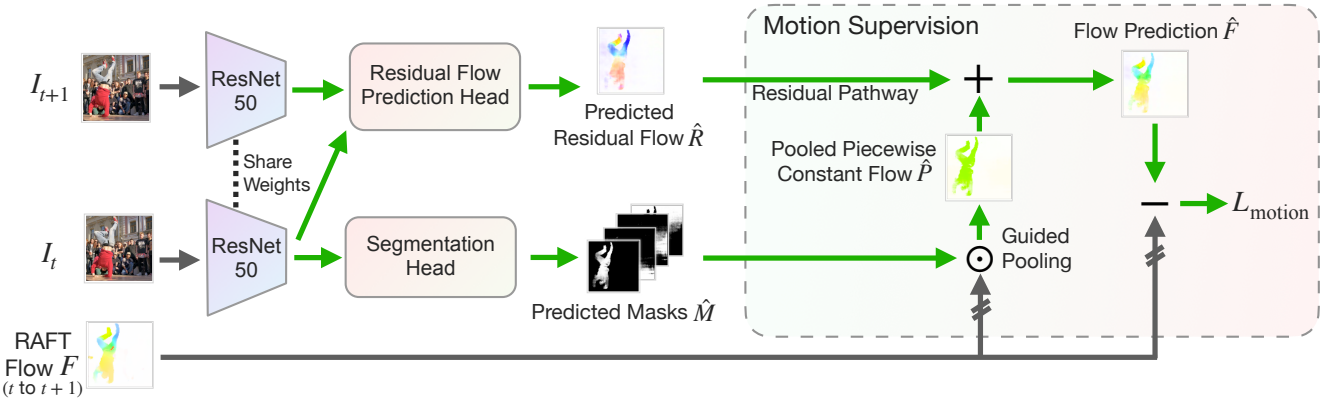


Figure 3. **Our object discovery stage uses motion as supervision and follows the principle of relaxed common fate**, in which training signals are obtained by reconstructing the reference RAFT flow with the sum from the two pathways: **1)** a piecewise constant flow pathway, which is created from pooling the RAFT flow with the predicted masks in order to model object-level motion; **2)** a predicted pixel-wise residual flow pathway, which models intra-object motion for articulated and deformable objects. Green arrows indicate gradient backprop.

based image segmentation. GWM [6] assumes smooth flows within an object and learns appearance-based segmentation by seeking the best segment-wise affine flows that fit RAFT flows. Such methods can discover stationary objects in videos and single images.

Joint appearance segmentation and motion estimation methods such as AMD [22] learn motion and segmentation simultaneously in a self-supervised manner such that their outputs can be used to successfully reconstruct the next frame based on how segments of the current frame move.

AMD is unique in that it has no preconception of optical flow or visual saliency. Since our model considers bootstrapping objectness from optical flow, for fair comparisons, we consider AMD+, a version of AMD with motion supervision from RAFT flows [38] instead.

Existing UVOS methods, whether they examine motion only or together with appearance, assume that objectness is revealed through common fate of motion: What move at the same speed belong together. We show that this notion fails for objects with articulation and reflection (Fig. 1). Our RCF first bootstraps objectness by relaxed common fate and then improves it by visual appearance grouping.

3. Objectness from Relaxed Common Fate

Our RCF consists of two stages: a motion-supervised object discovery stage (Fig. 3) and an appearance-supervised refinement stage (Fig. 4). Stage 1 formalizes relaxed common fate and learns segmentation by fitting RAFT flow with both object-level motion and intra-object motion. Stage 2 refines Stage 1's motion-based segmentations by appearance-based visual grouping and then use them to further supervise segmentation learning. Neither stage requires any annotation, making RCF *fully unsupervised*. We also present motion-appearance alignment as a model-agnostic label-free hyperparameter tuner.

3.1. Problem Setting

Let $I_t \in \mathbb{R}^{3\times h\times w}$ be the t^{th} frame from a sequence of T RGB frames, where h and w are the height and width of the image respectively. We will omit the subscript t except for input images for clarity. The goal of UVOS is to produce a binary segmentation mask $M \in \{0,1\}^{h\times w}$ for each time step t, with 1 (0) indicating the foreground (background).

To evaluate a method on UVOS, we compute the mean Jaccard index $\mathcal{J}$ (*i.e.*, mean IoU) between the predicted segmentation mask M and the ground truth G. In UVOS, the ground truth mask G is not available, and no human-annotations are used throughout training and inference.

3.2. Object Discovery with Motion Supervision

As shown in Fig. 3, during training, our method takes a pair of consecutive frames and RAFT flow between them as inputs. To instantiate the idea of common fate, our method begins by pooling the RAFT Flow with respect to the predicted masks, creating the piecewise constant flow pathway. As a relaxation, the predicted residual flow, which models intra-object motion for articulated and deformable objects, is added to the piecewise constant flow. The composite flow prediction is then supervised by the RAFT flow to train the model. At test time, only the backbone and the segmentation head are utilized to perform inference per frame.

Specifically, let $f(I_t) \in \mathbb{R}^{K\times H\times W}$ be the feature of I_t extracted from backbone $f(\cdot)$, where K, H, and W are the number of channels, height, and width of the feature. Let $\hat{M} = g(f(I_t)) \in \mathbb{R}^{C\times H\times W}$ be C soft segmentation masks extracted with a lightweight fully convolutional segmentation head $g(\cdot)$ taking the image feature from $f(\cdot)$. `Softmax` is taken across channels inside $g(\cdot)$ so that the C soft masks sum up to 1 for each of the $H \times W$ positions. Following [22], although there are C segmentation masks competing for each pixel (i.e., C output channels in $\hat{M}$), *only one* corresponds to the foreground, with the rest

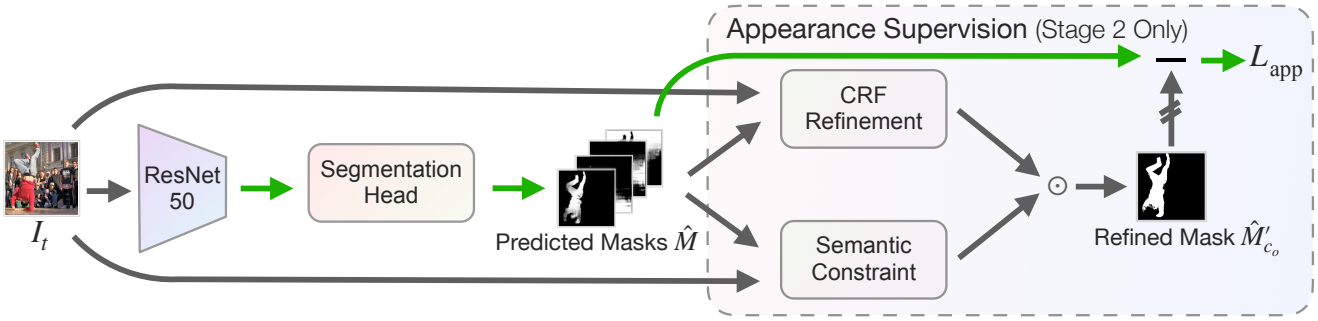


Figure 4. **Our appearance refinement stage** corrects misconceptions from motion supervision. The predicted mask is supervised by a refined mask based on both the CRF that enforces low-level appearance consistency (*e.g.*, color and texture) and the semantic constraint that enforces high-level semantic consistency. External frozen image features used to enforce the semantic constraint are omitted for clarity.

capturing background patches. We define c_o as the object channel index with value obtained in Sec. 3.4.

Following [6,18,41,43], we use off-the-shelf optical flow model RAFT [38] trained on synthetic datasets [9, 27] to provide motion cues between consecutive frames. Let $F \in \mathbb{R}^{2\times H\times W}$ be the flow output from RAFT from I_t to I_{t+1}.

Piecewise constant pathway. We first pool the flow according to each mask to form C flow vectors $\hat{P}_c \in \mathbb{R}^2$:

$$\hat{P}_c = \phi_2(\text{GuidedPool}(\phi_1(F), \hat{M}_c)) \tag{1}$$

$$\text{GuidedPool}(F, M) = \frac{\sum_{p=1}^{HW}(F \odot M)[p]}{\sum_{p=1}^{HW} M[p]} \tag{2}$$

where $[p]$ denotes the pixel index and $\odot$ element-wise multiplication. Following [22], ϕ_1 and ϕ_2 are two-layer lightweight MLPs that transform each of the motion vectors independently before and after pooling, respectively. We then construct predicted flow $\hat{P} \in \mathbb{R}^{2\times H\times W}$ according to the soft segmentation mask:

$$\hat{P} = \sum_{c=1}^{C} \text{Broadcast}(\hat{P}_c, \hat{M}_c) \tag{3}$$

$$\text{Broadcast}(\hat{P}_c, \hat{M}_c)[p] = \hat{P}_c \odot (\hat{M}_c[p]). \tag{4}$$

As the mask prediction $\hat{M}_c$ approaches binary during training, the flow prediction approaches a piecewise-constant function with respect to each segmentation mask, capturing common fate. Previous methods either directly supervise $\hat{P}$ with an image warping for self-supervised learning [22] or matches $\hat{P}$ and F by minimizing the discrepancies up to an affine factor (*i.e.*, up to first order) [6].

Nonetheless, hand-crafted non-learnable motion models, while capturing the notion of common fate, underfit complex optical flow in real-world videos, which often put object parts into different segmentation channels in order to minimize the loss, despite similar color or texture. [6] uses two mask channels as a remedy, still falling short for scenes with complex backgrounds.

Learnable residual pathway. Rather than using more complicated hand-crafted motion models to model the motion patterns in videos, we employ *relaxed* common fate by separately fitting object-level and intra-object motion by adding a *learnable* residual pathway $\hat{R}$ in addition to the piecewise constant pathway $\hat{P}$ to form the final flow prediction $\hat{F}$. The residual pathway models relative intra-object motion such as the relative motion of the dancer's feet to the body in Fig. 3.

Let $h(\cdot)$ be a lightweight module with three Conv-BN-ReLU blocks that take the concatenated feature of a pair of frames $\{I_t, I_{t+1}\}$ as input and predicts $\hat{R}' \in \mathbb{R}^{C\times 2\times H\times W}$, which includes C flows with per-pixel upper bound λ:

$$\hat{R}' = \lambda \tanh(h(\text{concat}(f(I_t), f(I_{t+1})))) \tag{5}$$

where the upper bound λ is set to 10 pixels unless stated otherwise. The C residual flows form aggregated residual flow $\hat{R}$ using mask predictions, which sums up with the piecewise constant pathway to form the final flow prediction $\hat{F}$:

$$\hat{R} = \sum_{c=1}^{C} \hat{R}'_c \odot \hat{M}_c \tag{6}$$

$$\hat{F} = \hat{P} + \hat{R} \tag{7}$$

In this way, $\hat{F}$ additionally takes into account relative motion that is within $(-\lambda, \lambda)$ for each spatial location. The added residual pathway provides greater flexibility by allowing the model to relax from common fate that does not take intra-object motion into account. This leads to better segmentation results for articulated and deformable objects.

At stage 1, we minimize the L1 loss between the predicted reconstruction flow $\hat{F}$ and target flow F in order to learn segmentation by predicting the correct flow:

$$L_{\text{stage 1}} = L_{\text{motion}} = \frac{1}{HW}\sum_{p=1}^{HW} ||\hat{F}[p] - F[p]||_1 \tag{8}$$

3.3. Refinement with Appearance Supervision

A primary focus of self-supervised learning is to find sources of useful training signals. While the residual pathway greatly improves segmentation quality, the supervision

still primarily comes from motion. This single source of supervision can lead to predictions that are optimal for flow prediction but often suboptimal from an appearance perspective. For instance, in Fig. 4, the segmentation prediction before refinement ignores a part of the dancer's leg, despite the ignored part sharing a very similar color and texture with the included parts. Furthermore, the RAFT flow tends to be noisy in areas where nearby pixels move very differently, which leads to segmentation ambiguity.

To address these issues, we propose to incorporate low- and high-level appearance signals as another source of supervision to correct the misconceptions from motion.

Appearance supervision with low-level intra-image cues. With the model in stage 1, we obtain the mask prediction $\hat{M}_{c_o}$ of I_t, where c_o is the objectness channel that could be found without annotation (Sec. 3.4). We then apply fully-connected conditional random field (CRF) [17], a training-free technique that refines the value of each prediction based on other pixels with an appearance and a smoothness kernel. The refined masks $\hat{M}'_{c_o}$ are then used as supervision to provide appearance signals in training:

$$\hat{M}'_{c_o} = \text{CRF}(\hat{M}_{c_o}) \tag{9}$$

$$L_{\text{app}} = \frac{1}{HW} \sum_{p=1}^{HW} ||\hat{M}_{c_o}[p] - \hat{M}'_{c_o}[p]||_2^2 \tag{10}$$

Since stage 2 is mainly misconception correction and thus much shorter than stage 1, we generate the refined masks for supervision only once between the two stages for efficiency.

The total loss in stage 2 is a weighted sum of both motion and appearance loss:

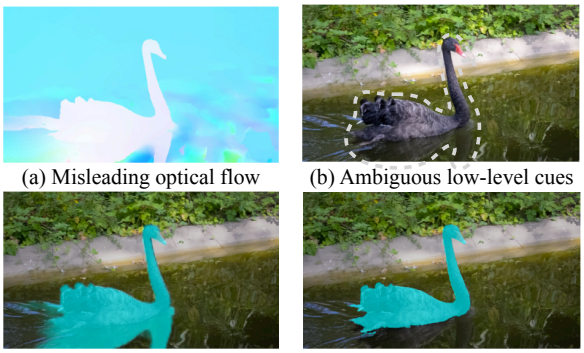
$$L_{\text{stage 2}} = w_{\text{app}} L_{\text{app}} + w_{\text{motion}} L_{\text{motion}} \tag{11}$$

where w_{app} and w_{motion} are weights for loss terms.

The CRF in our method for appearance supervision is different with the traditional CRF used in post-processing [6,45], as our refined masks provide the supervision for training the network. Furthermore, we show empirically that our method is orthogonal to the traditional CRF in the ablation (Sec. 4.4).

Appearance supervision with semantic constraint. Low-level appearance is still insufficient to address misleading motion signals from naturally occurring confounders with similar motion patterns. For example, the reflections share similar motion as the swan in Fig. 5, which is confirmed by low-level appearance. However, humans could recognize that the swan and the reflection have distinct semantics, with the reflection's semantics much closer to the background.

Inspired by this, we incorporate the statistically learned feature map from a frozen auxiliary DINO ViT [2,8] trained with self-supervised learning across ImageNet [34] without human annotation, to create a semantic constraint for mask prediction. We begin by taking the key features from the last transformer layer, denoted as $f_{\text{aux}}(I_t)$, inspired by [40].



(a) Misleading optical flow (b) Ambiguous low-level cues

(c) Seg. mask (pre-refine) (d) Seg. mask (post-refine)

Figure 5. **Semantic constraint mitigates false positives from naturally-occurring misleading motion signals.** The reflection has semantics distinct from the main object and is thus filtered out. The refined mask is then used as supervision to disperse the misconception in stage 2. Best viewed in color and zoom in.

Next, we compute and iteratively optimize the normalized cut [36] with respect to the mask to refine the mask.

Specifically, we initialize a 1-D vector $\mathbf{x}$ with a flattened and resized $\hat{M}_{c_o}$ with shape HW. Then we build an appearance-based affinity matrix A, where:

$$A_{ij} = \mathbb{1}(\text{sim}(f_{\text{aux}}(I_t)_i, f_{\text{aux}}(I_t)_j) \geq 0.2) \tag{12}$$

Next, we compute $\text{NCut}(A, \mathbf{x})$:

$$\text{Cut}(A, \mathbf{x}) = (1 - \mathbf{x})A\mathbf{x} \tag{13}$$

$$\text{NCut}(A, \mathbf{x}) = \frac{\text{Cut}(A, \mathbf{x})}{\sum_{i=1}^{HW}(A\mathbf{x})_i} + \frac{\text{Cut}(A, \mathbf{x})}{\sum_{i=1}^{HW}(A(1-\mathbf{x}))_i} \tag{14}$$

where $\text{sim}(\cdot, \cdot)$ cosine similarity. Since $\text{NCut}(A, \mathbf{x})$ is differentiable with respect to $\mathbf{x}$, we use Adam [15] to minimize $\text{NCut}(A, \mathbf{x})$ in order to refine $\mathbf{x}$ for $k = 10$ iterations. We denote the optimized vector as $\mathbf{x}^{(k)}$, which is thus the refined version of the mask that carries consistent semantics, thus decoupling the objects from their shadows and reflections. With the semantic constraint, Eq. (9) changes to:

$$\hat{M}'_{c_o} = \text{CRF}(\hat{M}_{c_o}) \odot \text{CRF}(\mathbf{x}^{(k)}) \tag{15}$$

where $\mathbf{x}^{(k)}$ is reshaped to 2D and resized to match the mask sizes prior to CRF.

Since the semantic constraint introduces an additional frozen model $f_{\text{aux}}(\cdot)$, we benchmark both *with* and *without* the semantic constraint for a fair comparison with previous methods. We use ***RCF** (w/o SC)* to denote RCF without the semantic constraint. Our method is still fully unsupervised even with the semantic constraint.

3.4. Label-free Hyperparameter Tuner

Following previous appearance-based UVOS work, our method also requires several tunable hyperparameters for high-quality segmentation. The most critical ones are the number of segmentation channels C and the object channel

index c_o. [6, 22] tune both hyperparameters either with a large labeled validation set or a hand-crafted metric tailored to a specific hyperparameter, limiting the capability towards other hyperparameters in a real-world setting.

We propose motion-appearance alignment as a metric to quantify the segmentation quality. The steps for tuning are:

1. Train a model with each hyperparameter setting.
2. Export the predicted mask $\hat{M}_{c_o}$ for each image in the *unlabeled* validation set.
3. Compute the negative normalized cuts $-\text{NCut}(A, \hat{M}_{c_o})$ w.r.t. $\hat{M}_{c_o}$ and the appearance-based affinity matrix A as the metric quantifying motion-appearance alignment.
4. Take the mean metric across all validation images.
5. Select the setting with the highest mean metric.

Our hyperparameter tuning method is model-agnostic and applicable to other UVOS methods. We also demonstrate its effectiveness in tuning weight decay and present the pseudo-code in the supp. mat.

4. Experiments

4.1. Datasets

We evaluate our methods using three datasets commonly used to benchmark UVOS, following previous works [6, 22, 41, 43, 45]. **DAVIS2016** [32] contains 50 video sequences with 3,455 frames in total. Performance is evaluated on a validation set that includes 20 videos with annotations at 480p resolution. **SegTrackv2** (STv2) [19] contains 14 videos of different resolutions, with 976 annotated frames and lower image quality than [32]. **FBMS59** [30] contains 59 videos with a total of 13,860 frames, 720 frames of which are annotated with a roughly fixed interval. We follow previous work to merge multiple foreground objects in STv2 and FBMS59 into one mask and train on all unlabeled videos. We adopt mean Jaccard index $\mathcal{J}$ (mIoU) as the primary evaluation metric.

4.2. Unsupervised Video Object Segmentation

Setup. Our architecture is simple and straightforward. We use a ResNet50 [12] backbone followed by a segmentation head and a residual prediction head. Both heads only consist of three Conv-BN-ReLU layers with 256 hidden units. This standard design allows efficient implementation in real-world applications. Unless otherwise stated, we use $C = 4$ object channels, which we determine without human annotation in Sec. 4.3. We also determine the object channel index c_o using the same approach. The RAFT [38] model we use is only trained on synthetic FlyingChairs [9] and FlyingThings [27] dataset without human annotation. For more details, please refer to supplementary materials.

Results. As shown in Tab. 1, RCF outperforms previous methods under fair comparison, often by a large margin. On DAVIS16, RCF surpasses the previous state-of-the-art method by 7.0% without post-processing (abbreviated as pp.). With CRF as the only pp., RCF improves on previous methods by 6.3% without techniques such as multi-step flow, multi-crop ensemble, and temporal smoothing. RCF also outperforms GWM [6] that employs more complex Swin-T + MaskFormer architecture [3, 24] by 9.7% w/o pp. Furthermore, RCF achieves significantly better results compared with TokenCut [40] that also uses normalized cuts on DINO features [2] (16.6% better w/o pp.). Despite the varying image quality in STv2 and FBMS59, RCF improves over past methods under fair comparison, by 9.1% and 4.5% without pp, respectively. Semantic constraint (SC) could be included if additional gains are desired. However, RCF still outperforms previous works without the semantic constraint (5.3% improvement on DAVIS16 w/o SC), thus *not relying on external frozen features*.

Methods	Post-process	DAVIS16	STv2	FBMS59
SAGE [39]		42.6	57.6	61.2
CUT [14]		55.2	54.3	57.2
FTS [31]		55.8	47.8	47.7
EM [28]		69.8	–	–
CIS [45]		59.2	45.6	36.8
MG [43]		68.3	58.6	53.1
AMD [22]		57.8	57.0	47.5
SIMO [18]		67.8	62.0	–
GWM [6]		71.2	66.7	60.9
GWM* [6]		71.2	69.0	66.9
OCLR† [41]		72.1	67.6	65.4
TokenCut [40]		64.3	59.6	60.2
MOD [7]		73.9	62.2	61.3
RCF		**80.9**	**76.7**	**69.9**
		(+7.0)	**(+9.1)**	**(+4.5)**
CIS [45]	CRF + SP‡	71.5	62.0	63.6
TokenCut [40]	CRF only	76.7	61.6	66.6
GWM* [6]	CRF + SP‡	73.4	72.0	68.6
OCLR† [41]	DINO-based‡	78.9	71.6	68.7
MOD [7]	DINO-based‡	79.2	69.4	66.9
RCF (w/o SC)	CRF only	82.0	78.8	71.9
RCF	CRF only	**83.0**	**79.6**	**72.4**
		(+6.3)	**(+12.0)**	**(+5.8)**

Table 1. **Our method achieves significant improvements over previous methods on common UVOS benchmarks**. RCF (w/o SC) indicates low-level refinement only (no f_{aux} used). $*$: uses Swin-Transformer w/ MaskFormer [3, 24] segmentation head orthogonal to VOS method and thus is not a fair comparison with us. † leverages manually annotated shapes from large-scale Youtube-VOS [42] to generate synthetic data. ‡: *SP*: significant post-processing (*e.g.*, multi-step flow, multi-crop ensemble, and temporal smoothing). *DINO-based*: performs contrastive learning or mask propagation on a pretrained DINO ViT model [2, 8] at test time; not a fair comparison with us. Our post-processing is a *CRF pass only*. CIS results are from [23].

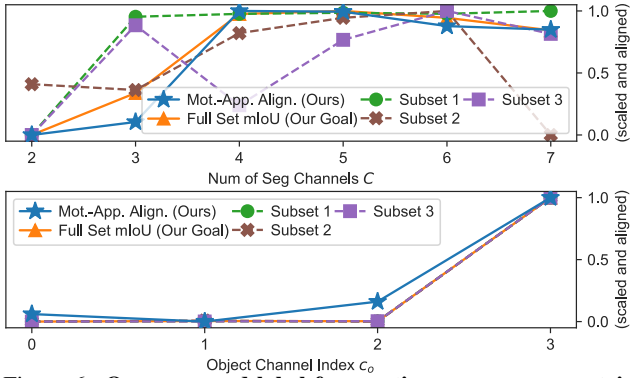


Figure 6. **Our proposed label-free motion-appearance metric aligns well with mIoU on the full validation set. Top:** When tuning the number of segmentation channels C, our method follows full validation set mIoU better than mIoU on validation subsets with 25% of the sequences labeled. **Bottom:** Our method correctly determines the object channel $c_o = 3$ for this run, without any human labels. Although c_o varies in each training run by design [22], our tuning method has negligible overhead and can be performed after training ends to find c_o *within seconds*.

4.3. Label-free Hyperparameter Tuning

We use motion-appearance alignment as a metric to tune two key hyperparameters: the number of segmentation masks C and the object channel index c_o. To simulate the real-world scenario that we only have limited labeled validation data, we also randomly sample 25% sequences three times to create three labeled subsets of the validation set to evaluate mIoU on. As shown in Fig. 6, for **the number of mask channels** C, despite *not using any manual annotation*, our label-free motion-appearance alignment closely follows the validation mIoU compared to mIoU on validation subsets, showing the effectiveness of our metric on hyperparameter tuning. Although increasing the number of channels improves the segmentation quality of our model by increasing its fitting power, such an increase saturates at $C = 4$. Therefore, we use $C = 4$ unless otherwise stated. Regarding **the object channel index** c_o, since it changes with each random initialization [22], optimal c_o needs to be obtained at the end of each training run. We propose to use only the first frame of each video sequence for finding c_o. With this adjustment, our tuning method completes within *only 3 seconds* for each candidate channel, which enables our tuning method to be performed after the whole training run with negligible overhead.

4.4. Ablation Study

Contributions of each component. As shown in Tab. 2, residual pathway allows more flexibility and contributes 7.8% mIoU. The appearance refinement in the second stage boosts the performance to 80.7%, resulting in a 9.6% gain in total. The CRF post-processing leads to 83.0% mIoU, an 11.9% increase over the baseline.

Residual pathway	Low-level refinement	Semantic constraint	CRF	$\mathcal{J}$ (↑)
				71.1
✓				78.9 (+7.8)
✓	✓			79.7 (+8.6)
✓	✓	✓		80.7 (+9.6)
✓	✓	✓	✓	**83.0 (+11.9)**

Table 2. **Effect of each component of our method (DAVIS16).** Residual pathway on its own provides the most improvement in our method. All components together contribute to an 11.9% gain.

Variants	DAVIS16 $\mathcal{J}$ (↑)
None	71.1
None (w/ robust loss [37])	74.0
Scaling	73.8
Residual (affine)	76.3
Residual	**78.9**

Table 3. **Ablations on additional pathway confirm our design choice of residual pathway.** We benchmark without the refinement stage to show the raw performance gain.

DAVIS16 $\mathcal{J}$ (↑)	Stage 1 only	Stage 1 & 2
Without post-processing	78.9	80.9
With CRF post-processing	**81.4**	**83.0**
Δ	+2.5	+2.1

Table 4. **The refinement CRF in our stage 2 is orthogonal to upsampling CRF in post-processing**, since the latter still gives significant improvements even with CRF in stage 2.

Designing additional pathway. In Tab. 3, we show that robustness loss [21,37] does not effectively reduce the impact of misleading motion. We also implemented a pixel-wise scaling pathway, which multiplies each value of the motion vector by a predicted value. Furthermore, we fit an affine transformation per segmentation channel as the residual. In our setting, the pixel-wise residual performs the best and is selected for our model, showing the effectiveness of a *learnable* and *flexible* motion model inspired by relative motion.

Orthogonality of our appearance supervision with post-processing. The refined masks after our appearance-based refinement have the same resolution as the original exported masks. Therefore, the refinement CRF in stage 2 has an orthogonal effect to the upsampling CRF in post-processing which is mainly used to create high-resolution masks from bilinearly-upsampled ones. As shown in Tab. 4, the gains that come from post-processing remain comparable as we apply appearance-based refinement in stage 2, which also shows our orthogonality to post-processing.

Modeling camera motion? RCF does not explicitly model the flow from camera motion. To investigate whether modeling camera motion could further benefit RCF, we estimate

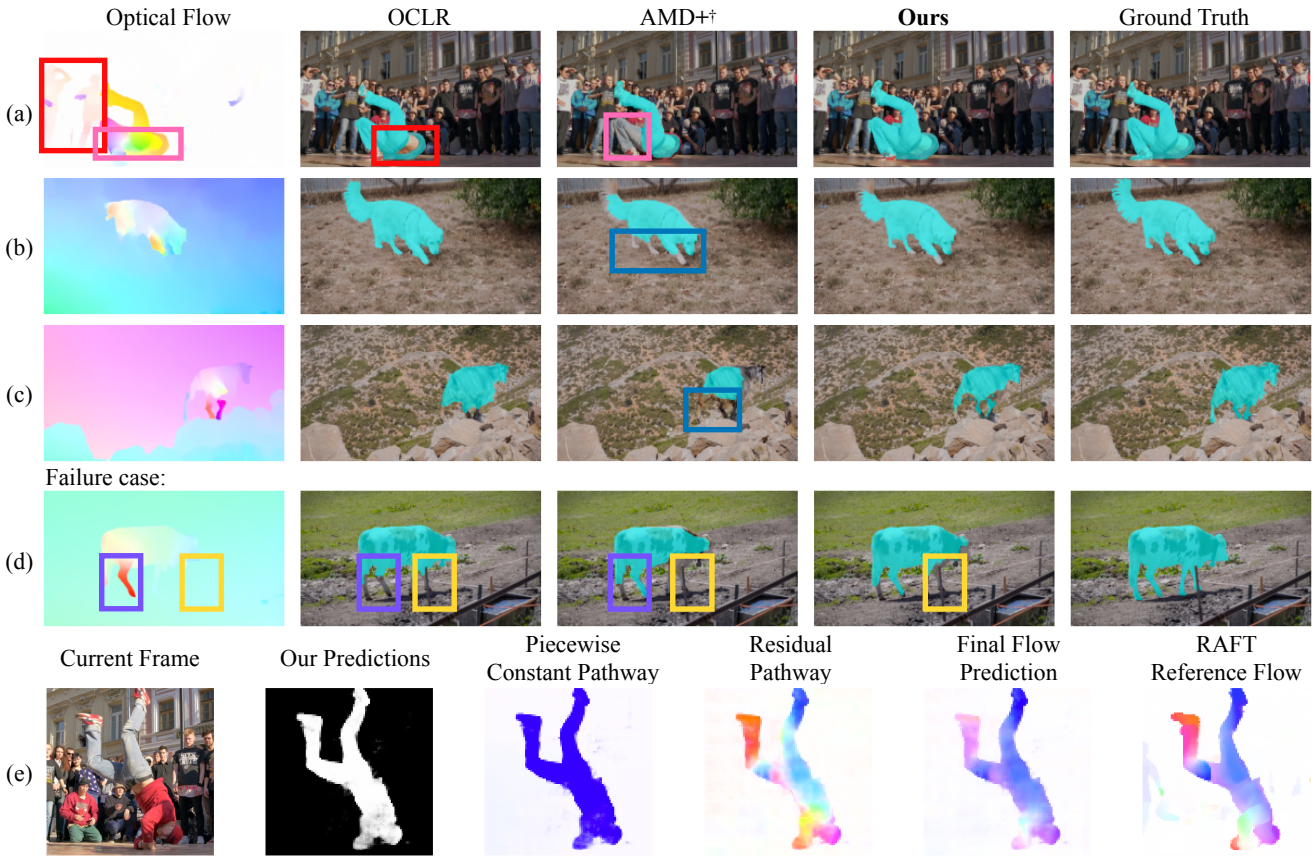


Figure 7. **Our method delivers great performance in challenging scenes.** Our method shows significant improvements compared to OCLR [41] and AMD [22] in scenes with complex foreground motion (a)(b), distracting background motion (a)(c), motion parallax from camera motion (c). In the failure case (d), neither motion nor appearance information is informative, leading to the front legs being missed from the segmentation. However, our method still outperforms previous works and segments most of the cow's hind legs. (e) shows that the piecewise-constant pathway and the residual pathway work together to fit the reference flow, resulting in high-quality segmentation. The symbol † denotes AMD with higher-quality RAFT flow [38] for a fair comparison. More visualizations are available in the supp. mat.

Camera motion modeling	No	Yes
DAVIS16 $\mathcal{J}$ (↑)	**78.9**	77.9

Table 5. **Modeling camera motion does not improve our method.** Lower segmentation quality results from removing camera motion as preprocessing. Only stage 1 is used in both settings.

it with the planar homography and RANSAC [11] and remove it as a preprocessing step prior to training our method. Despite the relatively accurate estimation when visualized, Tab. 5 shows that it is ineffective in improving the segmentation quality. We hypothesize that it is because 3D camera motion is equivalent to 3D scene motion in an opposite direction and thus additional modeling is unnecessary.

4.5. Visualizations and Discussions

Fig. 7 compares RCF with [22, 41] and shows its ability to handle challenging cases such as complex non-uniform foreground motion, distracting background motion, and camera motion including rotation. However, when neither motion nor appearance provides informative signals, RCF may suffer from the lack of information. For instance, in the absence of relative motion, RCF is misled by the similarity between the color of the cow's front legs and the color of the ground in Fig. 7(d). Although RCF has the ability to recognize multiple foreground objects with similar motion, it sometimes captures only one object when the objects move in very different patterns. Finally, RCF is not designed to separate multiple foreground objects. More visualizations and discussions are available in the supp. mat.

5. Summary

We present RCF, an unsupervised video object segmentation method based on relaxed common fate and appearance grouping. Our approach includes a motion-supervised object discovery stage with a learnable residual pathway, a refinement stage with appearance supervision, and using motion-appearance alignment as a label-free hyperparameter tuning method. Extensive experiments show our method's effectiveness and utility in challenging scenarios.

References

[1] T Brox, J Malik, and P Ochs. Freiburg-berkeley motion segmentation dataset (fbms-59). In *European Conference on Computer Vision (ECCV)*, 2010. 1, 2

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021. 2, 5, 6

[3] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. 2021. 6

[4] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *European Conference on Computer Vision*, pages 640–658. Springer, 2022. 1

[5] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. *Advances in Neural Information Processing Systems*, 34:11781–11794, 2021. 1

[6] Subhabrata Choudhury, Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. Guess what moves: Unsupervised video and image segmentation by anticipating motion. *arXiv preprint arXiv:2205.07844v1*, 2022. 1, 2, 3, 4, 5, 6

[7] Shuangrui Ding, Weidi Xie, Yabo Chen, Rui Qian, Xiaopeng Zhang, Hongkai Xiong, and Qi Tian. Motion-inductive self-supervised object discovery in videos. *arXiv preprint arXiv:2210.00221*, 2022. 6

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5, 6

[9] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 4, 6

[10] Alon Faktor and Michal Irani. Video segmentation by non-local consensus voting. In *BMVC*, volume 2, page 8, 2014. 2

[11] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 8

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 6

[13] Ge-Peng Ji, Keren Fu, Zhe Wu, Deng-Ping Fan, Jianbing Shen, and Ling Shao. Full-duplex strategy for video object segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4922–4933, 2021. 1

[14] Margret Keuper, Bjoern Andres, and Thomas Brox. Motion trajectory segmentation via minimum cost multicuts. In *Proceedings of the IEEE international conference on computer vision*, pages 3271–3279, 2015. 6

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[16] Yeong Jun Koh and Chang-Su Kim. Primary object segmentation in videos based on region augmentation and reduction. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7417–7425. IEEE, 2017. 2

[17] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24, 2011. 5

[18] Hala Lamdouar, Weidi Xie, and Andrew Zisserman. Segmenting invisible moving objects. 2021. 1, 2, 4, 6

[19] Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M Rehg. Video segmentation by tracking many figure-ground segments. In *Proceedings of the IEEE international conference on computer vision*, pages 2192–2199, 2013. 1, 2, 6

[20] Siyang Li, Bryan Seybold, Alexey Vorobyov, Xuejing Lei, and C-C Jay Kuo. Unsupervised video object segmentation with motion-based bilateral networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 207–223, 2018. 1, 2

[21] Liang Liu, Jiangning Zhang, Ruifei He, Yong Liu, Yabiao Wang, Ying Tai, Donghao Luo, Chengjie Wang, Jilin Li, and Feiyue Huang. Learning by analogy: Reliable supervision from transformations for unsupervised optical flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6489–6498, 2020. 7

[22] Runtao Liu, Zhirong Wu, Stella Yu, and Stephen Lin. The emergence of objectness: Learning zero-shot segmentation from videos. *Advances in Neural Information Processing Systems*, 34:13137–13152, 2021. 1, 2, 3, 4, 6, 7, 8

[23] Yong Liu, Ran Yu, Fei Yin, Xinyuan Zhao, Wei Zhao, Weihao Xia, and Yujiu Yang. Learning quality-aware dynamic memory for video object segmentation. In *European Conference on Computer Vision*, pages 468–486. Springer, 2022. 1, 6

[24] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 6

[25] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3623–3632, 2019. 1, 2

[26] Sabarinath Mahadevan, Ali Athar, Aljoša Ošep, Sebastian Hennen, Laura Leal-Taixé, and Bastian Leibe. Making a case for 3d convolutions for object segmentation in videos. *arXiv preprint arXiv:2008.11516*, 2020. 1

[27] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016. 4, 6

[28] Etienne Meunier, Anaïs Badoual, and Patrick Bouthemy. Em-driven unsupervised learning for efficient motion segmentation. *arXiv preprint arXiv:2201.02074*, 2022. 1, 6

[29] Bo Miao, Mohammed Bennamoun, Yongsheng Gao, and Ajmal Mian. Region aware video object segmentation with deep motion modeling. *arXiv preprint arXiv:2207.10258*, 2022. 1

[30] Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE transactions on pattern analysis and machine intelligence*, 36(6):1187–1200, 2013. 1, 2, 6

[31] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *Proceedings of the IEEE international conference on computer vision*, pages 1777–1784, 2013. 2, 6

[32] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016. 1, 2, 6

[33] Sucheng Ren, Wenxi Liu, Yongtuo Liu, Haoxin Chen, Guoqiang Han, and Shengfeng He. Reciprocal transformations for unsupervised video object segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15455–15464, 2021. 1, 2

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 2, 5

[35] Christian Schmidt, Ali Athar, Sabarinath Mahadevan, and Bastian Leibe. D2conv3d: Dynamic dilated convolutions for object segmentation in videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1200–1209, 2022. 1

[36] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000. 5

[37] D Sun, X Yang, MY Liu, and J Kautz. Pwc-net: Cnns for optical flow using pyramid. *Warping, and Cost Volume [J]*, 2017. 7

[38] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 2, 3, 4, 6, 8

[39] Wenguan Wang, Jianbing Shen, Ruigang Yang, and Fatih Porikli. Saliency-aware video object segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 40(1):20–33, 2017. 2, 6

[40] Yangtao Wang, Xi Shen, Yuan Yuan, Yuming Du, Maomao Li, Shell Xu Hu, James L Crowley, and Dominique Vaufreydaz. Tokencut: Segmenting objects in images and videos with self-supervised transformer and normalized cut. *arXiv preprint arXiv:2209.00383*, 2022. 5, 6

[41] Junyu Xie, Weidi Xie, and Andrew Zisserman. Segmenting moving objects via an object-centric layered representation. *arXiv preprint arXiv:2207.02206*, 2022. 1, 2, 4, 6, 8

[42] 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 *Proceedings of the European conference on computer vision (ECCV)*, pages 585–601, 2018. 6

[43] Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7177–7188, 2021. 1, 2, 4, 6

[44] Yanchao Yang, Brian Lai, and Stefano Soatto. Dystab: Unsupervised object segmentation via dynamic-static bootstrapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2826–2836, 2021. 2

[45] Yanchao Yang, Antonio Loquercio, Davide Scaramuzza, and Stefano Soatto. Unsupervised moving object detection via contextual information separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 879–888, 2019. 1, 2, 5, 6

[46] Mingmin Zhen, Shiwei Li, Lei Zhou, Jiaxiang Shang, Haoan Feng, Tian Fang, and Long Quan. Learning discriminative feature with crf for unsupervised video object segmentation. In *European Conference on Computer Vision*, pages 445–462. Springer, 2020. 1, 2

[47] Tianfei Zhou, Shunzhou Wang, Yi Zhou, Yazhou Yao, Jianwu Li, and Ling Shao. Motion-attentive transition for zero-shot video object segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13066–13073, 2020. 1, 2