LOCATE: Self-supervised Object Discovery via Flow-guided Graph-cut and Bootstrapped Self-training

Silky Singh
silsingh@adobe.com

Shripad Deshmukh
shdeshmu@adobe.com

Mausoom Sarkar
msarkar@adobe.com

Balaji Krishnamurthy
kbalaji@adobe.com

Media and Data Science Research
Adobe

Abstract

Learning object segmentation in image and video datasets without human supervision is a challenging problem. Humans easily identify moving salient objects in videos using the gestalt principle of common fate, which suggests that what moves together belongs together. Building upon this idea, we propose a self-supervised object discovery approach that leverages motion and appearance information to produce high-quality object segmentation masks. Specifically, we redesign the traditional graph cut on images to include motion information in a linear combination with appearance information to produce edge weights. Remarkably, this step produces object segmentation masks comparable to the current state-of-the-art on multiple benchmarks. To further improve performance, we bootstrap a segmentation network trained on these preliminary masks as pseudo-ground truths to learn from its own outputs via self-training. We demonstrate the effectiveness of our approach, named **LOCATE**, on multiple standard video object segmentation, image saliency detection, and object segmentation benchmarks, achieving results on par with and in many cases surpassing state-of-the-art methods. We also demonstrate the transferability of our approach to novel domains through a qualitative study on in-the-wild images. Additionally, we present extensive ablation analysis to support our design choices and highlight the contribution of each component of our proposed method. The project code is available at: https://github.com/silky1708/LOCATE

1 Introduction

Object segmentation, in its various forms, is a widely studied problem in computer vision [36]. The classic task finds critical applications across multiple domains, such as autonomous driving, augmented reality, human-computer interaction, video summarization etc. It is typically solved using deep neural networks trained on large annotated datasets created through enormous human efforts. Furthermore, dataset-specific considerations in model design limit the performance on real-world data.

To alleviate these issues with supervised segmentation, methods that work on the weaker forms of human supervision, for e.g., scribbles [33, 58, 59, 65, 79], clicks [2], image-level
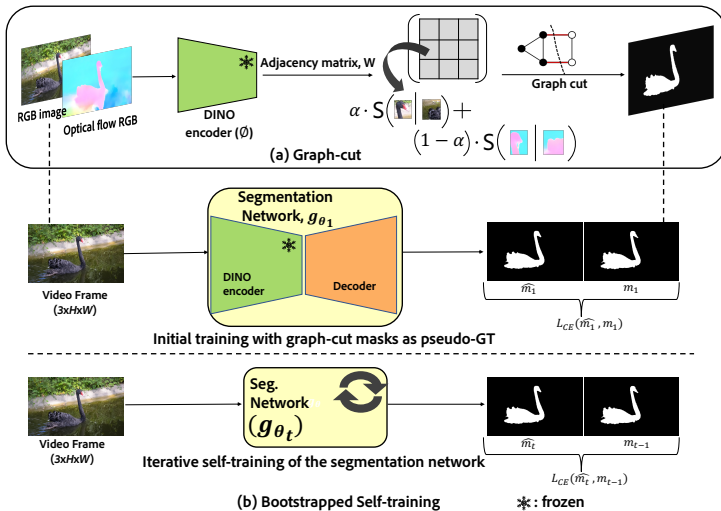
Figure 1: **Our proposed end-to-end pipeline (LOCATE) for self-supervised object discovery.** Our approach consists of two steps: (a) Graph-cut: we utilise a self-supervised ViT (DINO) to build a fully-connected graph on the image patches, incorporating both image and flow feature similarities to produce edge weights. We then perform graph-cut to produce a binary segmentation, followed by (b) Bootstrapped Self-training: we use graph-cut masks as pseudo-ground-truths to train a segmentation network; and subsequently train it iteratively on its own outputs.

tags [47, 59, 79], bounds [12], or semi-supervised segmentation techniques [12, 19, 20, 47, 49] were proposed that attempt segmentation in a setting where only a fraction of the image datasets are human-labelled. Nonetheless, both weak-supervised and semi-supervised techniques still rely on human supervision in some form. In this work, instead of relying on supervision in any form, we aim to design an end-to-end unsupervised pipeline that can be applied to any video dataset (real or synthetic) and transfers equally well on images. This opens up critical use cases for videos in the domains such as surgery, robotics, and exploratory research in deep seas and space.

A significant chunk of object segmentation approaches tackle the problem of Video Object Segmentation (VOS). In the literature [31, 44, 51, 76, 86], VOS is generally re-framed as a foreground-background separation problem, where the most salient object in the video *typically* forms the foreground. Interestingly, many of these approaches are based on the principle of common fate [75]: "things that move together belong together". This notion is captured by optical flow in videos – pixels that move with similar velocity, i.e., have similar optical flow, most likely belong to the same object. Some of the previous works [29, 41, 82] have advocated segmenting video objects using *only* optical flow as input. Nevertheless, we hypothesize that object appearance information in addition to the motion information can provide *stronger* cues for object discovery and relieve the network of ambiguities owing to occlusions or lack of motion.

So far, video and image segmentation approaches have primarily been studied disjointly. In the present work, we bridge this gap by training a segmentation network on individual frames of videos such that the resultant network can still be applied on standalone images. Our pipeline comprises of two steps. i) **Graph-cut:** For a given video frame, we build a graph(represented by adjacency matrix, W) with frame patches as the nodes and propose a linear combination of appearance (image) features with flow features to be used as edge

weights between the nodes. We perform graph-cut on the resultant matrix W to obtain a bi-partition of the image into foreground and background regions. ii) **Bootstrapped Self-training:** Next, we train a segmentation network using the acquired graph-cut masks as pseudo-ground truths. After training, the model can produce its own segmentation masks, which we then utilise for another round of training the same network from scratch. We continue this training procedure iteratively until convergence. This step allows the network to refine its predictions via self-training, and significantly boosts its ability to discover objects in videos and images. Combined together, we refer to this two-step approach by the name *LOCATE* (f**LO**w-guided graph-**C**ut **A**nd boo**T**strapped s**E**lf-training). Refer Fig. 1.

Using both the steps described above, we perform a comprehensive set of experiments, achieving state-of-the-art results across multiple video and image segmentation benchmarks. We believe the simplicity of our approach paves a new path for object segmentation in videos. The contributions of this work are summarized below:

1. We introduce **LOCATE**, a simple yet effective *fully unsupervised* pipeline to detect salient and foreground regions in videos as well as images.
2. Our approach achieves state-of-the-art results on unsupervised video object segmentation (VOS) benchmarks – DAVIS16 and SegTrackv2, while being comparable on FBMS59 dataset with state-of-the-art fully unsupervised approaches.
3. Our segmentation network trained only on videos fares equally well on images. When evaluated in a *zero-shot* manner, we are on par with or surpass current unsupervised state-of-the-art by large margins on image saliency detection (DUTS, ECSSD, OM-RON) and object segmentation (CUB, Flowers-102) benchmarks.
4. We provide a detailed analysis of each component of our architecture and justify the overall design choices of LOCATE.

2 Related Work

Unsupervised Video Object Segmentation (UVOS). UVOS aims to localize object(s) in a video sequence by separating foreground and background regions without any annotation [10, 16, 32, 37, 48, 50, 61, 76, 84, 85, 86]. The current state-of-the-art [10] uses a quadratic component flow model to predict regions containing simple motion patterns. Since optical flow provides rich visual cues for segmenting objects in a video, many works have modelled this problem as that of motion segmentation [24, 25, 26, 50, 57, 60, 85]. Layer extraction models [7, 23, 76, 86] decompose videos into layers of persistent motion groups, but with complex inference procedures. Brox and Malik [5] cluster long-term point trajectories based on pairwise distances to achieve temporally consistent object segmentation. Faktor and Irani [16] propose a voting scheme on co-occurring regions in the video sequence. Papazoglou and Ferrari [48] partition video frames into super-pixels and encourage foreground labels for super-pixels with independent motion. FusionSeg [15] combines appearance and motion modalities, while MG [82] focuses on motion segmentation via slot attention mechanism. AMD [35] uses two visual pathways to predict coherent object regions in RGB images and optical flow vectors. Another common approach is to model optical flow as piecewise parametric motion models, such as affine or quadratic [10, 21, 38, 41, 52].

Unsupervised Object Discovery (UOD). UOD aims to discover object information from images in an unsupervised manner, for e.g., bounding boxes, object masks etc. Recent works [18, 40, 56, 66, 67, 73, 91] have shown the potential of using features of a pre-trained network (supervised/self-supervised) for this task. Croitoru *et al*. [11] train a collection of

student networks to mimic the performance of a teacher for unsupervised object discovery. Caron *et al.* [6] have shown that semantic information emerges in a ViT trained in a self-supervised manner. Further building on this observation, LOST [56] proposes a seed expansion strategy to localize object(s) in a given image. We aim to solve a more complex task of object segmentation, not just localization. TokenCut [73] performs spectral decomposition on dense DINO [6] features to segment out the most salient object in an image. Motivated by the success of this approach, we incorporate motion feature similarities with appearance features to produce edge weights.

Object Segmentation and Image Saliency Detection. Existing works in unsupervised/weakly-supervised image saliency detection [42, 89, 90] focus on distilling hand-crafted priors into a deep network which is typically initialized with pretrained segmentation networks or image classifiers. In unsupervised object segmentation, broadly there are two categories of methods: generative or discriminative [22, 46] like contrastive learning. Generative approaches [1, 3, 4, 8] focus on generating foreground and background of an image separately that can then be combined to produce the final image. Differently, we utilise video datasets for a robust understanding of foreground objects, which can then be distilled to an image segmentation network.

Self-training. Self-training refers to propagating labels from a limited training set to a (large) set of unlabelled examples. This is achieved by training a network on the available labelled examples and treating its output on the unlabelled set as 'pseudo-labels' for subsequent round(s) of training. The label propagation is categorized as either hard [30, 80, 81] or soft assignment (often called knowledge distillation) [7] of labels. Recent works [64, 72, 88] have shown iterative self-training to be effective while training self-supervised segmentation models on noisy pseudo-labels. In contrast, to train our network, we keep the training set fixed in consecutive rounds of training, i.e., we don't 'propagate' labels to any held-out set.

3 Methodology

3.1 Preliminaries: Normalized Cut (NCut) for graph partitioning

Graph-cut, in its traditional form, is a widely studied problem [53]. Given a graph $G = (V, E)$ where V is a set of nodes and E is the set of edges, it seeks to partition G into two disjoint sets P and Q. The Ncut energy as proposed in [53] is minimized by the optimal graph-cut, as given by:

$$\text{Ncut}(P, Q) = \frac{U(P, Q)}{U(P, V)} + \frac{U(P, Q)}{U(Q, V)} \tag{1}$$

Here, U is a similarity measure between two sets. $U(P, Q) = \sum_{i,j} w(p_i, q_j)$, where p_i and q_j are the nodes in subgraphs P and Q respectively and $w(p_i, q_j)$ denotes the weight between these two nodes. A diagonal matrix, D can be formed with the diagonal elements given by: $d_i = \sum_j E_{ij}$. According to [53], minimizing Eq. (1) is equivalent to:

$$\min_x \text{Ncut}(x) = \min_y \frac{y^T (D - E) y}{y^T D y} \tag{2}$$

where $y \in \{1, -b\}^N$, b satisfies $y^T D 1 = 0$. Further, putting $z = D^{\frac{1}{2}} y$ in Eq. (2) results in the following optimization problem:

$$\min_z \frac{z^T D^{-\frac{1}{2}} (D - E) D^{-\frac{1}{2}} z}{z^T z} \tag{3}$$

which is equivalent to Rayleigh quotient [17]. Acc. to [17], the second smallest eigenvector

can be used to minimize the Ncut energy in Eq. (3). Taking $z = D^{\frac{1}{2}}y$, the second smallest eigenvector, say y_1, of the following eigensystem gives a solution to the Ncut problem.

$$(D - E)y = \lambda D y \tag{4}$$

To further bi-partition the graph, we use the average value of the second smallest eigenvector, $\overline{y_1} = \sum_j y_1^j$. Specifically, the bi-partitions P and Q are then given by $\{v_j \mid y_1^j \geq \overline{y_1}\}$ and $\{v_j \mid y_1^j < \overline{y_1}\}$ respectively.

3.2 Graph-cut for Video Object Discovery

Graph Construction. Consider a video frame $f \in \mathbb{R}^{H \times W \times 3}$, objects of which we wish to segment out. We create a fully-connected graph $G = (V, E)$, where V is obtained by dividing f into non-overlapping square patches of size $p_s \times p_s$, and E denotes the edges between these patches that quantify the similarity. More specifically, an adjacency matrix W underlying E comprises of elements $w_{ij} = S(v_i, v_j)$, where $S(\cdot)$ is the similarity measure between two given vertices (patches).

To incorporate motion signals in the edge weights, we propose to use similarities between image patch features and corresponding optical flow(in RGB format) [34] patch features, *in a linear combination*. We obtain the 'key' features from the last attention layer of a pre-trained DINO encoder [6]. Formally, we define S as:

$$S(v_i, v_j) = \alpha \cdot S'(\phi(v_i), \phi(v_j)) + (1 - \alpha) \cdot S'(\phi(\psi(v_i)), \phi(\psi(v_j))) \tag{5}$$

where $\alpha \in [0, 1]$, $\phi(\cdot)$ denotes the DINO encoder, $\psi(\cdot)$ denotes the RGB-optical flow estimator, i.e., model computing optical flow in 3-channel RGB format, and $S'(\cdot)$ is the cosine similarity function, given by $S'(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{||\vec{x}||_2 ||\vec{y}||_2}$. We obtain W using the S defined above, i.e., $W = [w_{ij}] = [S(v_i, v_j)]$.

$$w_{ij} \leftarrow \begin{cases} 1, & \text{if } w_{ij} \geq \tau \\ \varepsilon, & \text{otherwise} \end{cases} \tag{6}$$

Further, we normalize w_{ij}'s (Eq. 6) by thresholding on weight threshold hyper-parameter τ. The value of ε is set to 10^{-5}. Since $\varepsilon \neq 0$, we obtain a fully connected graph G on the patches of the frame f. This step is done independently for each frame in a video.

The inclusion of flow features in the graph construction differentiates our approach from other recent works [12, 34]. We demonstrate that our flow-guided graph-cut approach leads to significant gains in Sec. 4.

Foreground-Background separation via Graph-cut. We employ the Ncut algorithm (Sec. 3.1) to find the optimal bi-partition of image patches by solving for the second-smallest eigenvector of Eq. (4). To differentiate foreground patches from the background patches, we employ the heuristics proposed in [12]: (i) the foreground patches are more dominant, hence the patch corresponding to the maximum absolute value in the second smallest eigenvector should belong to the foreground, and (ii) foreground should occupy less than two corners in the image. We then assign the value 0 to the background patches, and 1 to the foreground patches to get a binary object segmentation mask for frame f. Also, since the foreground & background partitions are identified at *patch*-level, we perform a single step of post-processing using Conditional Random Fields (CRF) [28], to obtain binary segmentations at *pixel*-level.

3.3 Learning to segment via Bootstrapped Self-training

Although our flow-guided graph-cut approach is able to detect foreground objects extraordinarily well (Tab. 7), it still has limitations such as failing to identify small objects or to segment multiple disconnected foreground objects, etc. Thus, the segmentations obtained in Sec. 3.2 are noisy, nevertheless they still possess *objectness* information that can be distilled for more robust object discovery. To that end, we propose to use these as pseudo-ground-truths to train an image segmentation network, say $g_\theta : \mathbb{R}^{H \times W \times 3} \to \mathbb{R}^{H \times W \times 1}$, parameterized with weights θ. In our experiments, we use a pre-trained ViT (DINO) encoder with Mask-Former's [9] segmentation head as g_θ, but the method is applicable to all segmentation architectures. Moreover, the DINO encoder can easily be replaced with other (supervised/self-supervised) pre-trained backbones.

Given N video frames, $x_i \in \mathbb{R}^{H \times W \times 3}$, with corresponding graph-cut masks $m_i \in \mathbb{R}^{H \times W \times 1}$, we minimze the binary cross-entropy loss $\mathcal{L}_{CE}$ in the first round of training:

$$\theta_1^* = \arg\min_{\theta_1} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_{CE}(m_i, g_\theta(x_i)) \tag{7}$$

We observe that the initial round of training improves the quality of segmentation masks, especially at the boundaries, thus boosting the segmentation performance (Sec. 4.5). After training, we infer an initial set of segmentation masks for all the images in the training set using $g_{\theta_1^*}$. Next, we use these segmentation masks as pseudo-ground-truths for training the network again from scratch. We continue this process iteratively for multiple rounds, thus bootstrapping g_θ by training on its own outputs as pseudo-ground-truths – which we call *Bootstrapped Self-training*. Empirically, we find that the segmentation network is able to correct its own mistakes over multiple rounds until it saturates. For subsequent rounds of training, denoted by time steps $t \in \{2, 3, ...\}$, we optimize the following objective function:

$$\theta_t^* = \arg\min_{\theta_t} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_{CE}(g_{\theta_{t-1}^*}(x_i), g_{\theta_t}(x_i)) \tag{8}$$

In recent works [64, 77, 88], the self-training approach has been shown effective for self-supervised training of a segmentation network with noisy pseudo-labels. These methods incorporate more data in successive rounds of training; in contrast, we train the network on a fixed dataset in each round of self-training. Thus, to avoid overfitting, we train g_θ from scratch rather than initializing it from previous round's checkpoint.

4 Experiments and Results

4.1 Experimental Setup

Datasets. We evaluate our approach on standard VOS benchmarks: DAVIS16 [51], Seg-Trackv2 (STv2) [51] and FBMS59 [44]. These real-world datasets feature challenging conditions such as occlusions, motion blur, deformations, static objects etc. Following the procedure in [15], we merge multiple segmentation masks into one for SegTrackv2 and FBMS59 datasets for evaluation. For images, we evaluate our approach on the standard OMRON [83], DUTS [70], ECSSD [54], CUB [69] and Flowers-102 [43] datasets.

Graph-cut. We resize the images to a resolution of 480×848 for DAVIS16 and STv2 datasets, and 480×640 for the FBMS59 dataset before passing through DINO, based on the dominant aspect ratio of the dataset. We use DINO's ViT-B/8 architecture for the image & flow featurization (Sec. 4.6).

Metrics. For comparison on the VOS task, we compute the Jaccard metric, $\mathcal{J}$ and contour accuracy, $\mathcal{F}$. $\mathcal{J}$ is the mean intersection-over-union (mIoU) of the predicted and the ground-truth segmentation masks. For image segmentation tasks, we also compute accuracy and maxF_β scores, where $F_\beta = \frac{(1+\beta^2)\ \text{precision}\cdot\text{recall}}{\beta^2\cdot\text{precision}+\text{recall}}$. We use $\beta^2 = 0.3$. We report the sequence average of mIoU scores on DAVIS16, frame average for STv2 and FBMS59 datasets.

Optical Flow estimation. For optical flow estimation in the graph-cut step, we use an off-the-shelf model, ARFlow [54] which is trained in a completely unsupervised fashion on the synthetic Sintel dataset. We present an ablation study in Table 7.

Inference. At the time of inference, our network only takes a *single* image as input. Differently from existing works in VOS literature, we do not take additional inputs for e.g., optical flow, neither do we employ any test-time adaptations or post-processing techniques, rendering our approach very useful in practical applications. The runtime of our trained model is 0.42s (at an input resolution of 256×512).

4.2 Unsupervised Video Object Segmentation

Method	Supervision	Post-Processing	Inference RGB	Flow	DAVIS16 [ ] $\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	STv2 [ ] $\mathcal{J}\uparrow$	FBMS59 [ ] $\mathcal{J}\uparrow$
SAGE [ ]		✗	✓	✓	42.6	-	57.6	61.2
NLC [ ]		✗	✓	✓	55.1	-	67.2	51.5
CUT [ ]		✓	✓	✓	55.2	-	54.3	57.2
FTS [ ]	None	✗	✓	✓	55.8	-	47.8	47.7
AMD [ ]	(Fully Unsupervised)	✗	✓	✗	57.8	-	57.0	47.5
TokenCut [ ]		CRF	✓	✓	76.7	-	61.6	66.6
Ponimatkin *et al.* [ ]		CRF	✓	✓	80.2	77.5	74.9	70.0
LOCATE (Ours)		✗	✓	✗	**80.9**	**88.7**	**79.9**	68.8
MG [ ]	Sup. flow	✗	✗	✓	68.3	-	58.6	53.1
EM [ ]	Sup. flow	✗	✗	✓	69.3	70.7	55.5	57.8
MOD [ ]	Sup. flow	DINO-based TTA	✓	✗	79.2	-	69.4	66.9
SIMO [ ]	Synth. + Sup. flow	✗	✗	✓	67.8	-	62.0	-
CIS [ ]	Sup. flow	CRF + SP	✓	✓	71.5	-	62.0	63.5
ARP [ ]	Sup. flow & saliency	✗	✓	✓	76.3	71.1	57.2	59.8
OCLR‡ [ ]	Synth. + Sup. flow	DINO-based TTA	✓	✓	80.9	-	72.3	72.7
DS† [ ]	Sup. flow	✗	✓	✓	79.1	-	72.1	71.8
DyStaB** [ ]	Sup. feats.	CRF	✓	✓	80.0	-	74.2	73.2
GWM [ ]	Sup. flow	CRF + SP	✓	✗	80.7	86.9*	78.9	**78.4**

Table 1: **Quantitative comparison with unsupervised VOS approaches.** Our method, LOCATE, achieves or surpasses the state-of-the-art on DAVIS16 and SegTrackv2 benchmarks. (Methods above the dividing line are fully unsupervised approaches. † denotes optimization per video sequence. **DyStaB utilises supervised pre-training, ARP uses human supervision in the form of saliency maps. ‡ OCLR leverages manual annotations from large-scale Youtube-VOS data to generate synthetic data. Post-processing includes None, DINO-based test-time adaptation(TTA), CRF [23] or SP(Significant Post-processing techniques like multi-step flow, multi-crop ensemble, and temporal smoothing.) In terms of supervision, these methods either use None, Synth.- supervision from synthetic data, Sup. flow indicating supervised optical flow estimation models (e.g., RAFT), or Sup. feats. indicating usage of features from a model trained in a supervised manner. Inference indicates the inputs at the time of inference- optical flow or(and) RGB image.) Best scores are shown in **bold**. * scores calculated using the open-source implementation provided by respective authors.

Table 1 depicts the comparison between our method and the existing unsupervised VOS approaches. Our simple pipeline achieves state-of-the-art results on DAVIS16 and Seg-Trackv2 benchmarks. Compared to recent works, we also note that our approach is *fully* unsupervised and does not use *any* post-processing. Moreover, we do not require additional inputs like optical flow at the time of inference, thereby reducing the overhead and rendering our network applicable to image datasets as well. Despite the restricted conditions we operate in, we surpass all the recent works that utilise supervision in some form, and heavy post-processing techniques. Notably, some of the methods also utilise training on synthetic

Figure 2: Qualitative comparison of our method on VOS benchmarks – DAVIS16, SegTrackv2 and FBMS59. We note that our approach is able to produce high-quality object segmentations on multiple challenging scenes. Our method is able to detect (d) occluded object(s); (a),(e),(g) deformable objects with possible camouflage in (e); (a),(h),(j) complex backgrounds; (c) doesn't get confused with reflections; (d) can handle motion blur; (f) can segment multiple foreground objects but sometimes has difficulty as shown in (k); unable to identify parachute strings in (l), and over-segmentation due to scattered attention of DINO in (m).

data, while we do not use supervision in *any* form. On FBMS59, due to challenging scenes with multiple foreground objects, we do not perform as well, however we are still comparable to state-of-the-art fully-unsupervised methods. We show the qualitative results of our method in Fig. 2.
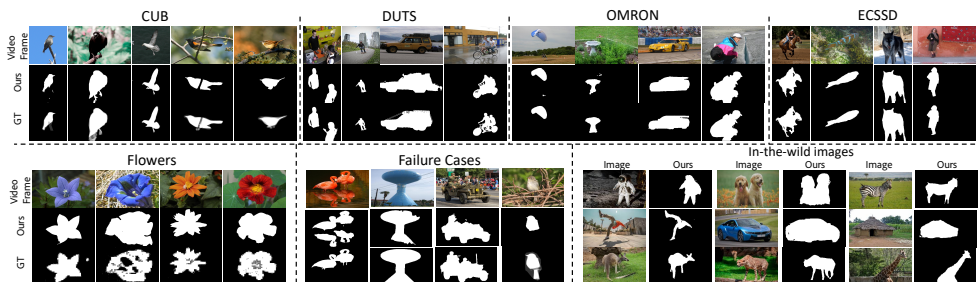
4.3 Image saliency detection and Object segmentation

Figure 3: Qualitative results of our method on image saliency (OMRON [83], DUTS [71], ECSSD [64]) and object segmentation benchmarks (CUB [69], Flowers-102 [43]). Our method applied in a *zero-shot* manner is able to detect foreground objects even with complicated backgrounds, multiple foreground objects, can segment deformable objects, possible camouflaged object, flowers with fine textures and intricate boundaries. However, it sometimes gets confused with realistic reflections, fails to completely segment inanimate or occluded objects.

We train our segmentation network on all *three* video datasets combined- DAVIS16, STv2 and FBMS59 until convergence and evaluate in a zero-shot manner on image segmentation benchmarks. A detailed quantitative comparison with existing state-of-the-arts is provided in Table 2. Although never trained on these datasets explicitly, we are on par with or surpass all the methods. We provide qualitative results on all the benchmarks in Fig. 3.

4.4 Transferability

In-the-wild foreground detection in images. We conduct a transferability study of our approach on in-the-wild images. We collect a set of random images downloaded from the internet, and evaluate the pre-trained segmentation network from Sec. 4.3 to detect foreground object(s) in these images. The qualitative results are shown in Fig. 3 – under the heading "In-the-wild images". Our model is able to segment foreground object(s) in all the challenging scenarios, demonstrating high transferability to unseen domains. Additional results are provided in the supplementary.

	CUB [■]			DUTS [■]			ECSSD [■]			OMRON [■]			Flowers [■]		
	Acc ↑	$\mathcal{J}$ ↑	maxF_β ↑	Acc ↑	$\mathcal{J}$ ↑	maxF_β ↑	Acc ↑	$\mathcal{J}$ ↑	maxF_β ↑	Acc ↑	$\mathcal{J}$ ↑	maxF_β ↑	Acc ↑	$\mathcal{J}$ ↑	maxF_β ↑
DeepUSPS [■]	-	-	-	77.3	30.5	42.5	79.5	44.0	58.4	77.9	30.5	41.4	-	-	-
Voynov et al. [■]	94.0	71.0	80.7	88.1	51.1	60.0	90.6	68.4	79.0	86.0	46.4	53.3	76.5	54.0	**76.0**
AMD [■]	-	-	-	-	-	60.2	-	-	-	-	-	-	-	-	-
Kyriazi et al. [■]	92.1	66.4	78.3	89.3	52.8	61.4	91.5	71.3	80.6	88.3	50.9	58.3	79.6	54.1	72.3
DeiT [■]	-	-	-	72.7	26.8	-	72.7	40.5	-	71.1	27.2	33.2	-	-	-
Kyriazi et al. [■]	-	**76.9**	-	-	51.4	-	-	73.3	-	-	56.7	-	-	-	-
DyStaB [■]	-	-	-	-	-	-	-	-	88.1	-	-	**73.9**	-	-	-
LOST [■]	-	-	-	88.7	57.2	69.7	91.6	72.3	83.7	81.8	48.9	57.8	-	-	-
TokenCut [■]	-	-	-	91.4	62.4	75.5	93.4	77.2	87.4	89.7	**61.8**	69.7	-	-	-
SelfMask [■]	-	-	-	92.3	62.6	-	**94.4**	**78.1**	-	90.1	58.2	-	-	-	-
GWM [■]	93.5	64.6	80.9	91.5	49.2	65.6	88.5	56.1	74.3	89.3	41.31	56.3	-	-	-
DependencyViT [■]	-	-	-	73.2	35.9	-	78.4	55.0	-	67.2	28.0	32.5	-	-	-
LOCATE (Ours)	**95.1**	70.8	**91.2**	**94.4**	**66.5**	**81.1**	93.3	75.9	**89.7**	**91.7**	51.3	66.7	**83.5**	**55.1**	73.2

Table 2: **Unsupervised image saliency detection and object segmentation.** Our model trained only on *videos* compares favorably with the state-of-the-art methods when applied in a zero-shot manner on image segmentation benchmarks. Best scores are shown in bold. Scores taken from [■, ■, ■].

Zero-shot evaluation on unseen videos. During iterative self-training (Sec. 3.3), we train the segmentation network from scratch in each round to ensure that the model does not overfit to video frames/images of the training dataset. We evaluate the transferability of such a network by first training it on some (video) dataset, and then testing it on another (video) dataset. Empirically, we train g_θ on the DAVIS16 dataset, and evalute it on SegTrackv2 and FBMS59 datasets in a zero-shot manner. We find that the network is equally effective on the unseen videos of test datasets. Further, the mean-IoU scores reported in Table 3 indicate that our bootstrapped self-training procedure does not deteriorate the transferability of the segmentation network.

Datasets	SegTrackv2	FBMS59
mIoU ($\mathcal{J}$ ↑)	65.56	67.83

Table 3: **Zero-shot evaluation on unseen videos.** Our segmentation network trained only on the DAVIS16 dataset is evaluated on the *val* split of STv2 and FBMS59 datasets in a zero-shot manner.

4.5 Trends in bootstrapped self-training

Datasets	Pseudo GT	Bootstrapping Rounds										
		Round 0	Round 1	Round 2	Round 3	Round 4	Round 5	Round 6	Round 7	Round 8	Round 9	Round 10
DAVIS16	Graph-cut masks	78.03	78.98$_{+0.95}$	79.91$_{+0.93}$	80.42$_{+0.51}$	80.56$_{+0.14}$	80.61$_{+0.05}$	80.66$_{+0.05}$	80.71$_{+0.05}$	**80.91**$_{+0.2}$	80.83$_{-0.08}$	80.74$_{-0.09}$
SegTrackv2		75.22	**79.94**$_{+4.72}$	79.53$_{-0.41}$	-	-	-	-	-	-	-	-
FBMS59		63.61	**65.64**$_{+2.03}$	65.30$_{-0.34}$	-	-	-	-	-	-	-	-
FBMS59	zero-shot	67.83	68.40$_{+0.57}$	68.51$_{+0.11}$	68.75$_{+0.24}$	68.77$_{+0.02}$	68.8$_{+0.23}$	**68.83**$_{+0.03}$	68.74$_{-0.09}$	-	-	-

Table 4: **Bootstrapped Self-training**. We bootstrap the segmentation network by iterative self-training on its own outputs. For FBMS59 zero-shot, we start with masks obtained from a network trained on DAVIS16 (Tab. 3).

In Table 4, we report the mean-IoU scores for bootstrapped self-training of segmentation networks trained on the DAVIS16, SegTrackv2 and FBMS59 datasets. The pseudo-ground-truths (scores reported under 'Round 0') are either obtained through our flow-guided graph-cut approach (Sec. 3.2), or in a zero-shot manner from a network trained on another dataset. We observe, in each case, that the quality of segmentation masks improves with each round of self-training until it saturates or declines in later rounds. Training on a collection of video frames enables the network to consolidate information across these examples and thus yields more accurate object segmentations.

4.6 Ablation Study

Encoder Model. We test our proposed framework with three different ViT image encoders: SelfPatch [■], DINO-v2 [■] and DINO [■], and the results are reported in Tab. 5. We find that SelfPatch performs comparable to DINO, while DINO-v2 degrades the performance.

We attribute this behaviour to the pre-training objectives of these encoders. SelfPatch primarily relies on patch-level contrastive training objectives, which leads to better patch-level representations. DINO-v2, on the other hand, loses the semantic information within an image because of its diversified pre-training procedure. Due to its superiority, we use the DINO encoder in all our experiments.

Encoder Model	Graph-cut ($\mathcal{J} \uparrow$)	Self-training (Round 1) ($\mathcal{J} \uparrow$)
DINO-v2 [15]	33.15	35.49
SelfPatch [57]	71.64	73.73
DINO [8]	78.03	78.98

Table 5: **Using our proposed framework with different ViT image encoders on the DAVIS16 dataset.** We use DINO-v2's ViT-B/14 and SelfPatch's available ViT-S/16 model.

DINO architecture variant. DINO's ViT-B and ViT-S variants produce features of dimension 768 and 384 respectively. ViT-B/8, the best performing DINO architecture for graph cut (refer Table 6), produces features of dimension 768 with an image patch size of 8.

DINO Architecture	ViT-S/8	ViT-S/16	ViT-B/8	ViT-B/16
DAVIS16($\mathcal{J} \uparrow$)	73.46	74.74	**76.76**	74.08

Table 6: **Ablation on the DINO architecture used for Graph-cut** (Refer Sec. 3.2). ViT-B/8 performs the best.

Optical flow estimation model. We show in Table 7 that our flow-guided graph-cut (Sec. 3.2) is not very sensitive to the choice of optical flow estimation model. Unsupervised ARFlow [34] performs comparable to supervised GMFlow [73] on all the datasets. However to respect fully unsupervised setting, we use ARFlow in all our experiments.

Optical Flow Model	DAVIS16($\mathcal{J} \uparrow$)	STv2($\mathcal{J} \uparrow$)	FBMS59($\mathcal{J} \uparrow$)
GMFlow [73]	76.76	**76.27**	**63.64**
ARFlow [34]	**78.03**	75.22	63.61

Table 7: **Ablation on flow model used in Graph-cut.** Interestingly, we find ARFlow, which is a fully unsupervised model, performs better than its supervised alternative on DAVIS16 and fares comparable on STv2 & FBMS59.

Hyper-parameters. We find that the similarity edge threshold $\tau = 0.25$ & the linear combination coefficient $\alpha = 0.7$ gives the best result (Tab. 8). Note that $\alpha = 1.0$ & $\alpha = 0.0$ represent the cases when *only* image features and *only* flow features are used for graph-cut respectively. The results confirm our hypothesis that the *combination* of motion and appearance information results in high-quality object segmentation masks.

α	0.0	0.2	0.4	0.5	0.6	0.7	0.8	0.9	1.0
DAVIS16($\mathcal{J} \uparrow$)	46.13	35.22	56.63	65.20	72.55	**75.96**	75.31	72.19	62.66

τ	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
DAVIS16($\mathcal{J} \uparrow$)	63.05	71.32	74.38	75.13	75.96	**76.76**	76.56	75.91	75.83

Table 8: **Hyper-parameter ablations.** Ablation on edge threshold τ & linear combination coefficient α, Sec. 3.2.

5   Conclusion

We present a comprehensive study of unsupervised object discovery in videos and images via graph-cut and bootstrapped self-training. Our simple approach outperforms strong baselines on multiple standard video and image benchmarks. Moreover, we qualitatively show that our approach is equally effective in the wild. Finally, we re-emphasize that motion information present in videos can be leveraged for more robust object discovery.

References

[1] Relja Arandjelović and Andrew Zisserman. Object discovery with a copy-pasting gan. *arXiv preprint arXiv:1905.11369*, 2019.

[2] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 549–565. Springer, 2016.

[3] Yaniv Benny and Lior Wolf. Onegan: Simultaneous unsupervised learning of conditional image generation, foreground segmentation, and fine-grained clustering. In *European Conference on Computer Vision*, pages 514–530. Springer, 2020.

[4] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. *Advances in Neural Information Processing Systems*, 32, 2019.

[5] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *European conference on computer vision*, pages 282–295. Springer, 2010.

[6] 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.

[7] Jason Chang and John W Fisher. Topology-constrained layered tracking with latent flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 161–168, 2013.

[8] Mickaël Chen, Thierry Artières, and Ludovic Denoyer. Unsupervised object segmentation by redrawing. *Advances in neural information processing systems*, 32, 2019.

[9] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021.

[10] 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:2205.07844*, 2022.

[11] Ioana Croitoru, Simion-Vlad Bogolin, and Marius Leordeanu. Unsupervised learning of foreground object segmentation. *International Journal of Computer Vision*, 127: 1279–1302, 2019.

[12] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1635–1643, 2015.

[13] Mingyu Ding, Yikang Shen, Lijie Fan, Zhenfang Chen, Zitian Chen, Ping Luo, Joshua B Tenenbaum, and Chuang Gan. Visual dependency transformers: Dependency tree emerges from reversed attention. *arXiv preprint arXiv:2304.03282*, 2023.

[14] 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.

[15] Suyog Dutt Jain, Bo Xiong, and Kristen Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3664–3673, 2017.

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

[17] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

[18] Olivier J Hénaff, Skanda Koppula, Evan Shelhamer, Daniel Zoran, Andrew Jaegle, Andrew Zisserman, João Carreira, and Relja Arandjelović. Object discovery and representation networks. *arXiv preprint arXiv:2203.08777*, 2022.

[19] Seunghoon Hong, Hyeonwoo Noh, and Bohyung Han. Decoupled deep neural network for semi-supervised semantic segmentation. *Advances in neural information processing systems*, 28, 2015.

[20] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. *arXiv preprint arXiv:1802.07934*, 2018.

[21] Allan Jepson and Michael J Black. Mixture models for optical flow computation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 760–761. IEEE, 1993.

[22] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9865–9874, 2019.

[23] Nebojsa Jojic and Brendan J Frey. Learning flexible sprites in video layers. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[24] Margret Keuper. Higher-order minimum cost lifted multicuts for motion segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4242–4250, 2017.

[25] 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.

[26] Margret Keuper, Siyu Tang, Bjoern Andres, Thomas Brox, and Bernt Schiele. Motion segmentation & multiple object tracking by correlation co-clustering. *IEEE transactions on pattern analysis and machine intelligence*, 42(1):140–153, 2018.

[27] 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.

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

[29] Hala Lamdouar, Weidi Xie, and Andrew Zisserman. Segmenting invisible moving objects. 2021.

[30] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013.

[31] 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.

[32] Siyang Li, Bryan Seybold, Alexey Vorobyov, Alireza Fathi, Qin Huang, and C-C Jay Kuo. Instance embedding transfer to unsupervised video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6526–6535, 2018.

[33] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3159–3167, 2016.

[34] 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.

[35] 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.

[36] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[37] 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.

[38] Aravindh Mahendran, James Thewlis, and Andrea Vedaldi. Self-supervised segmentation by grouping optical-flow. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[39] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Finding an unsupervised image segmenter in each of your deep generative models. *arXiv preprint arXiv:2105.08127*, 2021.

[40] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Deep spectral methods: A surprisingly strong baseline for unsupervised semantic segmentation and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8364–8375, 2022.

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

[42] Tam Nguyen, Maximilian Dax, Chaithanya Kumar Mummadi, Nhung Ngo, Thi Hoai Phuong Nguyen, Zhongyu Lou, and Thomas Brox. Deepusps: Deep robust unsupervised saliency prediction via self-supervision. *Advances in Neural Information Processing Systems*, 32, 2019.

[43] Maria-Elena Nilsback and Andrew Zisserman. Delving deeper into the whorl of flower segmentation. *Image and Vision Computing*, 28(6):1049–1062, 2010.

[44] 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.

[45] Maxime Oquab et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[46] Yassine Ouali, Céline Hudelot, and Myriam Tami. Autoregressive unsupervised image segmentation. In *European Conference on Computer Vision*, pages 142–158. Springer, 2020.

[47] George Papandreou, Liang-Chieh Chen, Kevin P Murphy, and Alan L Yuille. Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1742–1750, 2015.

[48] 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.

[49] Deepak Pathak, Philipp Krahenbuhl, and Trevor Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1796–1804, 2015.

[50] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2701–2710, 2017.

[51] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016.

[52] Georgy Ponimatkin, Nermin Samet, Yang Xiao, Yuming Du, Renaud Marlet, and Vincent Lepetit. A simple and powerful global optimization for unsupervised video object segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5892–5903, 2023.

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

[54] Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *IEEE transactions on pattern analysis and machine intelligence*, 38 (4):717–729, 2015.

[55] Gyungin Shin, Samuel Albanie, and Weidi Xie. Unsupervised salient object detection with spectral cluster voting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3971–3980, 2022.

[56] Oriane Siméoni, Gilles Puy, Huy V Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. Localizing objects with self-supervised transformers and no labels. *arXiv preprint arXiv:2109.14279*, 2021.

[57] Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *European conference on computer vision*, pages 438–451. Springer, 2010.

[58] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers. Normalized cut loss for weakly-supervised cnn segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1818–1827, 2018.

[59] Meng Tang, Federico Perazzi, Abdelaziz Djelouah, Ismail Ben Ayed, Christopher Schroers, and Yuri Boykov. On regularized losses for weakly-supervised cnn segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 507–522, 2018.

[60] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning video object segmentation with visual memory. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4481–4490, 2017.

[61] Pavel Tokmakov, Cordelia Schmid, and Karteek Alahari. Learning to segment moving objects. *International Journal of Computer Vision*, 127(3):282–301, 2019.

[62] Philip HS Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 356(1740):1321–1340, 1998.

[63] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablay-rolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[64] Wouter Van Gansbeke, Simon Vandenhende, and Luc Van Gool. Discovering object masks with transformers for unsupervised semantic segmentation. *arXiv preprint arXiv:2206.06363*, 2022.

[65] Paul Vernaza and Manmohan Chandraker. Learning random-walk label propagation for weakly-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7158–7166, 2017.

[66] Huy V Vo, Patrick Pérez, and Jean Ponce. Toward unsupervised, multi-object discovery in large-scale image collections. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*, pages 779–795. Springer, 2020.

[67] Van Huy Vo, Elena Sizikova, Cordelia Schmid, Patrick Pérez, and Jean Ponce. Large-scale unsupervised object discovery. *Advances in Neural Information Processing Systems*, 34:16764–16778, 2021.

[68] Andrey Voynov, Stanislav Morozov, and Artem Babenko. Object segmentation without labels with large-scale generative models. In *International Conference on Machine Learning*, pages 10596–10606. PMLR, 2021.

[69] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[70] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 136–145, 2017.

[71] 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.

[72] Xudong Wang, Rohit Girdhar, Stella X Yu, and Ishan Misra. Cut and learn for unsupervised object detection and instance segmentation. *arXiv preprint arXiv:2301.11320*, 2023.

[73] Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L Crowley, and Dominique Vaufreydaz. Self-supervised transformers for unsupervised object discovery using normalized cut. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14543–14553, 2022.

[74] 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.

[75] Max Wertheimer. Untersuchungen zur lehre von der gestalt. *Gestalt Theory*, 39(1): 79–89, 2017.

[76] Junyu Xie, Weidi Xie, and Andrew Zisserman. Segmenting moving objects via an object-centric layered representation. In *Advances in Neural Information Processing Systems*, 2022.

[77] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020.

[78] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8121–8130, 2022.

[79] Jia Xu, Alexander G Schwing, and Raquel Urtasun. Learning to segment under various forms of weak supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3781–3790, 2015.

[80] Qiantong Xu, Tatiana Likhomanenko, Jacob Kahn, Awni Hannun, Gabriel Synnaeve, and Ronan Collobert. Iterative pseudo-labeling for speech recognition. *arXiv preprint arXiv:2005.09267*, 2020.

[81] I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. *arXiv preprint arXiv:1905.00546*, 2019.

[82] 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.

[83] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3166–3173, 2013.

[84] 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.

[85] 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.

[86] Vickie Ye, Zhengqi Li, Richard Tucker, Angjoo Kanazawa, and Noah Snavely. Deformable sprites for unsupervised video decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2657–2666, 2022.

[87] Sukmin Yun et al. Patch-level representation learning for self-supervised vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[88] Andrii Zadaianchuk, Matthaeus Kleindessner, Yi Zhu, Francesco Locatello, and Thomas Brox. Unsupervised semantic segmentation with self-supervised object-centric representations. *arXiv preprint arXiv:2207.05027*, 2022.

[89] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, Lihe Zhang, Mingyang Qian, and Yizhou Yu. Multi-source weak supervision for saliency detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6074–6083, 2019.

[90] Jing Zhang, Tong Zhang, Yuchao Dai, Mehrtash Harandi, and Richard Hartley. Deep unsupervised saliency detection: A multiple noisy labeling perspective. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9029–9038, 2018.

[91] Runsheng Zhang, Yaping Huang, Mengyang Pu, Jian Zhang, Qingji Guan, Qi Zou, and Haibin Ling. Object discovery from a single unlabeled image by mining frequent itemsets with multi-scale features. *IEEE Transactions on Image Processing*, 29:8606–8621, 2020.