

Guess What Moves: Unsupervised Video and Image Segmentation by Anticipating Motion

Subhabrata Choudhury*
subha@robots.ox.ac.uk

Laurynas Karazija*
laurynas@robots.ox.ac.uk

Iro Laina
iro@robots.ox.ac.uk

Andrea Vedaldi
vedaldi@robots.ox.ac.uk

Christian Rupprecht
chrisr@robots.ox.ac.uk

Visual Geometry Group
University of Oxford
Oxford, UK

Abstract

Motion, measured via optical flow, provides a powerful cue to discover and learn objects in images and videos. However, compared to using appearance, it has some blind spots, such as the fact that objects become invisible if they do not move. In this work, we propose an approach that combines the strengths of motion-based and appearance-based segmentation. We propose to supervise an image segmentation network with the pretext task of predicting regions that are likely to contain simple motion patterns, and thus likely to correspond to objects. As the model only uses a single image as input, we can apply it in two settings: unsupervised video segmentation, and unsupervised image segmentation. We achieve state-of-the-art results for videos, and demonstrate the viability of our approach on still images containing novel objects. Additionally we experiment with different motion models and optical flow backbones and find the method to be robust to these change. Project page and code available at <https://www.robots.ox.ac.uk/~vgg/research/gwm>.

1 Introduction

The motion of objects in a video can be detected by methods such as optical flow and used to discover and segment them. A key benefit is that optical flow is object-agnostic: because it relies on low-level visual properties, it can extract a signal even before the objects are discovered, and can thus be used to establish an understanding of objectness.

The potential of motion as a cue is epitomized in *video segmentation* problems, where the input is a generic video sequence and the task is to extract the main object(s) in the

video. In fact, some methods [57, 90] adopt a motion-*only* approach to video object segmentation, arguing that motion patterns are much easier to model and interpret than appearance. However, this approach ignores appearance cues and is ‘blind’ to stationary objects.

Instead, we propose to use motion as *supervision* to discover objects in videos *and* still images without the need for manual annotations. We observe that different objects tend to generate distinctive optical flow patterns which can be well approximated by small parametric models, such as affine or quadratic. We use this fact to train a segmentation network that, given a *single RGB frame* as input, predicts *which image regions* are likely to contain such patterns. The idea is that these regions would then separate the objects from the background.

This approach has several useful properties. First, while motion is used for supervising the network, the latter implicitly learns the appearance of the objects, regularizing the segmentation. Second, because the network works with a single image as input, it does not observe the motion directly. The model must anticipate what *could* move, extracting objects even if they are not in motion. Third, the network avoids predicting the objects’ motion directly, which is a highly-ambiguous task given a single image as input; instead, it predicts only the support regions of the motion patterns, and the training loss measures the compatibility of such regions with the observed motion according to the assumed motion model.

While we are not the first to consider motion as a cue for decomposing an image into objects, our particular way of modeling motion is simple and versatile, and allows two application modes of our approach. First, we consider *internal learning for unsupervised motion segmentation* [80]. Given one or more videos as input (without labels), we optimize a network, as described above, to output a segmentation of the videos, effectively ‘observing’ motion via backpropagation. Our approach achieves state-of-the-art performance on standard benchmarks for unsupervised motion segmentation [91, 92].

The second mode is *transductive learning for unsupervised image segmentation*, which is intended to assess the generalization capabilities of our model as an image segmenter. In this case, the network is first trained on a number of training videos and then evaluated on a disjoint set of images. Since only appearance information is available at test time, the problem solved is not motion segmentation, but image segmentation. In this scenario, our model segments novel objects not observed during training, demonstrating the viability of our approach.

2 Related Work

Our work aims to combine motion and appearance cues for unsupervised object discovery, in that motion can be used as a cue to learn a general object segmenter for both videos and images. As such, there exist several related areas in literature, which we review next.

Unsupervised Video Object Segmentation. The aim of video object segmentation (VOS) is to densely label objects present in a video. Current VOS benchmarks [45, 63, 66] usually define the problem as foreground-background separation, where the foreground comprises the most salient objects. Efforts to reduce the amount of supervision follow two main directions, semi-supervised and unsupervised VOS. Semi-supervised methods require manual annotations for the object(s) of interest in an initial frame during inference; the goal is to re-localize these objects across the video [13]. Unsupervised VOS aims to discover object(s) of interest without the initial targets [25, 30, 46, 53, 55, 78]. However, most unsupervised VOS methods use, in fact, some form of supervised pre-training on external data.

Motion Segmentation. In videos, the background is usually relatively static whereas objects in the scene have independent motion, thus providing a strong ‘objectness’ signal. Thus, many works approach unsupervised video object segmentation as a motion segmentation problem. Several earlier methods address this problem by grouping point trajectories [10, 40, 41, 51, 52, 76], motion boundaries [63], voting [23] and layered models [13, 56]. More recently, Lamdouar et al. [44], Xie et al. [89] train motion models on generated scenes with synthetic 2D objects and generalize to real videos. CIS [22] proposes an adversarial framework, where an inpainter is tasked with predicting the optical flow of a segment based on context, while the generator aims to create segments with zero mutual information such that the context becomes uninformative. DyStaB [93] extends CIS using the segmentation output of a dynamic model to bootstrap a static one. In contrast to our method, this yields two separate models to choose from based on the application (*i.e.*, video or static image segmentation). Instead, AMD [50] employs a single model with separate appearance and motion ‘pathways’ and performs unsupervised test-time adaptation for video segmentation. Finally, MG [60] abandons the appearance pathway altogether, directly segmenting optical flow inputs with a Slot Attention-like architecture [51].

Closer to our approach, another line of work uses various motion models to group image regions. Early methods [61, 79] consider mixture models of flow to account for the fact that a region may contain multiple motion patterns. Another line of work [6, 7] segments object translation directions from motion angle field obtained by correcting for estimated rotation of the camera. Mahendran et al. [52] employ an affine flow model, using the entropy of flow magnitude histograms for loss to deal with noisy flow in real world. Meunier et al. [57] consider affine and quadratic motion models, however their method uses flows as input which makes it suitable only for videos during inference.

Unsupervised Image Segmentation. While we use motion as a learning signal, our method yields a general-purpose image segmentation network, separating an image into foreground and background, without using ground truth masks for supervision. Early work in unsupervised image segmentation makes use of hand-crafted priors, *e.g.* color contrast [19, 36], while some recent methods also combine handcrafted heuristics to generate pseudo-masks and use them to train using deep networks [60, 97, 98]. Others address this problem via mutual information maximization between different views of the input [42, 54]. A recently emerging line of work [9, 8, 17, 38, 56, 81] explores generative models to obtain segmentation masks. Many of them [9, 8, 17, 58] are based on the idea of generating foreground and background as separate layers and combine them to obtain a real image. Others [56, 81] analyze large-scale unsupervised GANs (*e.g.* BigGAN [9]) and find implicit foreground-background structure in them to generate a synthetic annotated training dataset. Alternative line of work explores feature maps of self-supervised Vision Transformers, such as DINO [42]. For example, STEGO [28] supports segmenting multiple *classes* in an image, performing *semantic* segmentation, by distilling features and class centroids from DINO. In Melas-Kyriazi et al. [55] and TokenCut [85], authors model image patches with an affinity graph based on DINO feature alignment and perform further analysis on this graph to extract masks. Shin et al. [71] cluster features of a variety of self-supervised backbones to produce candidate masks, using them to train a segmenter. Instead, our model is trained on video data using optical flow as a supervisory signal. However, since it only requires a single image as input at test time, we show that our method is applicable to this task, providing an alternative approach to unsupervised object segmentation.

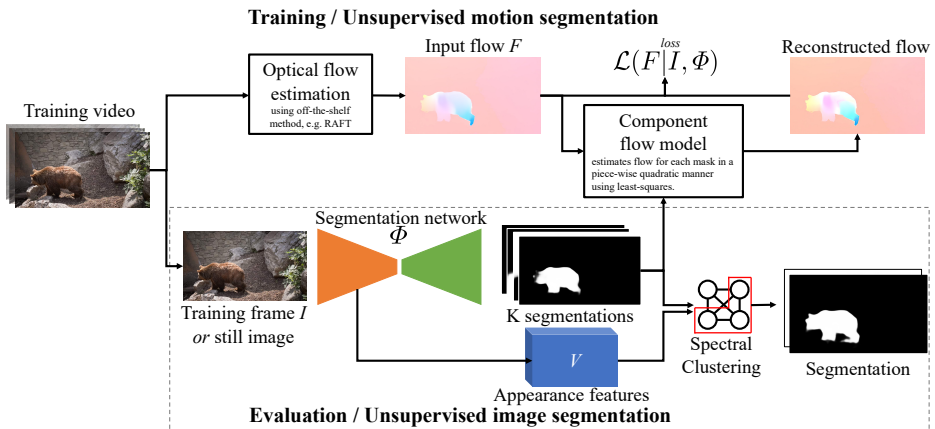


Figure 1: **Model Diagram.** We train a segmentation network to partition an image into K components without manual annotations. Our model is trained using individual frames from video as input and pre-computed optical flow as supervision. The predicted segments are used to approximate the input flow with piecewise quadratic flow models and the training loss is formulated as the error between the reconstructed and the input flow. Appearance features from the backbone are used to merge the predicted K segments into foreground and background components. Motion information is not required at test time and inference can be performed on still images. Optical flow is colored for visualization only.

Unsupervised Object Discovery. While the above methods often aim to segment the most salient object(s) in an image, unsupervised multi-object segmentation explores the problem of decomposing a scene into parts, which typically include each individual foreground object and the background. The usual approach is to learn structured object-centric representations, *i.e.* to model the scene with latent variables (slots) operating on a common representation [10, 20, 22, 23, 24, 27, 33, 47, 51, 72]. While these methods are image-based, extensions to video also exist [3, 5, 21, 34, 37, 42, 43, 53, 59, 73, 97]. These methods often operate in an auto-encoding fashion with inductive bias to separate objects derived from a reconstruction bottleneck [10], that is often dependent on the architecture and the latent variable model. We similarly impose a reconstruction bottleneck on the flow but use a simple model grounded in projective geometry, with a known closed-form solution. It is also important to note that unsupervised *multi*-object segmentation appears to be significantly more challenging, with current methods exploiting the simplicity of synthetic scenes [26, 35], while struggling on more realistic data [39]. Recently, Bao et al. [0] explore an extension of Slot Attention [51], guided by an external supervised motion segmentation algorithm, to real-world data. However, due to the difficulty of the problem, they operate in a constrained domain (autonomous driving) and consider only a limited number of object categories. We instead focus on wide variety of categories and settings encountered in common video segmentation datasets and consider both motion and appearance jointly.

3 Method

In this paper, we present a method that uses motion anticipation to discover and segment objects in images without the need for human annotations (overview in Fig. 1). We use

optical flow from video sequences as supervision for this problem. However, rather than predicting the flow directly, we task a general image segmentation network to predict image regions where motion may be explained by a simple coherent model. Such regions should align with optical flow patterns produced by objects that *could* move (but do not have to).

3.1 Segmentation by Motion Anticipation

Let $I \in \mathbb{R}^{3 \times H \times W} = (\mathbb{R}^3)^\Omega$ be an RGB image defined on a lattice $\Omega = \{1, \dots, H\} \times \{1, \dots, W\}$. Assume that the image is a frame in a video sequence and let $F \in (\mathbb{R}^2)^\Omega$ be the corresponding optical flow (extracted from the video by means of an off-the-shelf optical flow network, such as RAFT [14]). The goal is to decompose the image into K components (or regions), which is a classic segmentation problem. Hence, we learn a segmentation network $\Phi(I) \in ([0, 1]^K)^\Omega$ that, given the image I as input, assigns each pixel u to one of K components in a soft manner, with probabilities:

$$P(m_u = k \mid I, \Phi) = [(\Phi(I))_k]_u, \quad u \in \Omega, k \in \{1, \dots, K\}. \quad (1)$$

$m_u = k$ in Eq. (1) denotes the predicted mask corresponding to component k indexed by u . In particular, we seek to separate the foreground and background, for which one may choose $K = 2$, although as we show later (Section 3.2), this need not be the case.

More specifically, we train Φ to partition pixels according to the Gestalt principle of common fate [14, 82]. This is done by associating each region $k \in \{1, \dots, K\}$ to a model θ_k of the optical flow observed within it. That is, the optical flow corresponding to an input frame can be approximated by piece-wise parametric models, representing the motion or *flow pattern*, of each component independently. According to the common fate principle, pixels within the same region are expected to exhibit *coherent* motion.

A variety of motion models exist for describing the 2D flow of an object ([82, 87]). These are generally of the form $F_u \approx Au + b$, where parameters A, b can be recovered by solving a system of linear equations. One common choice is an affine model (where $u = [x, y]$ are pixel coordinates), which is sufficient if objects are smaller and further away from camera. The affine model, however, struggles if the depth of an object varies significantly resulting in more complex flow patterns. To factor out unknown depth information, each object can be modeled as a plane with a quadratic 8-parameter model [87]. Here, we allow for more complex geometry than planes, by using a simplified 12-parameter quadratic model $\theta_k = (A_k, b_k)$ with $A_k \in \mathbb{R}^{2 \times 5}$ and $b_k \in \mathbb{R}^2$ per region k . In this case, $u = [x, x^2, y, y^2, xy] \in \mathbb{R}^5$ includes quadratic and mixed terms of the pixel coordinates to model quadratic dependencies. The 12-parameter model also allows treating each flow direction independently. We assume that the model predicts the flow up to isotropic i.i.d. Gaussian noise, which results in a simple L^2 fitting loss:

$$-\log p(F_u \mid \theta_k) \propto \|F_u - A_k u - b_k\|^2. \quad (2)$$

Summing over all pixels, learning minimizes the energy function:

$$\mathcal{L}(F \mid \theta, I, \Phi) \propto \sum_{u \in \Omega} \sum_k \|F_u - A_k u - b_k\|^2 \cdot p(m_u = k \mid I, \Phi). \quad (3)$$

In the expression above, we *do not* know the flow parameters θ_k as the network only predicts the regions' extent. Instead, we *min-out* the parameters θ_k in the loss itself and compute

$$\mathcal{L}(F \mid I, \Phi) = \min_{\theta_{k \in \{1, \dots, K\}}} \mathcal{L}(F \mid \theta, I, \Phi). \quad (4)$$

The energy in Eq. (2) is quadratic in θ_k , resulting in a weighted least squares problem that can be efficiently solved in closed form (see supplementary material).

Our model is learned from a large collection \mathcal{T} of video frame-optical flow pairs (I, F) , minimizing the empirical risk:

$$\Phi^* = \operatorname{argmin}_{\Phi} \frac{1}{|\mathcal{T}|} \sum_{(I, F) \in \mathcal{T}} \mathcal{L}(F | I, \Phi) \quad (5)$$

3.2 Over-segmentation

While the 12-parameter model is more powerful than an affine one, it is still not sufficient to model arbitrary flow patterns. In complex scenes that contain foreground and background clutter, we often observe motion parallax effects. Additionally, non-rigid objects and self-occlusions can result in complex flow patterns within the object that are not captured accurately by the quadratic model.

To account for such complexity, we propose to *over-segment* the input image into $K > 2$ regions. Over-segmentation enables the model to use additional regions to explain several moving objects and to approximate varying parts of a single non-rigid object as well as motion parallax. To achieve a binary segmentation output, one needs a criterion to merge a number of predicted regions down to foreground and background.

We devise a criterion based on *appearance* cues to avoid the ambiguity associated with merging regions based on motion. To this end, we use a pre-trained self-supervised image encoder, such as DINO-ViT [14], to obtain dense features for the input image and merge the segments predicted by Φ based on feature similarity. Formally, let V_u denote the feature vector of pixel u obtained by the self-supervised encoder. Then, $\bar{V}_k = \sum_u V_u p(m_u = k | I, \Phi) / \sum_v p(m_v = k | I, \Phi)$ is the average feature vector for segment k , where pixels are weighed by their probability with which they belong to the segment. We compute the pairwise similarities of different regions via an affinity matrix $\Pi \in \mathbb{R}^{K \times K}$, where entries corresponding to segments i and j are set as

$$(\Pi)_{ij} = \max \left(\varepsilon, \left\langle \frac{\bar{V}_i}{\|\bar{V}_i\|_2}, \frac{\bar{V}_j}{\|\bar{V}_j\|_2} \right\rangle \right), \quad (6)$$

where only feature vectors pointing in the same direction are considered and $\varepsilon = 10^{-12}$ is a small constant that keeps the graph connected. We then perform spectral clustering [16, 55, 59] into two components using the affinity Π .

3.3 Two Scenarios: Motion vs Image Segmentation

We experiment with two modes of application of our model. The first scenario is *internal learning for unsupervised video segmentation*, where the network is evaluated on the same video sequences that have been used for optimization. This is effectively an unsupervised motion segmentation algorithm because the network not only receives as input appearance information, but incorporates motion information via backpropagation, observing indirectly optical flow too. While not explicitly stated in the respective papers, prior motion segmentation works such as [57, 90] also operate in this mode, while directly observing moving objects, often using optical flow as input.

The second scenario is *transductive learning for image segmentation*. In this case, the network is first trained using a number of unlabelled videos, and then used for single-image

	Inf. Input	Input	Flow	Runtime	DAVIS	STv2	FBMS	
	RGB	Flow	Resolution	Method	sec ↓	$\mathcal{J} \uparrow$	$\mathcal{J} \uparrow$	$\mathcal{J} \uparrow$
[84] SAGE	✓	✓	–	LDOF [40]	0.9	42.6	57.6	61.2
[25] NLC	✓	✓	–	SIFTFlow [43]	11	55.1	67.2	51.5
[44] CUT	✓	✓	–	LDOF [40]	103	55.2	54.3	57.2
[65] FTS	✓	✓	–	LDOF [40]	0.5	55.8	47.8	47.7
[72] CIS	✓	✓	192 × 384	PWCNet [47]	0.1	59.2	45.6	36.8
[61] AMD	✓	✗	128 × 224	✗	–	57.8	57.0	47.5
[90] MG	✗	✓	128 × 224	RAFT [42]	0.012	68.3	58.6	53.1
[67] EM	✗	✓	128 × 224	RAFT [42]	–	69.3	55.5	57.8
[85] OCLR	✓	✓	480 × 832	RAFT [42]	–	78.9	71.6	68.7
[92] DS [‡]	✓	✓	240 × 426	RAFT [42]	1800 (22.5) [‡]	79.1	72.1	71.8
Ours (UNet)	✓	✗	128 × 224	RAFT [42]	0.027	78.3	76.8	72.0
Ours (MaskFormer)	✓	✗	128 × 224	RAFT [42]	0.059	79.5	78.3	77.4
[92] CIS [†]	✓	✓	192 × 384	PWCNet [47]	11	71.5	62.0	63.5
[92] DyStaB ^{†*}	✓	✓	192 × 384	RAFT [42]	–	80.0	74.2	73.2
Ours[†] (w/ CRF)	✓	✗	128 × 224	RAFT [42]	3.73	80.7	78.9	78.4

Table 1: **Unsupervised video segmentation** on DAVIS2016, SegTrack-v2 (STv2), and FBMS59.

† denotes the usage of CRFs and other extra significant post-processing (e.g., multi-step flow, multi-crop, temporal smoothing for CIS [92]). ‡ DS is optimized per sequence; authors report 30 min training time for 80-frame video. * DyStaB utilises supervised pre-training.

foreground object segmentation on an *independent* validation/test set of still images. In this scenario, motion is only used as a supervisory signal: when the network is applied at test time, motion is not considered anymore and the network operates purely as an image-based segmenter. As for any transductive learning setting, the goal is to assess the generalization performance of the network on new images.

4 Experiments

As discussed above, our formulation allows us to evaluate our method in two settings: video object segmentation and general image/object segmentation. We show that learning a network that *guesses what moves* not only results in state-of-the-art performance in video segmentation, but also generalizes to image segmentation without further training.

4.1 Experimental Setup

Architecture. Our formulation enables us to use any standard image segmentation architecture for the model Φ . This has two main benefits: while training the model needs optical flow (and thus video data), inference can be performed on single images alone just like any image segmentation method. Second, using a standard architecture allows us to benefit from (self-)supervised pretraining, ensuring better convergence and broader generalization. We experiment with both convolutional and transformer-based architectures.

Datasets. For the *video segmentation* task, we use three popular datasets: DAVIS2016 (DAVIS) [66], SegTrackV2 (STv2) [45], as well as FBMS [63]. For the *image segmentation* task, we consider the Caltech-UCSD Birds-200 (CUB) dataset [87] and three saliency detection benchmarks: DUTS [83], ECSSD [40], and DUT-OMRON [90].

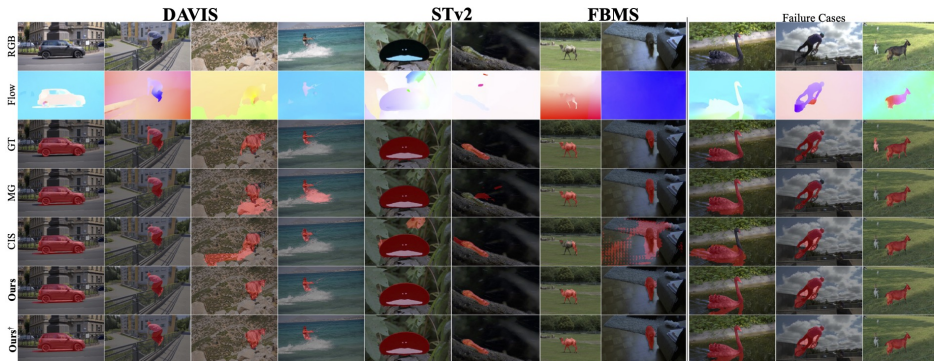


Figure 2: **Qualitative Comparison on DAVIS, STv2, and FBMS.** [†] – indicates use of CRF. Our method correctly segments objects in challenging conditions including strong parallax (2nd, 3rd seq.), small objects (4th), background motion (5th), camouflaged appearance (6th), non-rigid motion (7th) or no motion at all (8th seq.). In the failure cases, our method is confused by ripples and reflection in the water, the front wheel rotating in a different direction and multiple disconnected objects.

Optical Flow. Our method derives its learning signal from optical flow. We estimate optical flow for all frames on DAVIS, STv2, and FBMS following the practice of MotionGrouping [91]. We employ RAFT [77] (supervised) using the original resolution for our main experiments. Please see the supplement for experiments with other flow methods.

Training Details. We use MaskFormer [18] as our segmentation network, and use only the segmentation head. For the backbone and appearance features V , we leverage a ViT-B transformer, pre-trained on ImageNet [53] in a self-supervised manner using DINO [42] to avoid any external sources of supervision. We set the number of components to $K = 4$ unless otherwise noted. Please see the supplement for all details and hyper-parameter settings.

4.2 Unsupervised Video Segmentation

In Table 1 we report our performance on the DAVIS, STv2, and FBMS datasets and compare to other unsupervised video segmentation approaches. Our method achieves state-of-the-art performance, even without CRF post-processing. Fig. 2 provides a qualitative comparison of the results. Our model provides better segmentation with sharper boundaries despite complex non-rigid motion, parallax effects or lack-of-motion. However, on challenging scenarios our method still struggles to segment small details or non-connected instances.

Our method is not restricted to a specific segmentation architecture. To investigate, MaskFormer is replaced with a simple convolutional U-Net architecture [67], as in EM [67], and trained from scratch for a fair comparison. The U-Net based model achieves comparable results on DAVIS and FBMS and 76.8 on STv2 (Table 1), outperforming earlier methods even without transformers.

4.3 Flow Model and Number of Components

Using DAVIS, we now study the effectiveness of the individual components of the method. In Table 2 we evaluate the performance of the model under different flow models: constant,

Flow Model	K	DAVIS ($\mathcal{J}\uparrow$)	DAVIS ($\mathcal{J}_{\text{oracle}}\uparrow$)
Constant ($A = 0$)	4	76.8	77.7
Affine ($u = [x, y]$)	4	77.1	78.8
Quadratic (Eq. (2))	4	79.5	81.5
Quadratic (Eq. (2))	2	74.5	74.5
Quadratic (Eq. (2))	3	77.8	79.5
Quadratic (Eq. (2))	4	79.5	81.5
Quadratic (Eq. (2))	5	76.0	79.9

Table 2: **Flow Model and Number of Components.** We ablate the choice of flow model and the number of components K . More complex flow models improve performance, and over-segmentation helps until the assignment problem between components and the final binary segmentation becomes too difficult at $K = 5$. To evaluate the quality of the clustering of components we also report the oracle clustering performance as an upper bound.

affine, and quadratic. We find that more complex models lead to improved performance, likely due to the fact that many scenes in the DAVIS benchmark are highly dynamic with complex objects and backgrounds. Additionally, in the same table we evaluate how the number of components, K , influences the final performance after clustering. With $K = 2$ the model directly performs foreground-background separation but needs to model each with a single component which is often difficult, *e.g.* due to complex motions of deformable objects and/or parallax effects. Increasing the number of components is beneficial up to $K = 4$, after which the assignment problem from over-segmentation to foreground and background becomes too difficult for simple spectral clustering. This can be seen by evaluating the segmentation performance under an optimal oracle assignment of the components to foreground and background (oracle column in Table 2). In all cases $K \leq 4$, spectral clustering nearly reaches oracle performance.








	CUB			DUTS			ECSSD			OMRON		
	Acc	$\mathcal{J}\uparrow$	$\max F_{\beta}\uparrow$	Acc	$\mathcal{J}\uparrow$	$F_{\beta}\uparrow$	Acc	$\mathcal{J}\uparrow$	$F_{\beta}\uparrow$	Acc	$\mathcal{J}\uparrow$	$F_{\beta}\uparrow$
 Voynov <i>et al.</i>	94.0	71.0	80.7	88.1	51.1	60.0	90.6	68.4	79.0	86.0	46.4	53.3
 AMD	–	–	–	–	–	60.2	–	–	–	–	–	–
 Kyriazi <i>et al.</i>	92.1	66.4	78.3	89.3	52.8	61.4	91.5	71.3	80.6	88.3	50.9	58.3
 Kyriazi <i>et al.</i>	–	76.9	–	–	51.4	–	–	73.3	–	–	56.7	–
 DyStaB [†]	–	–	–	–	–	–	–	–	88.1	–	–	73.9
 TokenCut	–	–	–	90.3	57.6	–	91.8	71.2	–	88.0	53.3	–
 SelfMask	–	–	–	92.3	62.6	–	94.4	78.1	–	90.1	58.2	–
Ours	93.5	64.6	80.9	91.5	49.2	65.6	88.5	56.1	74.3	89.3	41.31	56.3

Table 3: **Unsupervised object segmentation** benchmark CUB and three saliency detection benchmarks: DUTS, ECSSD, and DUT-OMRON (*OMRON*). [†] DyStaB uses CRF post-processing, supervised pre-training, and self-training on each dataset. (SoTA only table - please see the supplement for a complete version of this table including many older methods.)

4.4 Unsupervised Image Segmentation

While the main aim of our work is object segmentation in videos, we also assess the image segmentation performance on common image segmentation and saliency benchmarks: CUB,

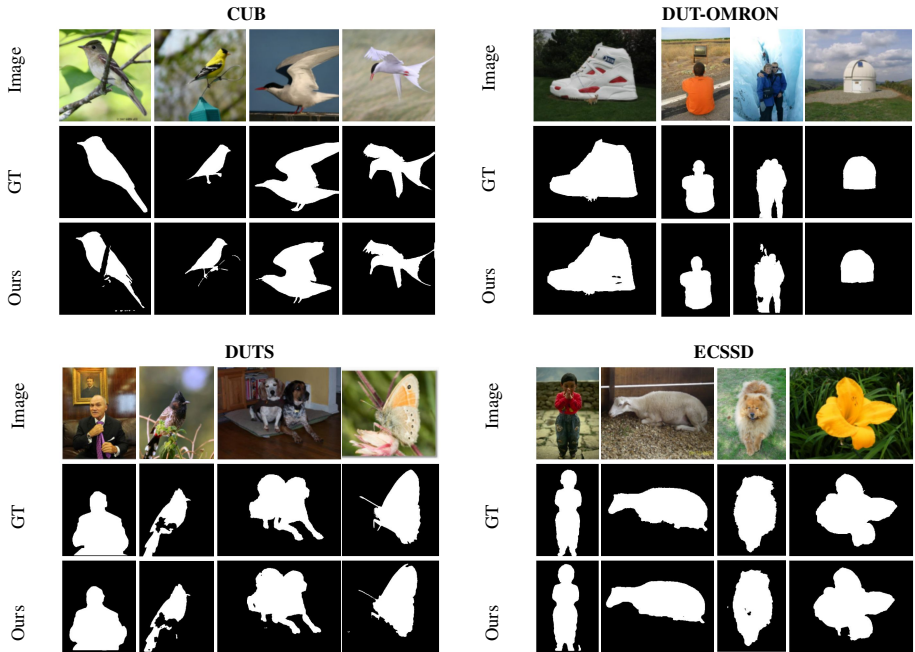


Figure 3: **Qualitative Comparison.** Our method can extract salient object in various environments and works even for novel object that were not included in the training data.

DUTS, DUT-OMRON, and ECSSD. For this experiment, we train our model on all three motion segmentation datasets (DAVIS, FBMS and STv2) jointly and apply the resulting network to the image segmentation benchmarks without any further fine-tuning. In Table 3, we report the performance of our method and compare to the current state of the art. It is worth noting that most prior work (except [55, 56, 89]) relies on dataset-specific training, self-training, post-processing or supervised pre-training to achieve image segmentation.

Finally, we evaluate the model qualitatively in Fig. 3 on all four benchmarks. We observe our model works well on a diverse set of classes, such as buildings, certain animals and plants, even though they were not part of the foreground (moving) objects in the training data.

5 Conclusions

We have proposed a simple approach to exploit the synergies between motion in videos and objectness for segmenting visual objects without supervision. The key idea is using motion anticipation as a learning signal: we train an image segmentation network to predict regions that likely contain simple optical flow patterns, as these have a high chance to correspond to objects. We find that the complexity of the motion model is important to model complicated flow patterns that can arise even for rigid objects. Our results show that this approach achieves state-of-the-art performance in video segmentation benchmarks. Future work could thus consider extensions to more sophisticated motion models, accounting for the 3D shape of objects, and to separate multiple objects.

Acknowledgments S. C. is supported by a scholarship sponsored by Facebook. L. K. is funded by EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines and Systems EP/S024050/1. I. L. and A. V. are supported by the European Research Council (ERC) grant IDIU-638009. I. L. is also supported by EPSRC grant VisualAI EP/T028572/1. C. R. is supported by Innovate UK (project 71653) on behalf of UK Research and Innovation (UKRI).

References

- [1] Gilad Adiv. Determining three-dimensional motion and structure from optical flow generated by several moving objects. *IEEE transactions on pattern analysis and machine intelligence*, (4):384–401, 1985.
- [2] Zhipeng Bao, Pavel Tokmakov, Allan Jabri, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert. Discovering objects that can move. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11789–11798, June 2022.
- [3] Daniel Bear, Chaofei Fan, Damian Mrowca, Yunzhu Li, Seth Alter, Aran Nayebi, Jeremy Schwartz, Li Fei-Fei, Jiajun Wu, Josh Tenenbaum, and Daniel L. Yamins. Learning physical graph representations from visual scenes. In *Advances in Neural Information Processing Systems*, volume 33, 2020.
- [4] 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.
- [5] Beril Besbinar and Pascal Frossard. Self-supervision by prediction for object discovery in videos. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 1509–1513. IEEE, 2021.
- [6] Pia Bideau and Erik Learned-Miller. It’s moving! a probabilistic model for causal motion segmentation in moving camera videos. In *European Conference on Computer Vision*, pages 433–449. Springer, 2016.
- [7] Pia Bideau, Aruni RoyChowdhury, Rakesh R Menon, and Erik Learned-Miller. The best of both worlds: Combining cnns and geometric constraints for hierarchical motion segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 508–517, 2018.
- [8] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. *Advances in Neural Information Processing Systems*, 32, 2019.
- [9] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *ArXiv*, abs/1809.11096, 2019.
- [10] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010.

- [11] Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE transactions on pattern analysis and machine intelligence*, 33(3):500–513, 2010.
- [12] Christopher P Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*, 2019.
- [13] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 221–230, 2017.
- [14] 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 (ICCV)*, pages 9650–9660, October 2021.
- [15] Jason Chang and John W. Fisher III. Topology-constrained layered tracking with latent flow. *2013 IEEE International Conference on Computer Vision*, pages 161–168, 2013.
- [16] Jeff Cheeger. A lower bound for the smallest eigenvalue of the laplacian. In *Proceedings of the Princeton conference in honor of Professor S. Bochner*, pages 195–199, 1969.
- [17] Mickaël Chen, Thierry Artières, and Ludovic Denoyer. Unsupervised object segmentation by redrawing. *Advances in neural information processing systems*, 32, 2019.
- [18] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *Advances in Neural Information Processing Systems*, 2021.
- [19] Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):569–582, 2015.
- [20] Eric Crawford and Joelle Pineau. Spatially invariant unsupervised object detection with convolutional neural networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence*, pages 3412–3420. AAAI Press, 2019.
- [21] Eric Crawford and Joelle Pineau. Exploiting spatial invariance for scalable unsupervised object tracking. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.
- [22] Patrick Emami, Pan He, Sanjay Ranka, and Anand Rangarajan. Efficient iterative amortized inference for learning symmetric and disentangled multi-object representations. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 2970–2981. PMLR, 18–24 Jul 2021.
- [23] Martin Engelcke, Adam R. Kosiorek, Oiwi Parker Jones, and Ingmar Posner. GENESIS: generative scene inference and sampling with object-centric latent representations. In *International Conference on Learning Representations*. OpenReview.net, 2020.

- [24] Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. Genesis-v2: Inferring unordered object representations without iterative refinement. *Advances in Neural Information Processing Systems*, 34, 2021.
- [25] Alon Faktor and Michal Irani. Video segmentation by non-local consensus voting. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014.
- [26] Rohit Girdhar and Deva Ramanan. Cater: A diagnostic dataset for compositional actions & temporal reasoning. In *International Conference on Learning Representations*, 2020.
- [27] Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *Proceedings of the International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2424–2433. PMLR, 2019.
- [28] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman. Unsupervised semantic segmentation by distilling feature correspondences. In *International Conference on Learning Representations*, 2022.
- [29] Xingzhe He, Bastian Wandt, and Helge Rhodin. Ganseg: Learning to segment by unsupervised hierarchical image generation. *arXiv preprint arXiv:2112.01036*, 2021.
- [30] Suyog Jain, Bo Xiong, and Kristen Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. *arXiv preprint arXiv:1701.05384*, 2017.
- [31] Allan D. Jepson and Michael J. Black. Mixture models for optical flow computation. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 760–761, 1993.
- [32] 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.
- [33] Jindong Jiang and Sungjin Ahn. Generative neurosymbolic machines. *Advances in Neural Information Processing Systems*, 33:12572–12582, 2020.
- [34] Jindong Jiang*, Sepehr Janghorbani*, Gerard De Melo, and Sungjin Ahn. Scalor: Generative world models with scalable object representations. In *International Conference on Learning Representations*, 2020.
- [35] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017.
- [36] N. Jojic and B.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, 2001.

- [37] Rishabh Kabra, Daniel Zoran, Goker Erdogan, Loic Matthey, Antonia Creswell, Matthew Botvinick, Alexander Lerchner, and Christopher P Burgess. SIMONE: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. *arXiv preprint arXiv:2106.03849*, 2021.
- [38] Asako Kanezaki. Unsupervised image segmentation by backpropagation. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 1543–1547. IEEE, 2018.
- [39] Laurynas Karazija, Iro Laina, and Christian Rupprecht. Clevrtex: A texture-rich benchmark for unsupervised multi-object segmentation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.
- [40] Margret Keuper, Bjoern Andres, and Thomas Brox. Motion trajectory segmentation via minimum cost multicuts. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 3271–3279, 2015.
- [41] Margret Keuper, Siyu Tang, Bjoern Andres, Thomas Brox, and Bernt Schiele. Motion segmentation and multiple object tracking by correlation co-clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1):140–153, 2020.
- [42] Thomas Kipf, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. *arXiv preprint arXiv:2111.12594*, 2021.
- [43] Adam Kosior, Hyunjik Kim, Yee Whye Teh, and Ingmar Posner. Sequential attend, infer, repeat: Generative modelling of moving objects. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.
- [44] Hala Lamdouar, Weidi Xie, and Andrew Zisserman. Segmenting invisible moving objects. In *BMVC*, 2021.
- [45] Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M. Rehg. Video segmentation by tracking many figure-ground segments. *2013 IEEE International Conference on Computer Vision*, pages 2192–2199, 2013.
- [46] 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.
- [47] Zhixuan Lin, Yi-Fu Wu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. SPACE: Unsupervised object-oriented scene representation via spatial attention and decomposition. In *International Conference on Learning Representations*. OpenReview.net, 2020.
- [48] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):978–994, 2010.

- [49] 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 (CVPR)*, June 2020.
- [50] 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, 2021.
- [51] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems*, volume 33, pages 11525–11538, 2020.
- [52] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.
- [53] 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.
- [54] 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*, September 2018.
- [55] 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 (CVPR)*, pages 8364–8375, June 2022.
- [56] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Finding an unsupervised image segmenter in each of your deep generative models. In *International Conference on Learning Representations*, 2022.
- [57] Etienne Meunier, Anaïs Badoual, and Patrick Bouthemy. Em-driven unsupervised learning for efficient motion segmentation. *CoRR*, abs/2201.02074, 2022.
- [58] Cheol-Hui Min, Jinseok Bae, Junho Lee, and Young Min Kim. Gatsbi: Generative agent-centric spatio-temporal object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3074–3083, 2021.
- [59] Tom Monnier, Elliot Vincent, Jean Ponce, and Mathieu Aubry. Unsupervised layered image decomposition into object prototypes. In *ICCV*, 2021.
- [60] 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. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

- [61] Peter Ochs and Thomas Brox. Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In *2011 international conference on computer vision*, pages 1583–1590. IEEE, 2011.
- [62] Peter Ochs and Thomas Brox. Higher order motion models and spectral clustering. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 614–621. IEEE, 2012.
- [63] Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36:1187–1200, 2014.
- [64] Yassine Ouali, Celine Hudelot, and Myriam Tami. Autoregressive unsupervised image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020.
- [65] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2013.
- [66] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Computer Vision and Pattern Recognition*, 2016.
- [67] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [68] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.
- [69] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.
- [70] 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, 2016.
- [71] 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 (CVPR) Workshops*, pages 3971–3980, June 2022.
- [72] Gautam Singh, Fei Deng, and Sungjin Ahn. Illiterate DALL-E learns to compose. In *International Conference on Learning Representations*, 2022.
- [73] Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. *arXiv preprint arXiv:2205.14065*, 2022.
- [74] Elizabeth S Spelke. Principles of object perception. *Cognitive science*, 14(1):29–56, 1990.

- [75] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [76] Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. Dense point trajectories by GPU-Accelerated large displacement optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010.
- [77] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision (ECCV)*, 2020.
- [78] Pavel Tokmakov, Cordelia Schmid, and Karteek Alahari. Learning to segment moving objects. *Int. J. Comput. Vision*, 127(3):282–301, mar 2019. ISSN 0920-5691.
- [79] Philip H. S. Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 356:1321 – 1340, 1998.
- [80] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Deep image prior. *IJCV*, 128(7):1867–1888, 2020.
- [81] 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.
- [82] Johan Wagemans, Jacob Feldman, Sergei Gepshtein, Ruth Kimchi, James R Pomerantz, Peter A Van der Helm, and Cees Van Leeuwen. A century of gestalt psychology in visual perception: Ii. conceptual and theoretical foundations. *Psychological bulletin*, 138(6):1218, 2012.
- [83] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [84] 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, 2018.
- [85] 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 (CVPR)*, pages 14543–14553, June 2022.
- [86] Yichen Wei, Fang Wen, Wangjiang Zhu, and Jian Sun. Geodesic saliency using background priors. In *ECCV*, 2012.
- [87] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.
- [88] Xide Xia and Brian Kulis. W-net: A deep model for fully unsupervised image segmentation. *arXiv preprint arXiv:1711.08506*, 2017.

- [89] Jun Xie, Weidi Xie, and Andrew Zisserman. Segmenting moving objects via an object-centric layered representation. *ArXiv*, abs/2207.02206, 2022.
- [90] 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.
- [91] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3166–3173. IEEE, 2013.
- [92] 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 (CVPR)*, June 2019.
- [93] 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.
- [94] 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 (CVPR)*, pages 2657–2666, June 2022.
- [95] Peiyu Yu, Sirui Xie, Xiaojian Ma, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Unsupervised foreground extraction via deep region competition. *Advances in Neural Information Processing Systems*, 34, 2021.
- [96] Polina Zablotskaia, Edoardo A Dominici, Leonid Sigal, and Andreas M Lehrmann. Unsupervised video decomposition using spatio-temporal iterative inference. *arXiv preprint arXiv:2006.14727*, 2020.
- [97] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, Lihe Zhang, Mingyang Qian, and Yizhou Yu. Multi-source weak supervision for saliency detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
- [98] Jing Zhang, T. Zhang, Yuchao Dai, Mehrtash Harandi, and Richard I. Hartley. Deep unsupervised saliency detection: A multiple noisy labeling perspective. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9029–9038, 2018.

Supplementary Material

In this supplementary material, we provide further details on our training parameters in Appendix A. Appendix B contains the closed form solution of the fitting of the flow model θ . Expanded experiments and ablations are found in Appendix C. Finally, more qualitative results are presented in Appendix D. See the project page, <https://www.robots.ox.ac.uk/~vgg/research/gwm>, for additional visualizations, code and models.

A Experimental Setup

Network. We use MaskFormer [L3] as our segmentation network¹, and use only the segmentation head. As MaskFormer predicts masks at 4 times lower resolution than input, we modify the PixelDecoder by appending $[Conv(3), UpsampleNN(2), Conv(1)] \times 2$ to its output layers to bring the masks back up to the input resolution.

For the backbone and appearance features V , we leverage a ViT-8 transformer, pre-trained on ImageNet [L5] in a self-supervised manner using DINO [L4] to avoid any external sources of supervision. For the hierarchical backbone features to decoder we use the key feature outputs from layers 6, 8, 10, 12.

The input RGB images are interpolated (bi-cubic) to 128×224 resolution for input to the network. We interpolate (nearest neighbor) the optical flow to 480×854 for the loss. Output segmentation logits are up-sampled using bi-linear interpolation to the flow resolution for training and again to annotation resolution for evaluation.

Training Hyperparameters. The networks are optimised using AdamW [L2], with learning rate of 1.5×10^{-4} , a schedule of linear warm-up from 1.0×10^{-6} to 1.5×10^{-4} over 1.5k iteration and polynomial decay afterwards. We use batch size of 8 and train for 15k iterations. We additionally employ gradient clipping when the 2-norm exceeds 0.01 for stability. The loss multiplier is 0.03.

UNet. For experiments using U-Net², we use the standard 4-layer version. The batch-size is increased to 16 and learning rate to 7.0×10^{-4} . We also clip the gradients only when 2-norm exceeds 5.0. All other settings, including optimizer and learning rate schedules, are kept the same. U-Net is not pre-trained and trained from scratch.

Optical Flow. Our method derives its learning signal from optical flow estimated using off-the-shelf frozen networks. We estimate optical flow for all frames on DAVIS, STv2, and FBMS following the practice of MotionGrouping [L1]. We employ RAFT [L4] (supervised) using the original resolution for our main experiments, and gaps between frames of $\{-2, -1, 1, 2\}$ for DAVIS and STv2, and $\{-6, -3, 3, 6\}$ on FBMS. When multiple flows are associated with a single frame (multiple gaps), we sample one at random for each iteration.

B Quadratic Flow Model: Closed Form Solution

Consider one of K regions m and define $w_u \propto P(m_u = k | I, \Phi)$ the posterior probability for that region, normalized so that $\sum_{u \in \Omega} w_u = 1$ (the scaling factor does not matter for the purpose of

¹Implementation from <https://github.com/facebookresearch/MaskFormer>.

²Implementation from <https://github.com/milesial/Pytorch-UNet>.

finding the minimizer). We can obtain the minimizer (A^*, b^*) and minimum of the energy

$$E(A, b) = \sum_{u \in \Omega} w_u \|F_u - Au - b\|^2 \quad (7)$$

as follows. Defining

$$\bar{u} := \begin{bmatrix} u \\ 1 \end{bmatrix}, \quad M := \begin{bmatrix} A & b \end{bmatrix} \in \mathbb{R}^{2 \times 6}$$

allows rewriting the energy as

$$E(M) = \sum_{u \in \Omega} w_u \|F_u - M\bar{u}\|^2 = \text{tr} \left(\Lambda_{FF} - M\Lambda_{\bar{F}\bar{\Omega}} - \Lambda_{F\bar{\Omega}}M^\top + M\Lambda_{\bar{\Omega}\bar{\Omega}}M^\top \right),$$

where

$$\Lambda_{FF} = \sum_{u \in \Omega} w_u F_u F_u^\top, \quad \Lambda_{F\bar{\Omega}} = \sum_{u \in \Omega} w_u F_u \bar{u}^\top, \quad \Lambda_{\bar{\Omega}F} = \Lambda_{F\bar{\Omega}}^\top, \quad \Lambda_{\bar{\Omega}\bar{\Omega}} = \sum_{u \in \Omega} w_u \bar{u} \bar{u}^\top.$$

are the (uncentered) second moment matrices of the flow F_u and homogeneous coordinate vectors \bar{u} . By inspection of the trace term, the gradient of the energy is given by:

$$\frac{dE(M)}{dM} = 2(\Lambda_{F\bar{\Omega}} - M\Lambda_{\bar{\Omega}\bar{\Omega}})$$

Hence, the optimal regression matrix M^* and corresponding energy value are

$$M^* = \Lambda_{F\bar{\Omega}} \Lambda_{\bar{\Omega}\bar{\Omega}}^{-1}, \quad E(M^*) = \text{tr}(\Lambda_{FF} - M^* \Lambda_{\bar{\Omega}\bar{\Omega}}).$$

Somewhat more intuitive results can be obtained by centering the moments and resolving for A and b instead of M . Specifically, define:

$$\mu_\Omega := \sum_{u \in \Omega} w_u u, \quad \mu_F := \sum_{u \in \Omega} w_u F_u.$$

The covariance matrices of the vectors are:

$$\begin{aligned} \Sigma_{FF} &= \sum_{u \in \Omega} w_u (F_u - \mu_F)(F_u - \mu_F)^\top, & \Sigma_{F\Omega} &= \sum_{u \in \Omega} w_u (F_u - \mu_F)(u - \mu_\Omega)^\top, \\ \Sigma_{\Omega F} &= \Lambda_{F\Omega}^\top, & \Sigma_{\Omega\Omega} &= \sum_{u \in \Omega} w_u (u - \mu_\Omega)(u - \mu_\Omega)^\top. \end{aligned}$$

It is easy to check that

$$\Lambda_{FF} = \Sigma_{FF} + \mu_F \mu_F^\top, \quad \Lambda_{F\bar{\Omega}} = \begin{bmatrix} \Sigma_{F\Omega} + \mu_F \mu_\Omega^\top & \mu_F \end{bmatrix}, \quad \Lambda_{\bar{\Omega}\bar{\Omega}} = \begin{bmatrix} \Sigma_{\Omega\Omega} + \mu_\Omega \mu_\Omega^\top & \mu_\Omega \\ \mu_\Omega^\top & 1 \end{bmatrix}.$$

From this:

$$\begin{aligned} M^* &= \Lambda_{F\bar{\Omega}} \Lambda_{\bar{\Omega}\bar{\Omega}}^{-1} = \begin{bmatrix} \Sigma_{F\Omega} + \mu_F \mu_\Omega^\top & \mu_F \end{bmatrix} \begin{bmatrix} \Sigma_{\Omega\Omega} + \mu_\Omega \mu_\Omega^\top & \mu_\Omega \\ \mu_\Omega^\top & 1 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \Sigma_{F\Omega} + \mu_F \mu_\Omega^\top & \mu_F \end{bmatrix} \begin{bmatrix} \Sigma_{\Omega\Omega}^{-1} & -\Sigma_{\Omega\Omega}^{-1} \mu_\Omega \\ -\mu_\Omega^\top \Sigma_{\Omega\Omega}^{-1} & 1 + \mu_\Omega^\top \Sigma_{\Omega\Omega}^{-1} \mu_\Omega \end{bmatrix} \\ &= \begin{bmatrix} \Sigma_{F\Omega} \Sigma_{\Omega\Omega}^{-1} & \mu_F - \Sigma_{F\Omega} \Sigma_{\Omega\Omega}^{-1} \mu_\Omega \end{bmatrix} = \begin{bmatrix} A^* & b^* \end{bmatrix}. \end{aligned}$$

Hence, the optimal regression coefficients and energy value are also given by:

$$A^* = \Sigma_{F\Omega} \Sigma_{\Omega\Omega}^{-1}, \quad b^* = \mu_F - A^* \mu_\Omega.$$

	Model	Flow	DAVIS ($\mathcal{J}\uparrow$)	FBMS ($\mathcal{J}\uparrow$)
[62]	AMD (100 steps)	\times	57.8	47.5
	Ours (Zero shot)	ARFlow	62.5	65.4
	Ours (20 steps)	ARFlow	65.2	67.6
[63]	EM	RAFT	69.3	57.8
	Ours (Zero shot)	RAFT	66.8	73.2
	Ours (20 steps)	RAFT	76.3	77.1

Table 4: **Generalization performance on unseen videos.** Few unsupervised methods operate in this setting. AMD trains on YT-VOS, followed by 100 test-time adaptation steps, while EM trains on FlyingThings3D using flow as input. We use (fully unsupervised) ARFlow for fair comparison with AMD. Our method shows better performance after observing motion. (Test-time adaptation uses the training loss. No GT is involved at any point.)

Backbone model	Backbone pretraining	Sup.	DAVIS $\mathcal{J}\uparrow$	STv2 $\mathcal{J}\uparrow$	FBMS $\mathcal{J}\uparrow$
ViT-8	ImageNet DINO	\times	79.5	78.3	77.4
UNet	None	\times	78.3	76.8	72.0
SWIN-tiny	ImageNet MOBY	\times	78.3	77.4	74.6
SWIN-tiny	ImageNet CLS	\checkmark	78.9	77.7	75.5
SWIN-tiny	None	\times	78.3	75.2	68.8
Resnet-50	ImageNet CLS	\checkmark	77.5	75.8	72.9

Table 5: **Effect of Pretraining/Backbone.** Our method with MaskFormer benefits from pre-training, with slight improvement offered by supervised (*CLS*) over unsupervised (*MOBY*) pretraining (using SWIN transformer). Comparable results can be obtained with training from scratch. Best results are obtained using DINO features.

C Further Experiments

C.1 Generalization in Unsupervised Video Segmentation

We also test our model in a video *generalization* setting. In contrast to the protocol of [90, 92], where evaluation set is observed together with training to infer masks jointly³, here we train only on frames from the training set. We report performance on unseen videos. In this case, our method independently segments a collection of frames from a new video, with no way to incorporate motion information.

To “observe” motion on *unseen* inputs, we also report results after taking 20 test-time adaptation steps (using our unsupervised loss) for each evaluation sequence in isolation (c.f. AMD [62] takes 100 test-time adaptations steps). That is after training, we follow our training setup (optimizer, rate, batch size) and feed frames from the evaluation video and corresponding optical flow, calculate loss and take gradient steps. Despite other methods using much larger training sets, our approach shows better performance (Table 4).

C.2 Ablation Studies

³Note, no annotations are observed at any point.

Model	K	Merge	DAVIS $\mathcal{J}\uparrow$	STv2 $\mathcal{J}\uparrow$	FBMS $\mathcal{J}\uparrow$
Ours	$K = 4$	✓	79.5	78.3	77.4
Spectral clustering	$K = 2$	✗	15.79	14.89	27.45
K-Means	$K = 4$	✓	41.79	34.84	48.80
K-Means	$K = 2$	✗	20.24	21.14	38.25

Table 6: **Feature Clustering without Motion.** We experiment with offline clustering of DINO features to assess the importance of our motion-based formulation. Simply clustering DINO features using K-Means or spectral clustering [53] into 2 clusters performs worse. Over-clustering and merging using our cluster-merging approach performs better but still fails to reach our performance.

Opt. Flow	Sup.	DAVIS ($\mathcal{J}\uparrow$)
[44] ARFlow	✗	66.9
[45] PWCNet	✓	74.9
[46] RAFT	✓	79.5

Table 7: **Choice of Optical Flow Method.** Measuring the influence of the method to extract optical flow.

Method	DAVIS ($\mathcal{J}\uparrow$)
[47] MG	53.2
[48] AMD	57.8
Ours	66.9

Table 8: **Fully Unsupervised Video Object Segmentation.** Comparison to the state of the art in unsupervised VOS without reliance on *any* supervision

Pretraining. Compared to recent methods for video segmentation [47, 48], one of the benefits of our formulation is that we can leverage unsupervised pretraining for the segmentation network (*e.g.*, for the ViT backbone of MarkFormer). This enables our method to be trained in only 15k iterations. Here, we investigate the importance of the backbone. To this end we replace ViT with Swin-tiny pretrained using MOBY (self-supervised) in Table 5. The performance differences are small.

Additionally, we investigate the effect of other pretraining strategies on the performance. Switching to a model pretrained on ImageNet with image-level supervision (*i.e.* a classification task) only slightly improves performance showing that the method does not need to rely on supervised pre-training. Finally, we train the model using same settings for 20k iterations from scratch, without any pre-training. This results in comparable performance on DAVIS but reduced performance on the smaller datasets. Comparing backbones without pre-training, UNet gives better results than SWIN-tiny, likely due to smaller networks being easier to train on small datasets.

Feature Clustering without Motion. To demonstrate the potential of using motion for discovering objects, in Table 6, we compare to additional baselines that only rely on clustering visual features. Spectral feature clustering with $K = 2$ (based on [53]), on the same visual features we use to merge segments (*i.e.*, DINO) after over-clustering, shows (somewhat unsurprisingly) that learning from motion is important for motion segmentation. Similarly, K-means ($K = 2$) on the same features also falls behind our method. Yet, we show that K-means also benefits from over-clustering ($K = 4$) and then merging.

	CUB			DUTS			ECSSD			OMRON		
	Acc	$\mathcal{J} \uparrow$	$maxF_{\beta} \uparrow$	Acc	$\mathcal{J} \uparrow$	$F_{\beta} \uparrow$	Acc	$\mathcal{J} \uparrow$	$F_{\beta} \uparrow$	Acc	$\mathcal{J} \uparrow$	$F_{\beta} \uparrow$
[83] WNet [†]	-	24.8	-	-	-	-	-	-	-	-	-	-
[62] IIC-seg	-	36.5	-	-	-	-	-	-	-	-	-	-
[8] PertGAN	-	38.0	-	-	-	-	-	-	-	-	-	-
[62] ReDO	84.5	42.6	-	-	-	-	-	-	-	-	-	-
[62] UISB	-	44.2	-	-	-	-	-	-	-	-	-	-
[8] OneGAN	-	55.5	-	-	-	-	-	-	-	-	-	-
[62] DRC	-	56.4	-	-	-	-	-	-	-	-	-	-
[62] GANSeg	-	62.9	-	-	-	-	-	-	-	-	-	-
[62] Voynov <i>et al.</i>	94.0	71.0	80.7	88.1	51.1	60.0	90.6	68.4	79.0	86.0	46.4	53.3
[62] AMD	-	-	-	-	-	60.2	-	-	-	-	-	-
[62] Kyriazi <i>et al.</i>	92.1	66.4	78.3	89.3	52.8	61.4	91.5	71.3	80.6	88.3	50.9	58.3
[62] Kyriazi <i>et al.</i>	-	76.9	-	-	51.4	-	-	73.3	-	56.7	-	-
[62] DyStaB [†]	-	-	-	-	-	-	-	-	88.1	-	-	73.9
[85] TokenCut	-	-	-	90.3	57.6	-	91.8	71.2	-	88.0	53.3	-
[62] SelfMask	-	-	-	92.3	62.6	-	94.4	78.1	-	90.1	58.2	-
Ours	93.5	64.6	80.9	91.5	49.2	65.6	88.5	56.1	74.3	89.3	41.31	56.3

Table 9: **Expanded unsupervised object segmentation** benchmark CUB and three saliency detection benchmarks: DUTS, ECSSD, and DUT-OMRON (*OMRON*). [†] DyStaB uses CRF post-processing, supervised pre-training, and self-training on each dataset.

Flow Estimation. Finally, our method relies on optical flow estimated by frozen, off-the-shelf networks. So far we have been using RAFT [62], as such optical flow network was adopted in our baselines. In Table 7, we also consider PWCNet [45] and fully-unsupervised ARFlow [49]. We observe that the performance of the flow estimator has an impact on the final performance of our method. Finally, we compare our *fully* unsupervised model (which uses self-supervised pretraining and flow) to fully unsupervised state-of-the-art methods. Appearance-Motion Decomposition (AMD) [60] works end-to-end and directly extracts motion features from pairs of images with a PWCNet-like architecture, while MotionGrouping (MG) [90] and our method use ARFlow [62] for optical flow estimation. In Table 8 we show that our method achieves a significant improvement over previous approaches.

D Additional Results and Discussion

We provide a further breakdown of our results in Tables 10 to 12, reporting per sequence evaluation results on the video segmentation tasks.

Video object segmentation and egomotion. We note that some sequences have pronounced egomotion (*e.g.*, camera shaking in *libby* of DAVIS or inside a moving car in *camel01* of FBMS). Our model performs well on these sequences, demonstrating that it can handle egomotion. When *only* the camera is moving, the resulting optical flow would still highlight objects due to parallax. This provides a learning signal, however, it would likely be weaker for objects farther away from the camera. As our method works on a per-frame basis and does not *require* flow during inference, this should not have an impact at test time. However, fine-tuning on scenes with only egomotion (see Appendix C.1 for experiments investigating test-time adaptation) and only small or far away objects, might lead to the model learning to ignore them.

Sequence	<i>w/o CRF</i>			<i>w/ CRF</i>		
	$\mathcal{J}(M)$	$\mathcal{J}(R)$	$\mathcal{J}(D)$	$\mathcal{J}(M)$	$\mathcal{J}(R)$	$\mathcal{J}(D)$
blackswan	67.0	100.0	-0.8	67.4	100.0	1.1
bmx-trees	58.2	76.9	19.9	59.8	76.9	17.5
breakdance	86.2	100.0	4.9	87.4	100.0	5.2
camel	89.4	100.0	5.7	90.6	100.0	5.5
car-roundabout	81.4	90.4	26.7	81.2	90.4	25.8
car-shadow	84.3	100.0	9.0	83.9	100.0	8.0
cows	90.4	100.0	3.4	91.3	100.0	3.2
dance-twirl	87.4	100.0	-7.1	88.8	100.0	-6.2
dog	92.9	100.0	-1.7	93.9	100.0	-1.6
drift-chicane	78.6	98.0	2.2	82.0	100.0	2.6
drift-straight	80.6	100.0	7.2	82.1	100.0	8.2
goat	78.6	100.0	1.7	75.8	100.0	4.5
horsejump-high	84.9	100.0	6.4	88.0	100.0	4.6
kite-surf	64.4	97.9	4.5	67.5	97.9	3.1
libby	82.9	100.0	8.6	84.5	100.0	8.6
motocross-jump	74.1	78.9	4.1	75.1	81.6	4.1
paragliding-launch	62.2	65.4	33.5	64.1	66.7	35.8
parkour	86.1	100.0	-4.5	88.1	100.0	-3.1
scooter-black	82.1	97.6	-4.3	82.1	100.0	-4.3
soapbox	79.2	100.0	-2.8	81.0	100.0	-0.4
Average	79.5	95.3	5.8	80.7	95.7	6.1

Table 10: **Result breakdown on DAVIS16 validation sequences.** (M), (R), and (D) are mean, recall and decay of IoU, respectively

Image segmentation. For unsupervised image segmentation, we show some additional qualitative results for CUB in Fig. 4, DUT-OMRON in Fig. 5, DUTS in Fig. 6, and ECSSD in Fig. 7. Our model, trained on a combined dataset of DAVIS, FBMS and STv2, is robust enough to handle a wide array of classes from the above datasets in varying context. Our model can segment both stationary and non-stationary objects and works well when multiple objects are in the foreground. In Fig. 8, we show a few failure cases for all datasets, where the model struggles mostly with ambiguous foreground objects and, in particular, with close-ups of stationary objects, *e.g.* signs (ECSSD) and buildings (DUT-OMRON). The model also has issues with boundaries for many objects, *i.e.* the foreground objects are correctly identified but the model fails to fully segment them. For example, in DUTS, the snake in the first image has a well segmented head, however, the model does not segment its body accurately.

Sequence	<i>w/o CRF</i> $\mathcal{J}(M)$	<i>w/ CRF</i> $\mathcal{J}(M)$
drift	86.1	86.5
birdfall	67.8	57.1
girl	84.5	86.3
cheetah	57.0	50.8
worm	83.7	84.0
parachute	90.6	93.2
monkeydog	22.9	22.6
hummingbird	57.3	57.2
soldier	77.4	77.4
bmw	76.4	77.5
frog	84.1	86.7
penguin	77.7	76.8
monkey	75.0	75.8
bird of paradise	92.3	94.0
Seq. Avg.	73.8	73.3
Frame Avg.	78.3	78.9

Table 11: Sequence breakdown on Seg-Trackv2 dataset.

Sequence	<i>w/o CRF</i> $\mathcal{J}(M)$	<i>w/ CRF</i> $\mathcal{J}(M)$
camel01	86.8	91.0
cars1	86.9	86.8
cars10	64.6	64.8
cars4	81.5	82.4
cars5	81.6	82.1
cats01	87.7	89.5
cats03	69.4	63.2
cats06	66.5	67.4
dogs01	76.3	75.6
dogs02	85.3	86.4
farm01	90.8	90.5
giraffes01	82.1	83.9
goats01	79.9	83.7
horses02	80.4	83.6
horses04	59.8	60.5
horses05	72.8	74.5
lion01	75.1	75.0
marple12	81.9	81.6
marple2	84.4	85.9
marple4	81.1	82.4
marple6	95.1	95.1
marple7	76.6	77.6
marple9	95.4	96.3
people03	90.1	91.0
people1	85.3	87.2
people2	88.1	89.7
rabbits02	91.2	91.2
rabbits03	81.5	84.4
rabbits04	43.8	44.1
tennis	73.3	74.2
Seq. Avg.	79.8	80.7
Frame Avg.	77.4	78.4

Table 12: Sequence breakdown on FBMS59 dataset

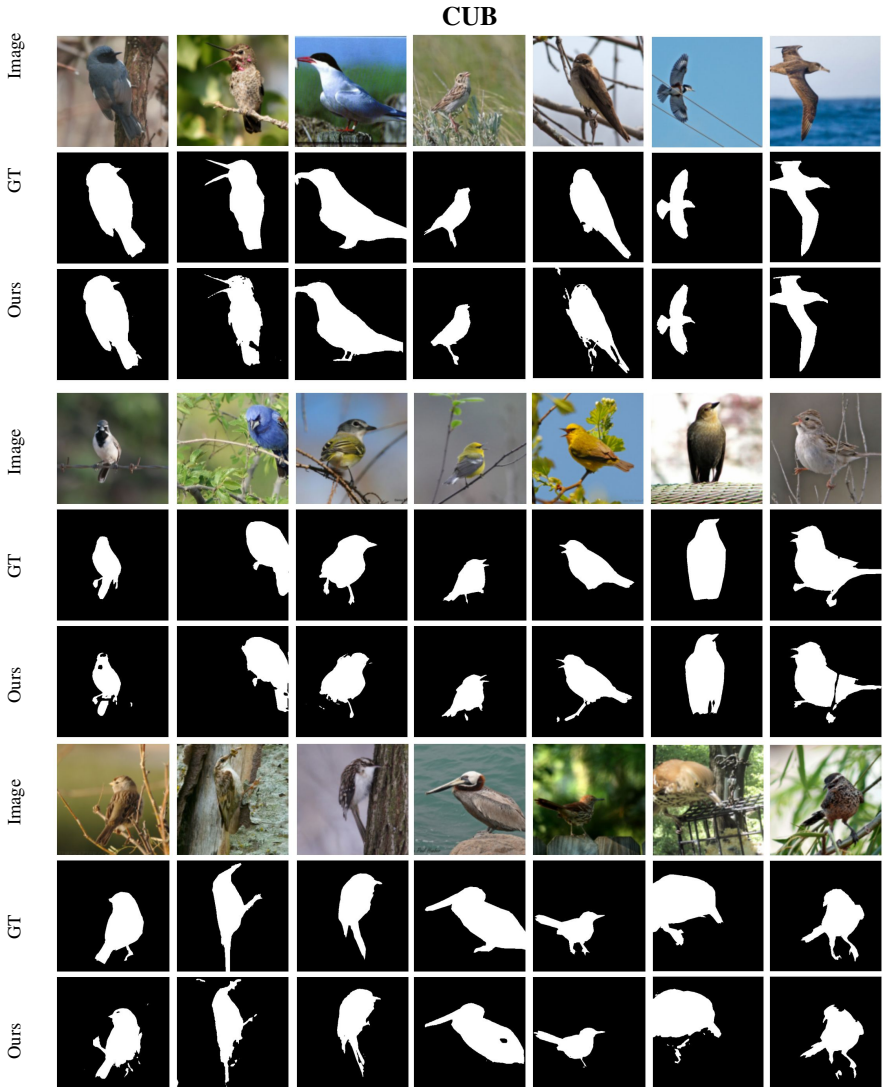


Figure 4: **Qualitative Comparison on CUB.** We train our model on a combined dataset of DAVIS, FBMS and STv2. Our method can extract birds in different environments and poses. Our model can segment different species of birds



Figure 5: **Qualitative Comparison on DUT-OMRON.** We train our model on a combined dataset of DAVIS, FBMS and STv2. Our model can segment both stationary and non-stationary objects and is robust enough to work on a wide range of classes



Figure 6: **Qualitative Comparison on DUTS.** We train our model on a combined dataset of DAVIS, FBMS and STv2. We can segment a wide array of classes. Our model performs well on scenes where multiple objects are in the foreground

ECSSD

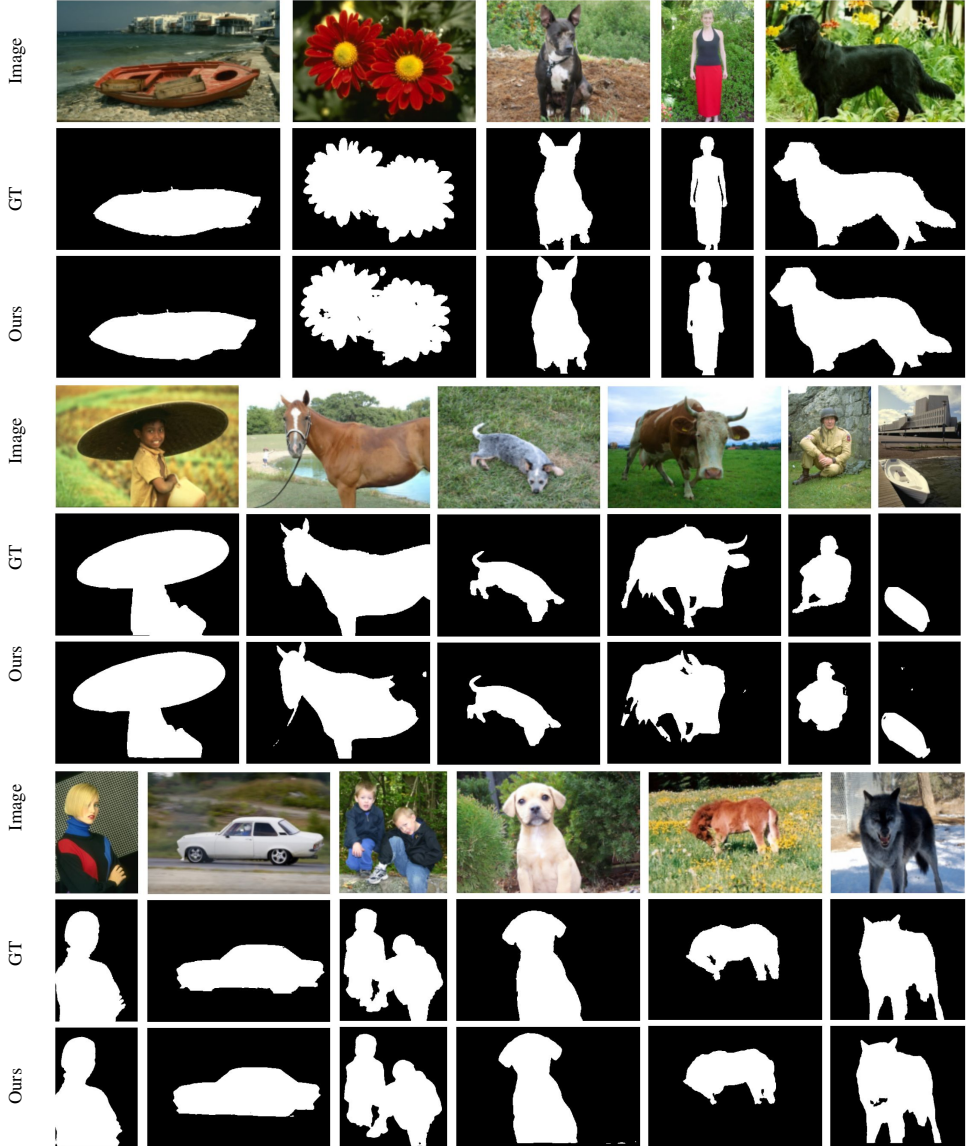


Figure 7: **Qualitative Comparison on ECSSD.** We train our model on a combined dataset of DAVIS, FBMS and STv2. Our model can segment objects from different classes in complex poses

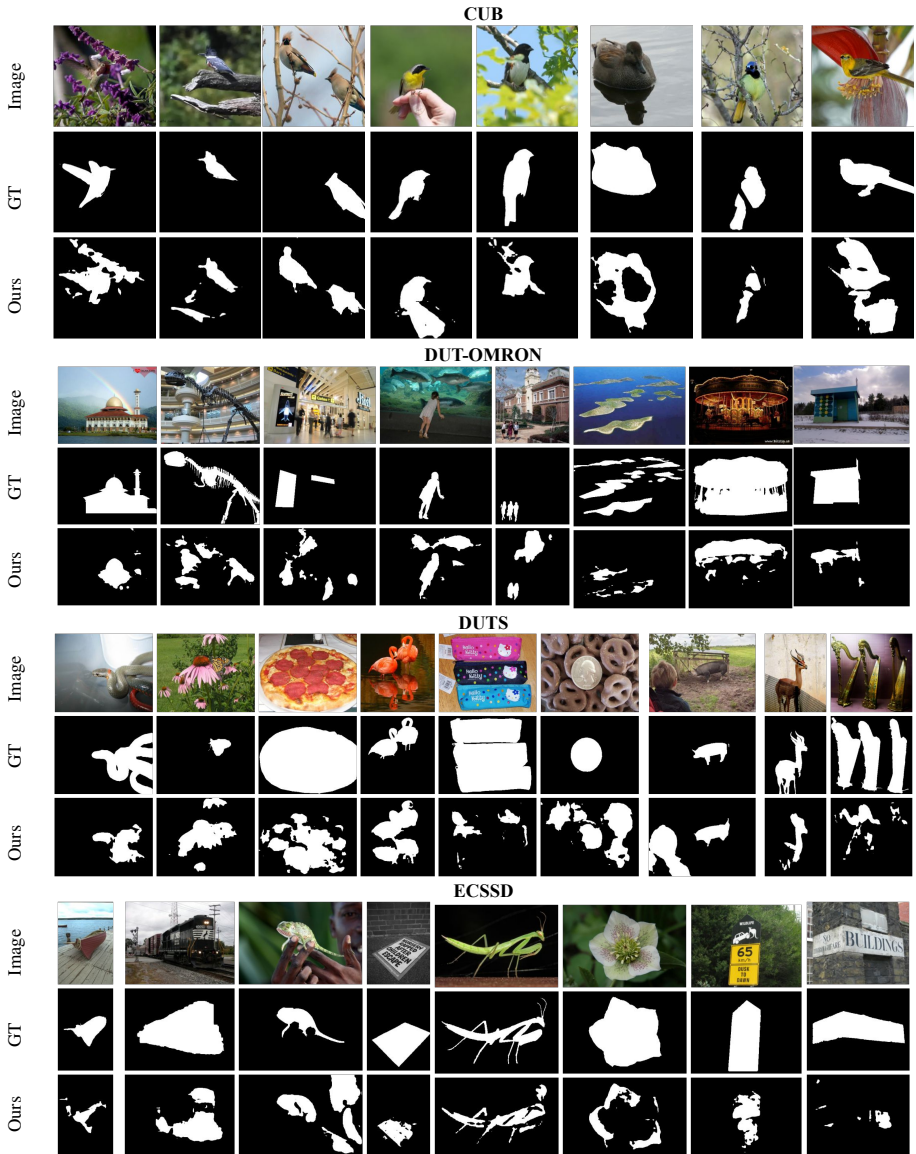


Figure 8: **Qualitative Comparison of Failure Cases.** We train our model on a combined dataset of DAVIS, FBMS and STv2. Our method can extract salient object in various environments. The model has difficulty where the foreground object is ambiguous — when there are multiple prominent objects but only few are annotated as salient object. The model also has issues with predicting the object boundaries well for some instances