Segmenting Invisible Moving Objects

Hala Lamdouar
lamdouar@robots.ox.ac.uk
Weidi Xie
weidi@robots.ox.ac.uk
Andrew Zisserman
az@robots.ox.ac.uk

Visual Geometry Group
Department of Engineering Science
University of Oxford, UK

Abstract

Biological visual systems are exceptionally good at perceiving objects that undergo changes in appearance, pose, and position. In this paper, we aim to train a computational model with similar functionality to segment the *moving* objects in videos. We target the challenging cases when objects are "invisible" in the RGB video sequence – for example, breaking camouflage, where visual appearance from a static scene can barely provide informative cues, or locating the objects as a whole even under partial occlusion.

To this end, we make the following contributions: (i) In order to train a motion segmentation model, we propose a scalable pipeline for generating synthetic training data, significantly reducing the requirements for labour-intensive annotations; (ii) We introduce a dual-head architecture (hybrid of ConvNets and Transformer) that takes a sequence of *optical flows* as input, and learns to segment the moving objects even when they are partially occluded or stop moving at certain points in videos; (iii) We conduct thorough ablation studies to analyse the critical components in data simulation, and validate the necessity of Transformer layers for aggregating temporal information and for developing object permanence. When evaluating on the MoCA camouflage dataset, the model trained only on synthetic data demonstrates state-of-the-art segmentation performance, even outperforming strong supervised approaches. In addition, we also evaluate on the popular benchmarks DAVIS2016 and SegTrackv2, and show competitive performance despite only processing optical flow. The project webpage is at: www.robots.ox.ac.uk/~vgg/research/simo/

1 Introduction

Object segmentation is undoubtedly one of the most widely researched problems in computer vision. Early attempts took inspiration from psychology, trying to understand the fundamental rules that cause people to see things as "unified" groups [47], and to translate them into the computational world, for instance, through clustering algorithms [37] that grouped the pixels with certain characteristics, *e.g.* semantics, color, intensity, texture, and motion.

In the recent literature, segmentation has been primarily treated as a problem of pixel-wise classification, and tackled by training deep networks on large-scale image or video datasets. This has two drawbacks: first, the requirements for heavy annotations limits the

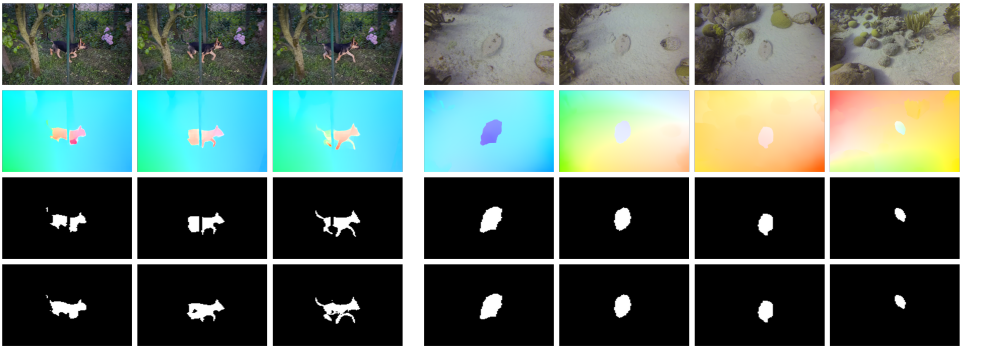


Figure 1: **Video sequences from DAVIS2016 (left) and MoCA (right).** Top to bottom: RGB sequence, optical flows, modal (only visible parts) and amodal (object as a whole) masks from our model prediction. Note that, our model exploits the optical flow as input, and is able to infer the whole mask of the objects even when they are partially occluded (the dog is partially occluded by the pole), or hidden in the environment (the fish is blended into the background environment).

scalability; second, the assumption that objects can be well-identified via their visual appearances is often an over-simplification of our visual world, since objects may *not* be "visible" in a static scene. For instance, to segment objects that are camouflaged, or only partially visible due to occlusions (as shown in Figure 1), the visual appearance alone usually provides inadequate visual cues, thus challenging the existing segmentation models. Instead, motion cues are required for segmentation, and that is the objective of this paper.

To address the challenge of limited availability of groundtruth annotations, we propose a scalable pipeline for generating synthetic data, and advocate a *Sim2Real* training procedure – once trained on synthetic data, the model can directly generalise to the downstream task, *e.g.* segmenting moving objects in real videos *without* finetuning on manual annotations. Generally speaking, although the visual scenes in real-world videos might be arbitrarily complex, the motion between two frames can always be factored into a combination of camera and object motion [14], effectively discarding the nuisance factors in visual appearance, *e.g.* color, texture, illumination. We exploit such observation to minimise the gap between the simulated and real videos. Specifically, for camera motion, it can often be approximated by a global transformation, while for an individual object, if it does move, pixels on it tend to form groups moving in similar direction and rate (and similarly in the optical flow). To this end, we simulate videos by composing moving objects onto a canvas that undergoes an independent transformation.

In practise, one challenge of only using optical flow comes from the fact that, objects in videos may stop moving or be partially occluded at any time point, leaving no effective cues for segmentation. As a consequence, one key ability of a functional vision system is to develop a sense of object permanence, *i.e.* realising that objects continue to exist even when they cannot be seen explicitly. To fulfil this goal, we propose a dual-head architecture, a hybrid of ConvNets and a Transformer, digesting a sequence of optical flows, and aggregating temporal information to flexibly output segmentation for only moving parts (modal), or the whole object even it is temporally static or under partial occlusion (amodal).

To summarise, we make the following contributions: (i) To address the challenge from limited availability of training data, we propose a scalable pipeline for generating synthetic video sequences with objects moving independently to the background motion; (ii) We intro-

duce a dual-head architecture, a hybrid of ConvNet and Transformer, which takes a sequence of optical flow images as input, jointly processing them to segment the moving objects at any time point, and to predict both modal and amodal segmentations; (iii) After training on synthetic video sequences, we show *state-of-the-art* performance on the MoCA camouflage dataset, *without* even finetuning on manual annotations. In addition, we evaluate on two other popular benchmarks for video object segmentation, DAVIS2016 and SegTrackv2, and also show competitive performance.

2 Related Work

Video object segmentation considers the task of grouping pixels that belong to the same object. In the recent literature, semi-supervised video object segmentation (**semi-supervised VOS**), and unsupervised video object segmentation (**unsupervised VOS**) have attracted increasing attentions from the community [3, 5, 7, 9, 12, 13, 15, 17, 18, 19, 20, 21, 28, 30, 31, 32, 34, 38, 42, 43, 44, 45, 46, 49, 49, 53]. Specifically, semi-supervised VOS aims to continuously segment the objects that are specified by the user in the first frame; unsupervised VOS considers to localise and segment the *prominent* object in a video sequence. In practise, all previous approaches heavily rely on manual annotations, as an alternative, in this work, we consider the problem of video object segmentation from the perspective of *Sim2Real*, by that we mean to train the model on synthetic data, and demonstrate direct generalisation to real video sequences.

Motion segmentation shares some similarity with unsupervised VOS, but focuses only on the *moving* objects. In the literature, motion segmentation generally aims to segment the objects that move independently to the camera. Specifically, [6, 30, 48] tackle the problem through the lens of clustering, with the goal to assign the same labels to pixels undergoing similar motion patterns; while [9, 41, 42] train deep networks that map the motion fields to segmentation masks. In [2, 4, 22, 51], the authors propose to highlight the independently moving object by compensating the background motion, either by registering consecutive frames, or explicitly estimating camera motions. [52] proposes an adversarial setting, where a generator is trained to produce masks, jittering the input flow, such that the inpainter fails to estimate the missing information. In constrained scenarios, such as the autonomous driving domain, [36] proposes to jointly optimise depth, camera motion, optical flow and motion segmentation. Concurrent work [50] adopts a layered representation to train a generative model, with motion segmentation being a by-product. In contrast to the existing approaches, we propose to train the architecture purely with synthetic video sequences, taking no prior knowledge of the objects' category or shape, and advocate direct generalisation to real videos.

Camouflage breaking aims to discover the objects that are hiding in the scene, where the visual appearance barely provide informative cues [8, 22, 23]. As such, the objects will only be apparent when they start to move, thus is closely related to motion segmentation.

Optical flow usually treated as the default motion representation in computer vision applications, where synthetic data has been predominantly adopted for training models, *e.g.* MPI Sintel [6], FlyingThings3D [29] and Monkaa [29], pose estimation [10], AutoFlow [39].

Amodal segmentation refers to the task of segmenting the object as whole, including the portions that are partially occluded [25, 35, 55]. Part of this work follows this line of research and aims to extend amodal perception to video object segmentation.

3 Synthetic Data Generation

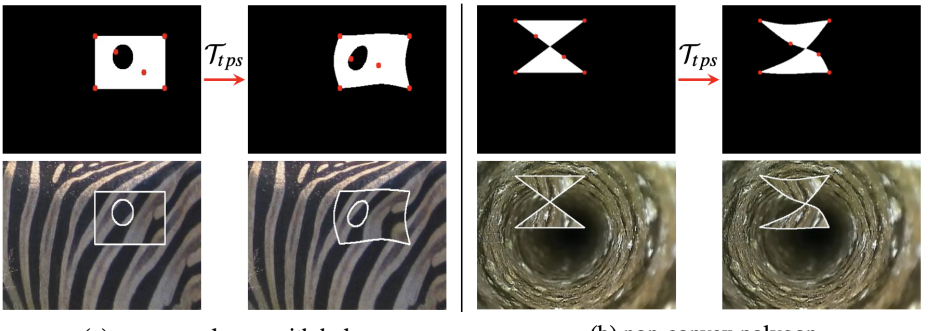
In this section, we describe a scalable pipeline for generating synthetic video sequences to train our proposed motion segmentation model. Specifically, we generate RGB sequences with objects being textured with samples from the DTD texture dataset [8], the optical flow sequences can either be computed from groundtruth transformations, or by an off-the-shelf flow estimator, *e.g.* RAFT [40]. In the following sections, we detail the generation process, and show an example of the generated video sequence in Figure 4. We refer the reader to supplementary material for more figures and details on the generation procedure.

3.1 Foreground Objects

Synthetic shapes. We start with simple 2D shapes, namely a polygonal sprite, which is generated from 4-vertex polygons, convex and non-convex, with random holes. To increase the complexity, we occasionally replace polygons with real objects masks, for example, silhouettes that are sourced from a large scale dataset, *e.g.* YouTube-VOS.

Non-rigid objects. Apart from affine transformations on the entire object, we also generate non-rigid motions by applying thin plate splines $\mathcal{T}_{tps}$ with 6 control points. Let $P_{tl}, P_{tr}, P_{br}, P_{bl}$ denote the vertices of the generated polygon starting from top left to bottom left. The control points are chosen at these vertices and two additional points located at $(x_{P_{tl}} + 0.3 * w, y_{P_{tl}} + 0.3 * h)$ and $(x_{P_{br}} - 0.3 * w, y_{P_{br}} - 0.3 * h)$, where h and w stand for the maximal height and width of the polygon.

Articulated objects. To simulate the articulated objects, we split the object into 2 to 4 parts and apply different 2D rotations around the designated articulation vertices, with rotation angles $\theta \in \left[-\frac{\pi}{3}, +\frac{\pi}{3}\right]$.



(a) convex polygon with hole (b) non-convex polygon

Figure 2: **Generating foreground objects.** Generated object masks with thin plate spline $\mathcal{T}_{tps}$, the control points are shown in red.

3.2 Generating Motion Sequences

At this stage, we compose moving objects onto a canvas that undergoes an independent transformation, shown in Figure 3. For the optical flow, we consider two options: one is to use the groundtruth flow based on the applied transformations, and the other is to run an off-the-shelf flow estimator on the RGB sequence, *e.g.* RAFT. In the latter case, the background scenes can also be sourced from other videos. Additionally, we also incorporate

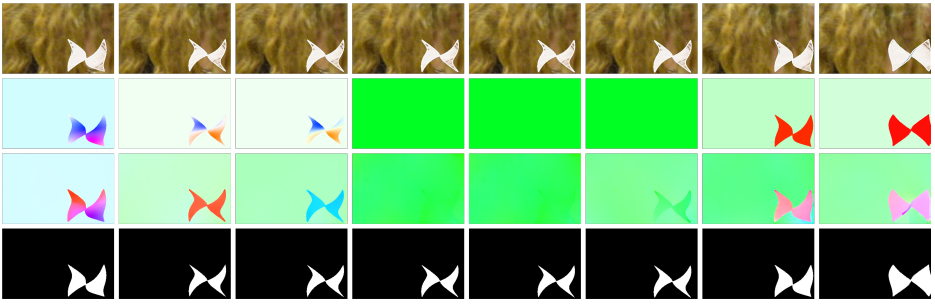


Figure 3: **Sequence of moving sprite undergoing a static subsequence.** From top to bottom: RGB frames, groundtruth optical flows, RAFT flows, segmentation masks.

static sub-sequences at various temporal locations, with the foreground object following the same transformation as the background. In such cases, objects will temporally disappear in the flow field, forcing the model to develop a sense of object permanence through temporal information.

3.3 Artificial Occluders

To simulate partial occlusions, we superimpose occluders on the canvas. They can be of arbitrary shapes, and follow the *same* motion as the background. As the occluders' trajectories intersect with the foreground object, the generated groundtruth segmentation masks will fall into two categories: masks that only delineate the visible parts of the object; or masks that delineate the objects as a whole, including the occluded parts. Both are shown in Figure 4.

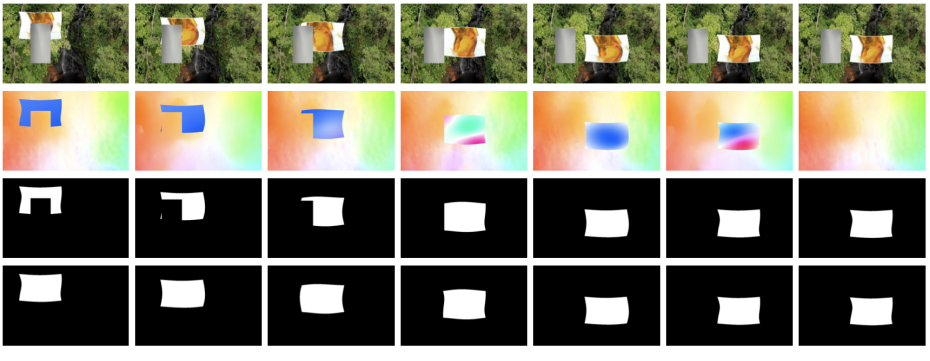


Figure 4: **Sequence of a moving sprite undergoing occlusion.** From top to bottom: RGB frames with the background sourced from a real video, and occluders are selected from different textures; optical flow sequence that is computed with RAFT; visible object (modal) segmentation; amodal segmentation.

4 Architecture

In this section, we present our proposed architecture for robustly segmenting moving objects, as illustrated in Figure 5. It is composed of a hybrid architecture, with ConvNets and Transformers. Specifically, the ConvNets are used to extract features from each input optical flow image, ending up as a sequence of feature maps, with the Transformer built on top to aggregate spatial-temporal information, and output framewise segmentations.

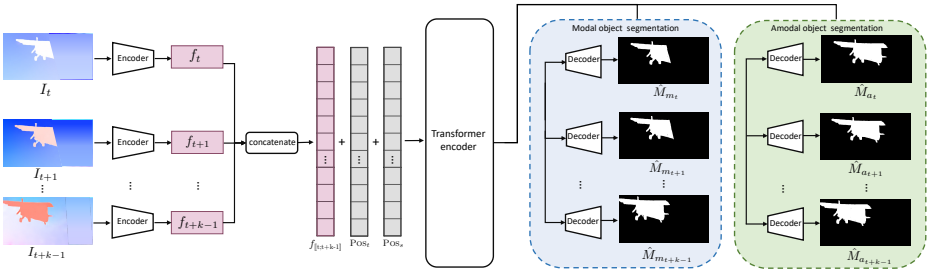


Figure 5: **Illustration of our proposed dual-head hybrid architecture for independent motion segmentation.** Our model takes a sequence of optical flows as input, and encodes each with a light-weight ConvNet, then positional spatio-temporal information is added to the outputted features $f_{[t,t+k-1]}$ before passing through the transformer encoder. Lastly, the output is fed into the dual-head decoder, predicting amodal $\hat{M}_{a_i}$ and modal masks $\hat{M}_{m_i}$.

Motion Encoder. Given a sequence of optical flows computed from consecutive frames, $X = \{I_t, I_{t+1}, ...I_{t+k-1}\} \in \mathbb{R}^{k\times C_0\times H_0\times W_0}$, we process the motion fields with a ConvNet:

$$\{f_t, f_{t+1}, \ldots, f_{t+k-1}\} = \Phi_{\text{ENC}}(X)$$

where $f_i \in \mathbb{R}^{C\times H\times W}$ refers to the feature map, H, W, C denote height, width and channels.

Transformer Encoder. As inputs to the Transformer Encoder, we reshape the feature maps to a sequence of tokens:

$$\mathbb{V} = \text{RESHAPE}(\{f_t, f_{t+1}, \ldots, f_{t+k-1}\}) + \text{POS}_t + \text{POS}_s$$

with $\mathbb{V} \in \mathbb{R}^{C\times kHW}$ representing a total of kHW vectors. To keep track of the spatial-temporal position for each token, learnable spatial (POS_s) and temporal (POS_t) encodings are also added. To this end, we pass the sequence of vectors into a standard Transformer Encoder with N layers ($N = 3$ in our case), each consisting of a stack of Multi-Head self-Attention, Layer Normalisation, and residual connections:

$$\overline{\mathbb{V}} = \text{TRANSFORMER-ENCODER}(\mathbb{V})$$

$\overline{\mathbb{V}} \in \mathbb{R}^{C\times kHW}$ refers to the output from the Transformer Encoder.

Discussion. Unlike existing video segmentation works that mainly rely on RGB sequences, we only use optical flow as input. This has both pros and cons. On the one hand, flow has been shown to be highly effective in discarding the nuisance factors in visual appearance, *e.g.* texture, illumination, greatly facilitating the *Sim2Real* procedure to generalise towards real videos. On the other hand, it also brings challenges to segment the objects if they stop moving or are partially occluded at certain points of the video, as they cannot be seen explicitly in the flow fields. At this point, the Transformer layers play a critical role for aggregating temporal information, effectively capturing the object permanence, *i.e.* the object does not vanish even if it stops moving, instead, it just stays in the same place. In Section 6, we experimentally validate the usefulness of such Transformer layers.

Dual-Head Decoder. The outputs from the Transformer Encoder are passed into a dual-head decoder, with one head dedicated to predicting framewise amodal moving object masks $\hat{M}_{a_i}$, and the other one focuses on the visible or modal moving object masks $\hat{M}_{m_i}$, as shown

in Figure 5. Symmetrically, the decoder uses a similar architecture to the encoder, with convolution transposes to recover high resolution, and incorporating skip layers in a U-Net-like manner to fuse fine-grained features. We refer readers to the supplementary material for more detailed architecture descriptions.

Training. We train the hybrid dual-head architecture using pixel-wise classification:

$$\mathcal{L} = \frac{1}{k} \sum_{i=t}^{t+k-1} (\mathcal{L}_{BCE}(\hat{M}_{m_i}, M_{m_i}) + \mathcal{L}_{BCE}(\hat{M}_{a_i}, M_{a_i}))$$

5 Experiments

In this section, we start by describing the evaluation benchmarks and metrics, followed by the implementation details.

5.1 Evaluation Benchmarks

We conduct evaluation on four different datasets, three of them are used for unsupervised video object segmentation, and one synthetic dataset to test amodal segmentation.

DAVIS2016 [33] contains a total of 50 sequences (30 for training and 20 for validation), depicting diverse moving objects such as animals, people, and cars. The dataset contains 3455 1080p frames with pixel-wise annotations for the predominantly moving objects.

SegTrackv2 [24] contains 14 sequences and 976 annotated frames. Each sequence has 1-6 moving objects, with challenges from motion blur, appearance change, complex deformation, occlusion, slow motion, and interacting objects.

Moving Camouflaged Animals (MoCA) [22] contains 141 HD video sequences, depicting 67 kinds of camouflaged animals moving in natural scenes. Both temporal and spatial annotations are provided in the form of tight bounding boxes for every 5th frame. Following [50], we use the provided motion labels (locomotion, deformation, static) to filter out videos with predominantly no locomotion, resulting in 88 video sequences and 4803 frames.

Dataset for Amodal Segmentation (AMSeg). We evaluate on two datasets for amodal segmentation. Specifically, in terms of quantitative evaluation, we test on a synthetic dataset generated with moving objects and occluder, totalling 500 sequences with $15K$ frames. While for qualitative visualisaiton, we use a subset of DAVIS2016 sequences, with the masks manually completed for occluded regions. More sample sequences can be found in the supplementary material. We will release the generated sequences and annotations in the project webpage.

5.2 Evaluation Metrics

Depending on the provided annotations, we use two different evaluation metrics.
Segmentation (Jaccard). For DAVIS2016, SegTrackv2, and AMSeg, pixelwise segmentations are available, we report region similarity ($\mathcal{J}$) over the test set. For SegTrackv2, we follow the common practice [16, 52] and combine multiple objects as one single foreground.

Localization (Jaccard & Success Rate). For MoCA, as only bounding box annotations are provided, we report results in the form of detection success rate [11, 26], under different IoU thresholds ($\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$).

5.3 Implementation Details

To get flow sequences on synthetic and real videos, we adopt off-the-shelf estimators, *e.g.* PWC-Net [38], RAFT [40]. For all datasets, optical flows are computed at the original resolution between image pairs, and converted into 3-channel images with the standard colorwheel used in the flow community [1, 50]. This transformation is equivalent to the vector field representation with the advantage of offering an easier visualisation of the flow sequences. Our architecture takes as input sequence of size $X \in \mathbb{R}^{k\times 3\times H_0\times W_0}$, $H_0 = W_0 = 128$. Note that, for simplicity, all notations here have ignored the *batch* dimension (we use a batch size of 8 for training). The Motion Encoder projects the input samples to feature maps with smaller spatial resolution, $\mathbb{R}^{k\times C\times H\times W}$, with $C = 512$, $H = \frac{H_0}{16}$ and $W = \frac{W_0}{16}$. For the Transformer Encoder, we use multi-head attention with 8 heads, and 3 layers. During training, we adopt the Adam optimiser with a learning rate of 5×10^{-4}. Instead of densely processing k consecutive frames, we proceed with a random selection of $l < k$ frames with corresponding absolute positional encoding (within the original sequence of length k). Effectively, this operation acts as a temporal dropout of input flows, reducing the required memory for training long video sequences, here, we use $k = 16$ and $l = 8$ in our training. At inference time, we process the entire sequence with all the k temporal samples.

6 Results

Here, we conduct ablation studies by varying one variable at a time, *e.g.* critical components for data simulation, and necessity of transformer for developing object permanence. After that, we compare with other state-of-the-art approaches on different benchmarks.

6.1 Ablation Studies

As shown in Table 1, we report the results by training models on synthetic data generated with increasing complexities. Note that, all models are directly applied to MoCA, DAVIS2016 and AMSeg datasets, *without* any fine-tuning included.

Choice of optical flow. While comparing the results of models trained on different optical flows, two phenomenon can be observed: *First*, training on RAFT flows are preferable than on groundtruth flow, *e.g.* Ours-(B,C) vs. Ours-(D,E). As during inference time, RAFT flow is also used, effectively introducing minimal domain gaps between training and testing scenarios. *Second*, RAFT provides higher quality flow estimations compared to other methods, *e.g.* PWCNet, showing superior results *cf.* Ours-(F,G).

Choice of background motion. Unsurprisingly, while training our proposed model, simulating backgrounds with a homography transformation substantially outperforms that of only a static background, *e.g.* Ours-A vs. Ours-B; using frames from other videos as background, *i.e.* real motions are introduced, can further boost the performance on downstream benchmarks, *e.g.* Ours-E vs. Ours-F.

Artificial occluder. While comparing Ours-E to Ours-D, we validate the usefulness of introducing artificial occluders for data simulation, this is especially true on AMSeg dataset, which is specifically designed to evaluate model's ability on amodal segmentation.

Transformer Encoder. We conduct experiment by ablating the Transformer Encoder, as shown by Ours-F vs. Ours-H, the former shows superior performance on all the datasets,

	Sequence	Bg Motion	Flow	Occluder	Transformer	MoCA	DAVIS2016	AMSeg
Ours-A	OF	Static	GT	✗	✓	23.1	10.4	11.4
Ours-B	OF	Homography	GT	✗	✓	54.1	44.2	71.5
Ours-C	OF	Homography	GT	✓	✓	52.5	48.8	69.2
Ours-D	RGB	Homography	RAFT	✗	✓	57.7	50.1	84.7
Ours-E	RGB	Homography	RAFT	✓	✓	59.9	52.2	90.3
Ours-F	**RGB**	**Homography + Real**	**RAFT**	✓	✓	**68.6**	**67.8**	**91.0**
Ours-G	RGB	Homography + Real	PWCNet	✓	✓	62.9	56.4	81.1
Ours-H	RGB	Homography + Real	RAFT	✓	✗	61.3	65.0	84.4

Table 1: **Ablation study on synthetic data generation.** Performance is measured by Jaccard on all datasets. Specifically, **Sequence** denotes the format of simulated video sequence, in RGB or optical flow (OF) (note that, in all cases, our pipeline only takes as input the optical flow sequence, either created during the simulation process or computed from the generated RGB sequences); **Bg Motion** refers to the background motion used during data simulation; **Flow** refers to the sources of optical flows for training. During inference, all the optical flow are computed from RAFT.

clearly demonstrating the necessity of Transformer Encoder, potentially on cases where objects stop moving or partially occluded at certain time point.

6.2 Comparison with State-of-the-art

In this section, we take the best setting discovered from the ablation study (**Ours-F**), and compare its segmentation results with previous approaches on the public benchmarks, *e.g.* MoCA, DAVIS2016, SegTrackv2. Again, in all cases, we train the model on synthetic data, and directly evaluate on the downstream tasks, *without* using any manual annotations.

6.2.1 On MoCA

As shown in Table 2, we achieve the state-of-the-art results on MoCA, significantly outperforming the previous approach (CIS [52] by over 19%, MoSeg [50] by over 4%). Notably, our proposed model even shows superior performance than the top supervised approaches, *e.g.* COSNet and MATNet.

					Success Rate					
Model	Sup.	RGB	Flow	$\mathcal{J}\uparrow$	$\tau=0.5$	$\tau=0.6$	$\tau=0.7$	$\tau=0.8$	$\tau=0.9$	SR_{mean}
CIS [52]	✗	✓	✓	49.4	0.556	0.463	0.329	0.176	0.030	0.311
MoSeg [50]	✗	✗	✓	64.2	0.712	0.670	0.599	0.492	0.246	0.544
Ours	✗	✗	✓	**68.6**	**0.772**	**0.717**	**0.623**	0.464	**0.255**	**0.566**
COD [27]	✓	✗	✓	44.9	0.414	0.330	0.235	0.140	0.059	0.236
COSNet [27]	✓	✓	✗	50.7	0.588	0.534	0.457	0.337	0.167	0.417
MATNet [64]	✓	✓	✓	64.2	0.712	0.670	0.599	**0.492**	0.246	0.544

Table 2: **Results on MoCA dataset.** Successful localization rate for various thresholds τ following the same metric proposed in [52].

6.2.2 On DAVIS2016 & SegTrackv2

In this section, we report results on DAVIS2016 and SegTrackv2. Notably, in these two benchmarks, motion is not playing the dominant role as it is in MoCA, as the objects can

often be well-identified by their appearance. This can be observed from the results of COSNet, which shows significantly stronger performance on DAVIS than on MoCA, despite not using any motion information.

As shown in Table 6a, our model trained on synthetic data still shows competitive performance, achieving state-of-the-art results on DAVIS2016 among all self-supervised learning approaches. While for SegTrackv2, the videos occasionally include multiple objects, with only a subset of them moving, in such case, no informative cues will support object segmentation in the flow input. We achieve competitive performance nonetheless.

Model	Sup.	RGB	Flow	DAVIS16	STv2
SAGE [51]	✗	✓	✓	42.6	57.6
NLC [13]	✗	✓	✓	55.1	**67.2**
CUT [21]	✗	✓	✓	55.2	54.3
FTS [35]	✗	✓	✓	55.8	47.8
CIS [57]	✗	✓	✓	59.2	45.6
Ours	✗	✗	✓	**67.8**	62.0
SFL [8]	✓	✓	✓	67.4	–
FSEG [18]	✓	✓	✓	70.7	61.4
LVO [46]	✓	✓	✓	75.9	57.3
ARP [41]	✓	✓	✓	76.2	57.2
COSNet [29]	✓	✓	✗	80.5	–
MATNet [59]	✓	✓	✓	82.4	–
3DC-Seg [30]	✓	✓	✓	**84.3**	–

(a) **Results on DAVIS2016 and SegTrackv2**

(b) **Qualitative Results on DAVIS2016, MoCA, SegTrackv2**

Figure 6: **Results on DAVIS2016 and SegTrackv2.**

Qualitative results. As shown in Figure 6b, the following phenomenon can be observed: *First*, our model provides high-quality segmentation masks, despite only being trained on optical flow computed from synthetic data; *Second*, the model is able to maintain the object permanence through frames. This is particularly evident for the second and fourth examples, where the flow fields fail to capture part of the moving object, or completely fail at a specific time step. Our model is able to recover the missing parts of the dancer in the second example, and the bird in the fourth column, while MATNet [54] only provides partial segmentations; *Third*, as can be seen in the third column, our amodal head is capable of estimating the correct shape of the object despite the occlusion by jumping poles.

7 Conclusions

To summarise, this paper considers the problem of segmenting *moving* objects in videos. Specifically, we propose a scalable pipeline for generating synthetic video sequences, with objects moving independently to the background motion. This has shown to be effective for resolving the challenge from limited availability of training data; In addition, we introduce a dual-head architecture (hybrid of ConvNet and Transformer), which enables to jointly process a sequence of optical flows, and segment the moving objects at any time point, outputting both modal (only visible parts) and amodal (object as a whole) segmentations. After training our proposed model on synthetic video sequences, we show *state-of-the-art* performance on the MoCA camouflage datasets, even outperforming supervised approaches. On DAVIS2016 and SegTrackv2, our proposed model also demonstrates competitive performance, despite not using *any* manual annotation.

Acknowledgements.

This research is supported by the UK EPSRC funded CDT in Autonomous Intelligent Machines and Systems (AIMS), the EPSRC Programme Grant VisualAI EP/T028572/1, a Schlumberger Studentship, and a Royal Society Research Professorship.

References

[1] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 2011.

[2] Pia Bideau and Erik Learned-Miller. It's moving! a probabilistic model for causal motion segmentation in moving camera videos. In *Proc. ECCV*, 2016.

[3] Pia Bideau and Erik Learned-Miller. A detailed rubric for motion segmentation. *arXiv preprint arXiv:1610.10033*, 2016.

[4] Pia Bideau, Rakesh R Menon, and Erik Learned-Miller. Moa-net: self-supervised motion segmentation. In *Proc. ECCV Workshop*, 2018.

[5] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *Proc. ECCV*, 2010.

[6] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012.

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

[8] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proc. CVPR*, 2014.

[9] Achal Dave, Pavel Tokmakov, and Deva Ramanan. Towards segmenting anything that moves. In *Proc. ICCV Workshop*, 2019.

[10] Carl Doersch and Andrew Zisserman. Sim2real transfer learning for 3d human pose estimation: motion to the rescue. In *Proc. NeurIPS*, 2019.

[11] Mark Everingham, Luc Van Gool, Chris K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010.

[12] Deng-Ping Fan, Wenguan Wang, Ming-Ming Cheng, and Jianbing Shen. Shifting more attention to video salient object detection. In *Proc. CVPR*, 2019.

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

[14] Richard I. Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[15] Allan Jabri, Andrew Owens, and Alexei A. Efros. Space-time correspondence as a contrastive random walk. In *Advances in Neural Information Processing Systems*, 2020.

[16] Suyog Jain, Bo Xiong, and Kristen Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos. *arXiv preprint arXiv:1701.05384*, 2017.

[17] 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 *Proc. CVPR*, 2017.

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

[19] Margret Keuper, Bjoern Andres, and Thomas Brox. Motion trajectory segmentation via minimum cost multicuts. In *Proc. ICCV*, 2015.

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

[21] Zihang Lai, Erika Lu, and Weidi Xie. MAST: A memory-augmented self-supervised tracker. In *Proc. CVPR*, 2020.

[22] Hala Lamdouar, Charig Yang, Weidi Xie, and Andrew Zisserman. Betrayed by motion: Camouflaged object discovery via motion segmentation. In *Proc. ACCV*, 2020.

[23] Trung-Nghia Le, Tam V Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *CVIU*, 2016.

[24] Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M. Rehg. Video segmentation by tracking many figure-ground segments. In *Proc. ICCV*, 2013.

[25] Ke Li and Jitendra Malik. Amodal instance segmentation. In *European Conference on Computer Vision*, pages 677–693. Springer, 2016.

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollar. Microsoft coco: Common objects in context. In *Proc. ECCV*, 2014.

[27] 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 *Proc. CVPR*, 2019.

[28] K-K Maninis, Sergi Caelles, Yuhua Chen, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. Video object segmentation without temporal information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018.

[29] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proc. CVPR*, 2016.

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

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

[32] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *Proc. ICCV*, 2013.

[33] 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 *Proc. CVPR*, 2016.

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

[35] Pulak Purkait, Christopher Zach, and Ian Reid. Seeing behind things: Extending semantic segmentation to occluded regions. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1998–2005. IEEE, 2019.

[36] Anurag Ranjan, Varun Jampani, Lukas Balles, Deqing Sun, Kihwan Kim, Jonas Wulff, and Michael J. Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *Proc. CVPR*, 2019.

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

[38] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proc. CVPR*, 2018.

[39] Deqing Sun, Daniel Vlasic, Charles Herrmann, Varun Jampani, Michael Krainin, Huiwen Chang, Ramin Zabih, William T Freeman, and Ce Liu. Autoflow: Learning a better training set for optical flow. In *CVPR*, 2021.

[40] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Proc. ECCV*, 2020.

[41] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning motion patterns in videos. In *Proc. CVPR*, 2017.

[42] Pavel Tokmakov, Cordelia Schmid, and Karteek Alahari. Learning to segment moving objects. *International Journal of Computer Vision*, 2019.

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

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

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

[46] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *Proc. CVPR*, 2019.

[47] Max Wertheimer. Untersuchungen zur lehre von der gestalt. ii. *Psychologische forschung*, 4(1):301–350, 1923.

[48] Christopher Xie, Yu Xiang, Zaid Harchaoui, and Dieter Fox. Object discovery in videos as foreground motion clustering. In *Proc. CVPR*, 2019.

[49] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv:1809.03327*, 2018.

[50] Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. In *Proc. ICCV*, 2021.

[51] Gengshan Yang and Deva Ramanan. Learning to segment rigid motions from two frames. In *Proc. CVPR*, 2018.

[52] Yanchao Yang, Antonio Loquercio, Davide Scaramuzza, and Stefano Soatto. Unsupervised moving object detection via contextual information separation. In *Proc. ICCV*, 2019.

[53] Zhao Yang, Qiang Wang, Luca Bertinetto, Song Bai, Weiming Hu, and Philip H.S. Torr. Anchor diffusion for unsupervised video object segmentation. In *Proc. ICCV*, 2019.

[54] Tianfei Zhou, Shunzhou Wang, Yi Zhou, Yazhou Yao, Jianwu Li, and Ling Shao. Motion-attentive transition for zero-shot video object segmentation. In *AAAI*, volume 34, pages 13066–13073, 2020.

[55] Yan Zhu, Yuandong Tian, Dimitris Metaxas, and Piotr Dollár. Semantic amodal segmentation. In *Proc. CVPR*, pages 1464–1472, 2017.