Betrayed by Motion: Camouflaged Object Discovery via Motion Segmentation

Hala Lamdouar, Charig Yang, Weidi Xie, and Andrew Zisserman

Visual Geometry Group, University of Oxford
{lamdouar,charig,weidi,az}@robots.ox.ac.uk
http://www.robots.ox.ac.uk/~vgg/data/MoCA

Abstract. The objective of this paper is to design a computational architecture that discovers camouflaged objects in videos, specifically by exploiting motion information to perform object segmentation. We make the following three contributions: (i) We propose a novel architecture that consists of two essential components for breaking camouflage, namely, a differentiable registration module to align consecutive frames based on the background, which effectively emphasises the object boundary in the difference image, and a motion segmentation module with memory that discovers the moving objects, while maintaining the object permanence even when motion is absent at some point. (ii) We collect the first large-scale Moving Camouflaged Animals (MoCA) video dataset, which consists of over 140 clips across a diverse range of animals (67 categories). (iii) We demonstrate the effectiveness of the proposed model on MoCA, and achieve competitive performance on the unsupervised segmentation protocol on DAVIS2016 by only relying on motion.

Keywords: Camouflage Breaking; Motion Segmentation;

1 Introduction

We consider a fun yet challenging problem of breaking animal camouflage by exploiting their motion. Thanks to years of evolution, animals have developed the ability to hide themselves in the surrounding environment to prevent being noticed by their prey or predators. Consider the example in Figure 1a, discovering the fish by its appearance can sometimes be extremely challenging, as the animal's texture is indistinguishable from its background environment. However, when the fish starts moving, even very subtly, it becomes apparent from the motion, as shown in Figure 1c. Having the ability to segment objects both in still images, where this is possible, and also from motion, matches well to the two-stream hypothesis in neuroscience. This hypothesis suggests that the human visual cortex consists of two different processing streams: the ventral stream that performs recognition, and the dorsal stream that processes motion [1], providing strong cues for visual attention and detecting salient objects in the scene.

In recent years, computer vision research has witnessed tremendous progress, mainly driven by the ability of learning effective representations for detecting,

(a) t-th frame. (b) (t+1)-th frame. (c) Optical flow.

Fig. 1: Two consecutive frames from the camouflage dataset, with a bounding box denoting the salient object. When the object starts moving, even subtly, we are able to detect it more easily, as shown, in the computed optical flow.

segmenting and classifying objects in *still images*. However, the assumption that objects can be well-segmented by their appearance alone is clearly an oversimplification; that is to say, if we draw an analogy from the two-stream hypothesis, the computer vision systems trained on images can only mimic the function of the ventral stream. The goal of this paper is to develop a computational architecture that is able to process motion representations for breaking camouflage, *e.g.* by taking optical flow (Figure 1c) as input, and predicting a segmentation mask for the animal of interest.

Unfortunately, simply relying on motion will not solve our problem completely, as, *first*, optical flow estimation itself remains extremely challenging and is under active research. In practice, modern optical flow estimation techniques provide a fairly good indication of rough object motion, but not fine-grained details, *e.g.* the exact shape of the objects and their contours. To compensate for the missing details, we propose to use a differentiable registration module for aligning consecutive frames, and use the difference of the registered images as auxiliary information to determine the exact contour; *Second*, if the motion stops at certain points in the video sequence, then a memory module is required to maintain the object permanence, as is done in [2], *i.e.* to capture the idea that objects continue to exist even when they cannot be seen explicitly in the motion representation.

Another main obstacle encountered when addressing the challenging task of camouflage breaking is the limited availability of benchmarks to measure progress. In the literature, there is a Camouflaged Animals video dataset, released by Bideau *et al.* [3], but this has only 9 clips on 6 different kinds of animals (about 840 frames). To overcome this limitation, we collect a **Mo**ving **C**amouflaged **A**nimal dataset, termed **MoCA**, which consists of 141 video sequences (37K frames), depicting 67 kinds of camouflaged animals moving in natural scenes. Both temporal and spatial annotations are provided in the form of tight bounding boxes on every 5th frame and the rest are linearly interpolated.

To summarize, in this paper, we make the following contributions: *First*, we propose a novel architecture with two essential components for breaking camouflage, namely, a differentiable registration module to align the background (regions other than the camouflaged animal) of consecutive frames, which effectively

highlights the object boundary in the difference image, and a motion segmentation module with memory that discovers moving objects, while maintaining the object permanence even when motion is absent at some point. *Second*, we collect the first large-scale video camouflage benchmark (MoCA), which we release to the community for measuring progress on camouflage breaking and object tracking. *Third*, we demonstrate the effectiveness of the proposed model on MoCA, outperforming the previous video segmentation approaches using motion. In addition, we also benchmark on DAVIS2016, achieving competitive performance on the unsupervised segmentation protocol despite using only motion. Note that, DAVIS2016 is fundamentally different from MoCA, in the sense that the objects are visually distinctive from the background, and hence motion may not be the most informative cue for segmentation.

2 Related Work

Our work cuts across several areas of research with a rich literature, we can only afford to review some of them here.

Video Object Segmentation [3, 4, 5, 6, 7, 8, 9, 2, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23] refers to the task of localizing objects in videos with pixel-wise masks. In general, two protocols have recently attracted an increasing interest from the vision community [4, 5], namely unsupervised video object segmentation (**unsupervised VOS**), and semi-supervised video object segmentation (**semi-supervised VOS**). The former aims to automatically separate the object of interest (usually the most salient one) from its background in a video sequence; and the latter aims to re-localize one or multiple targets that are specified in the first frame of a video with pixel-wise masks. The popular methods to address the unsupervised VOS have extensively relied on a combination of appearance and motion cues, *e.g.* by clustering trajectories [6, 7], or by using two stream networks [8, 9, 2, 10]; or have purely used appearance [14, 22, 24, 25]. For semi-supervised VOS, prior works can roughly be divided into two categories, one is based on mask propagation [11, 12, 13, 14, 15, 16], and the other is related to few shot learning or online adaptation [17, 18, 19].

Camouflage Breaking [3, 26] is closely related to the unsupervised VOS, however, it poses an extra challenge, as the object's appearance from *still image* can rarely provide any evidence for segmentation, *e.g.* boundaries. As such, the objects or animals will only be apparent when they start to move. In this paper, we are specifically interested in this type of problem, *i.e.* breaking the camouflage in a class-agnostic manner, where the model takes no prior knowledge of the object's category, shape or location, and is asked to discover the animal with pixel-wise segmentation masks whenever they move.

Image Registration/Alignment is a long-standing vision problem with the goal of transferring one image to another with as many pixels in correspon-

dence as possible. It has been applied to numerous applications such as video stabilization, summarization, and the creation of panoramic mosaics. A comprehensive review can be found in [27, 28]. In general, the pipeline usually involves both correspondence estimation and transformation estimation. Traditionally, the alignment methods apply hand-crafted features, *e.g.* SIFT [29], for keypoint detection and matching in a pair of images, and then compute the transformation matrix by solving a linear system. To increase the robustness of the geometric transformation estimation, RANdom SAmple Consensus (RANSAC) [30] is often adopted. In the deep learning era, researchers have constructed differentiable architectures that enable end-to-end optimization for the entire pipeline. For instance, [31] proposed a differentiable RANSAC by relaxing the sparse selection with a soft-argmax operation. Another idea is to train a network with binary classifications on the inliers/outliers correspondences using either ground truth supervision or a soft inlier count loss, as in [32, 33, 34, 35], and solve the linear system with weighted least squares.

3 A video registration and segmentation network

At a high level, we propose a novel computational architecture for breaking animal camouflage, which *only* considers motion representation as input, *e.g.* optical flow, and produces the segmentation mask for the moving objects. Specifically, as shown in Figure 2, the model consists of two modules: (i) a differentiable registration module for aligning consecutive frames, and computing the *difference* image to highlight the fine-grained detail of the moving objects, *e.g.* contours and boundaries (Section 3.2); and (ii) a motion segmentation network, which takes optical flow together with the difference image from the registration as input, and produces a detailed segmentation mask (Section 3.3).

3.1 Motion Representation

In this paper, we utilise optical flow as a representation of motion. Formally, consider two frames in a video sequence, I_t and I_{t+1}, each of dimension $\mathbb{R}^{H\times W\times 3}$, the optical flow is defined as the displacement field $F_{t\rightarrow t+1} \in \mathbb{R}^{H\times W\times 2}$ that maps each pixel from I_t to a corresponding one in I_{t+1}, such that:

$$I_t(\mathbf{x}) = I_{t+1}(\mathbf{x} + F_{t\rightarrow t+1}(\mathbf{x})), \tag{1}$$

where $\mathbf{x}$ represents the spatial coordinates (x, y) and F represents the vector flow field in both horizontal and vertical directions. In practice, we use the pretrained PWCNet [36] for flow estimation, some qualitative examples can be found in Figure 1 and Figure 5.

3.2 Differentiable Registration Module

One of the main challenges of segmentation with optical flow is the loss of rich fine-grained details due to motion approximations. In order to recover the sharp

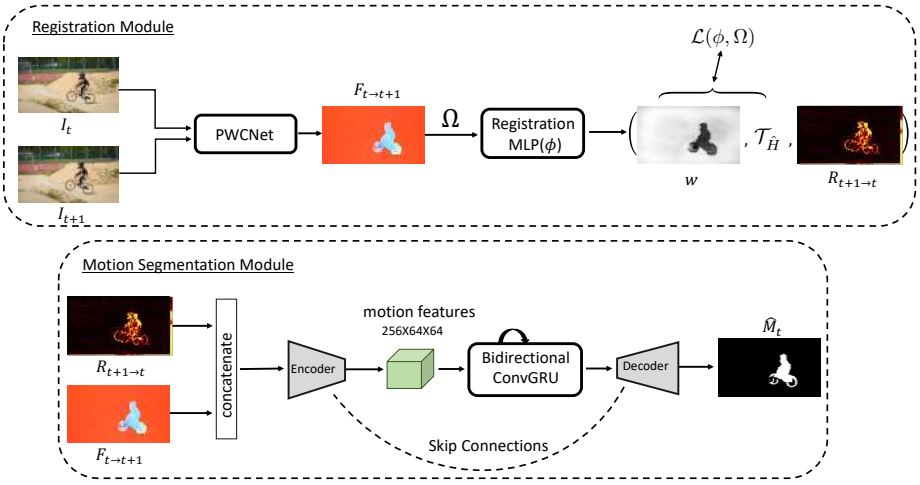


Fig. 2: Architecture Overview. The proposed architecture is composed of two different modules, namely registration and motion segmentation.

contours of the objects under motion, we seek a low level RGB signal which ideally suppresses the background and highlights the object's boundaries. In this paper, we use the image difference between consecutive frames after camera motion compensation. To this end, a reasonable assumption to make is that the foreground object undergoes an independent motion with respect to the global transformation of the background. We propose a differentiable registration module for estimating this transformation, which we approximate with a homography ($\mathcal{T}_{\hat{H}}$) between consecutive frames, and then compute the *difference* image after alignment ($R_{t+1\to t} = |\mathcal{T}_{\hat{H}}(I_{t+1}) - I_t|$), which will provide cues for the animal contours.

The key here is to train a registration module that accepts a set of correspondences obtained from the consecutive frames, outputs an homography transformation matrix (H), and an inlier weight map (w), which, ideally, acts like a RANSAC process, and has 1's for every background pixel, and 0's for every foreground pixel (moving object's). In this paper, we parametrize the registration module with Multiple Layer Perceptrons (MLPs), *i.e.* $[H, w] = \phi(\{p_s, p_t\}; \theta_r)$, where $p_s \in \mathcal{R}^{mn\times 2}$ denotes the spatial coordinates of all pixels (normalized within the range $[-1, 1]$) in the source image, and their corresponding position in the target image ($p_t \in \mathcal{R}^{mn\times 2}$), based on the computed optical flow, and θ_r are the trainable parameters.

Homography Transformation. In order to be self-contained, we summarise here the homography computation. Mathematically, a homography transformation ($\mathcal{T}_H$) maps a subset of points $\mathcal{S}_s$ from the source image to a subset of points

$\mathcal{S}_t$ in the target image; in our case, the source and target images refer to I_{t+1} and I_t respectively:

$$\forall {p_i}^s \in \mathcal{S}_s,\ \exists {p_i}^t \in \mathcal{S}_t \quad {p_i}^t = \mathcal{T}_H({p_i}^s) = \alpha_i H {p_i}^s, \tag{2}$$

where H is the matrix associated with the homography transformation $\mathcal{T}_H$ with 8 degrees of freedom, and α_i a non-zero scalar. This formulation can be expressed using homogeneous coordinates of ${p_i}^s$ and ${p_i}^t$ as:

$$\begin{pmatrix} {x_i}^t \\ {y_i}^t \\ 1 \end{pmatrix} = \alpha_i H \begin{pmatrix} {x_i}^s \\ {y_i}^s \\ 1 \end{pmatrix}. \tag{3}$$

Using the standard Direct Linear Transform (DLT) [37], the previous equation can be written as:

$$A\ vec(H) = \mathbf{0}, \tag{4}$$

where $vec(H) = \begin{pmatrix} h_{11}\ h_{12}\ h_{13}\ h_{21}\ h_{22}\ h_{23}\ h_{31}\ h_{32}\ h_{33} \end{pmatrix}^T$ is the vectorised homography matrix and A the data matrix. The homography H can therefore be estimated by solving such over-complete linear equation system. For more details on the DLT computation, refer to the extended ArXiv version of this paper.

Training objective. In order to train the registration module ($\phi(\cdot;\theta_r)$), we can optimize:

$$\mathcal{L} = \frac{1}{\sum w} \sum_{\Omega} w \cdot ||\mathcal{T}_H(p_s) - p_t||_2 + R(w), \tag{5}$$

where Ω refers to all the pixels on the mn grid. Note that, the homography $\mathcal{T}_H$ transformation in this case can be solved with a simple weighted least square (WLS) and differentiable SVD [33] for parameter updating. To avoid trivial solution, where the weight map can be full of zeros that perfectly minimize the loss, we add a regularization term ($R(w)$), that effectively encourages as many inliers as possible:

$$R(w) = -\gamma \sum_{p \in \Omega} l_p - \frac{1}{mn} \sum_{\Omega} \big\{ l_p \cdot \log(w) + (1 - l_p) \cdot \log(1 - w) \big\}, \tag{6}$$
$$\text{where} \quad l_p = \sigma\{(\epsilon - ||\mathcal{T}_H(p_s) - p_t||_2)/\tau\}.$$

In our training, $\gamma = 0.05$, $\tau = 0.01$, $\epsilon = 0.01$ and $m = n = 64$. The first term in $R(w)$ (l_p) refers to a differentiable inlier counting [32, 34]. The rest of the terms aim to minimize the binary cross-entropy at each location of the inlier map, as in [38], forcing the predictions to be classified as inlier (1's) or outlier (0's).

3.3 Motion Segmentation Module

After introducing the motion representation and registration, we consider a sequence of frames from the video, $I \in \mathcal{R}^{T\times 3\times H\times W}$, where the three channels refer to a concatenation of the flow (2 channels) and difference image (1 channel). For simplicity, here we use a variant of UNet [39] with the bottleneck feature maps being recurrently processed.

Specifically, the **Encoder** of the segmentation module will independently process the current inputs, ending up with motion features of $\mathcal{R}^{T\times 256\times 64\times 64}$, where $T, 256, 64, 64$ refer to the number of frames, number of channels, height, and width respectively. After the Encoder, the **memory module** (a bidirectional convGRU [40] is used in our case) operates on the motion features, updating them by aggregating the information from time steps in both directions. The **Decoder** takes the updated motion features, and produces an output binary segmentation mask, *i.e.* foreground vs background. The Motion Segmentation Module is trained with pixelwise binary cross-entropy loss.

This completes the descriptions of the two individual modules used in the proposed architectures. Note that the entire model is trained together as it is end-to-end differentiable.

4 MoCA: a new Moving Camouflaged Animal dataset

One of the main obstacles encountered when addressing the challenging task of camouflage breaking is the limited availability of datasets. A comparison of existing datasets in Table 1. Bideau *et al.* published the first Camouflaged Animals video dataset with only 9 clips, and Le *et al.* proposed the CAMO dataset with only single image camouflage, and therefore not suitable for our video motion segmentation problem.

To overcome this limitation, we collect the **Mo**ving **C**amouflaged **A**nimal dataset, termed **MoCA**, which consists of 141 video sequences depicting various camouflaged animals moving in natural scenes. We include the PWC-Net optical flow for each frame and provide both spatial annotations in the form of bounding boxes and motion labels every 5-th frame with the rest linearly interpolated.

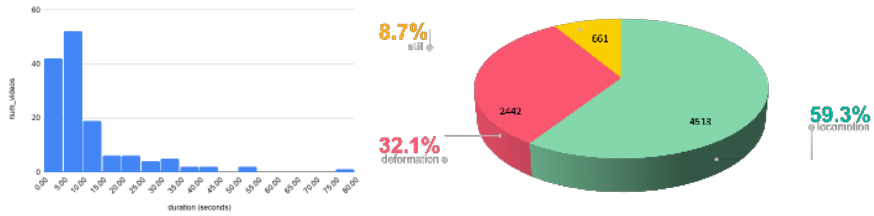
4.1 Detailed statistics

MoCA contains 141 video sequences collected from YouTube, at the highest available resolution, mainly 720×1280, and sampled at $24fps$. A total of 37250 frames spanning 26 minutes. Each video represents a continuous sequence depicting one camouflaged animal, ranging from 1.0 to 79.0 seconds. The distribution of the video lengths is shown in Figure 3a. The dataset is labelled with a bounding box in each frame, as well as a motion label for the type of motion. While annotating the data, we distinguish three types of motion:

- **Locomotion:** when the animal engages in a movement that leads to a significant change of its location within the scene *e.g.* walking, running, climbing, flying, crawling, slithering, swimming, *etc.*

Table 1: Statistics for recent camouflage breaking datasets; "# Clips" denotes the number of clips in the dataset; "# Images" denotes the number of frames or images in the dataset; "# Animals" denotes the number of different animal categories in the dataset; "Video" indicates whether videos are available.

Datasets	# Clips	# Images	# Animals	Video
Camouflaged Animals [41]	9	839	6	✓
CAMO [26]	0	1250	80	×
MoCA (Ours)	141	37K	67	✓



(a) Distribution of video lengths. (b) Distribution of motion labels.

Fig. 3: Statistics for the Moving Camouflaged Animals (MoCA) dataset. In (a), the x-axis denotes the video duration, and the y-axis denotes the number of video sequences. (b) the distribution of frames according to their motion types (still, deformation and locomotion), see text for the detailed definitions.

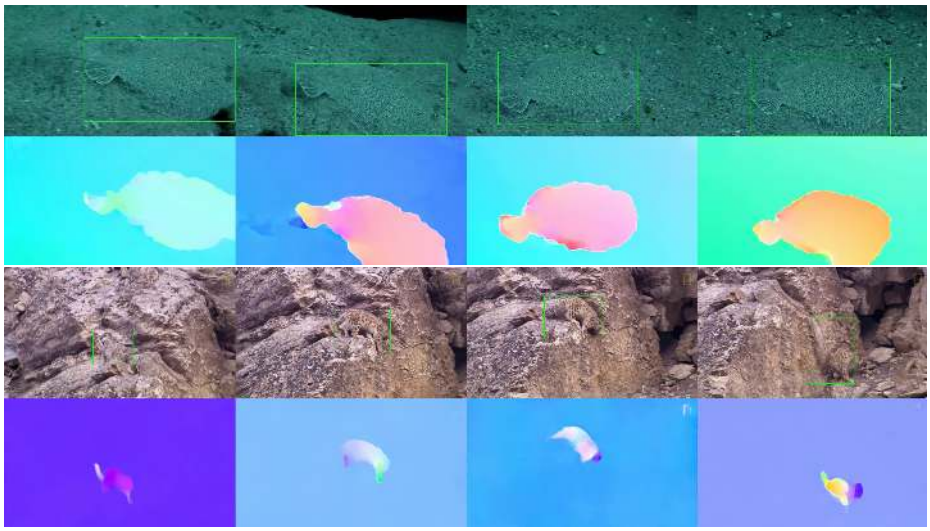


Fig. 4: Example sequences from the Moving Camouflaged Animals (MoCA) dataset with their corresponding optical flows.

- **Deformation:** when the animal engages in a more delicate movement that only leads to a change in its pose while remaining in the same location *e.g.* moving a part of its body.
- **Static:** when the animal remains still.

As shown in Figure 3b, motion wise, the dataset contains 59.3% locomotion, 32.1% deformations, and 8.7% still frames. Examples from the dataset are shown in Figure 4.

5 Experiments

In this section, we detail the experimental setting used in this paper, including the datasets, evaluation metrics, baseline approaches and training details.

5.1 Datasets

DAVIS2016 refers to the Densely Annotated VIdeo Segmentation dataset [42]. It consists of 50 sequences, 30 for training and 20 for testing, captured at $24fps$ and provided at two resolutions. We use the $480p$ version in all experiments. This dataset has the advantage that accurate pixelwise ground truth segmentations are provided, $3,455$ annotations in total, as well as spanning a variety of challenges, such as occlusions, fast-motion, non-linear deformation and motion-blur. We train the model on the DAVIS 2016 training set and report results on its validation splits.

Synthetic Moving Chairs. In order to train the differentiable registration module properly, we use video sequences that are synthetically generated. Specifically, we use the 3D-rendered objects from the Flying Chairs dataset as foreground, and take random images from YouTube-VOS as background. We then apply rigid motions, *e.g.* homographies, to the background, and simulate an independent motion for the foreground object. Note that, with such a synthetic dataset, we have complete information about the background homography transformation, optical flow, inlier maps, and object masks, which enables us to better initialise the registration module before training on real video sequences. These synthetic video sequences were only used to pre-train the registration and motion segmentation modules. The extended ArXiv version of this paper includes example images.

Evaluation Metrics. Depending on the benchmark dataset used, we consider two different evaluation metrics. For **DAVIS2016**, we follow the standard protocol for unsupervised video object segmentation proposed in [42], namely, the mean region similarity $\mathcal{J}$, which is the intersection-over-union (IOU) of the prediction and ground truth; and mean contour accuracy $\mathcal{F}$, which is the F-measure defined on contour points from the prediction and the ground truth. For **MoCA**, as we only have the bounding box annotations, we define the metric

as the IOU between the groundtruth box and the minimum box that includes the predicted segmentation mask. Note that, we follow the same protocol used in Bideau *et al.* [3], meaning, we only evaluate the segmentation of the animals under locomotion or deformation (*not* in static frames).

5.2 Baselines

We compare with five previous state-of-the-art approaches [43, 2, 10, 22, 23]. In [2], Tokmakov *et al.*, the LVO method uses a two-stream network for motion segmentation, where the motion stream accepts optical flow as input via a MPNet [43] architecture, and the appearance stream uses an RGB image as input. Similar to ours, a memory module is also applied to recurrently process the frames. A more recent approach [10] adapts the Mask-RCNN architecture to motion segmentation, by leveraging motion cues from optical flow as a bottom-up signal for separating objects from each other, and combines this with appearance evidence for capturing the full objects. For fair comparison across methods, we use the same optical flow computed from PWCNet for all the flow-based methods. To this end, we re-implement the original MPNet [43], and train on the synthetic FT3D dataset [44]. For LVO [2], we also re-train the pipeline on DAVIS2016. For Seg-det [10] and Seg-track [10], we directly replace the flows from Flownet2 [45] with the ones from PWCNet in the model provided by the authors. In all cases, our re-implemented models outperform or match the performance reported in the original papers. In addition, we also compare our method to AnchorDiff [22] and COSNet [23], both approaches have been trained for unsupervised video object segmentation with only RGB video clips, and show very strong performance on DAVIS.

5.3 Training and Architecture Details

Registration Module. We adopt the architecture of [38], which is MLPs with 12 layer residual blocks. We first train the registration module on the Synthetic Moving Chairs sequences described in 5.1, for $10K$ iterations using an Adam optimizer with a weight decay of 0.005 and a batch size of 4. For a more stable training, we use a lower learning rate, *i.e.* 5×10^{-5}, avoiding the ill-conditioned matrix in the SVD.

Motion Segmentation Module. We adopt a randomly initialized ResNet-18. Frame-wise segmentation is trained from scratch on the synthetic dataset, together with the pre-trained registration module. We further include the bidirectional ConvGRU and finetune the whole pipeline on DAVIS 2016, with each sequence of length 11, batch size of 2, for a total of $25K$ iterations. For all trainings, we use frames with a resolution of $\mathcal{R}^{256\times256\times3}$.

6 Results

In this section, we first describe the performance of our model and previous state-of-the-art approaches on the new MoCA dataset, and then compare segmentation performance on the DAVIS2016 benchmark.

Table 2: Mean Intersection Over Union on MoCA for the different motion type subsets. "All_Motion" refers to the overall performance

Input	Model	RGB	Flow	Register	Memory	Locomotion	Deform	All_Motion
Flow	MPNet [43]	×	✓	×	✓	21.3	**23.5**	22.2
	ours-A	×	✓	×	✓	31.3	17.8	27.6
	ours-B	×	✓	MLP	×	29.9	15.6	25.5
	ours-C	×	✓	MLP	✓	**47.8**	20.7	**39.4**
	ours-D	×	✓	RANSAC	✓	42.9	19.2	35.8
RGB	AnchorDiff [22]	✓	×	×	×	30.9	29.4	30.4
	COSNet [23]	✓	×	×	×	35.9	35.1	36.2
Both	LVO [2]	✓	✓	×	✓	30.6	34.9	30.6
	Seg-det [10]	✓	✓	×	×	16.9	18.7	17.9
	Seg-track [10]	✓	✓	×	×	29.9	32.2	30.2
	ours-E	✓	✓	MLP	✓	**45.0**	**38.0**	**42.4**

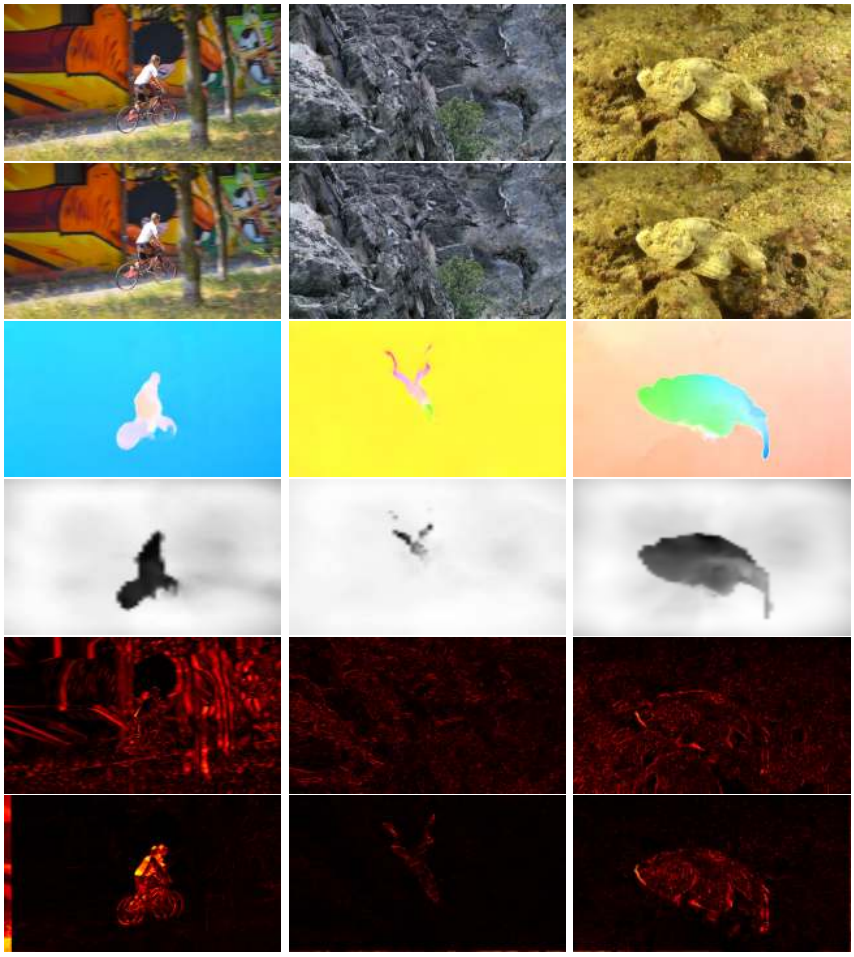
6.1 Results on the MoCA Benchmark

We summarise all the quantitative results in Table 2 and discuss them in the following sections. In Figure 5, we illustrate the effect of the differentiable registration and Figure 6 shows examples of the overall segmentation method.

Effectiveness of registration/alignment. To demonstrate the usefulness of the differentiable registration module, we carry out an ablation study. By comparing ours-A (without registration), ours-D (registration using RANSAC) and Ours-C (registration using our trainable MLPs), it is clear that the model with trainable MLPs for alignment helps to improve both the animal discovery on video sequences with locomotion and deformation, outperforming ours-A (without registration) and ours-D (RANSAC).

In Figure 5 we visualise the results from the registration module, *e.g.* the inlier map, difference image before alignment (second last row) and after alignment (last row). It is clear that the difference images computed after alignment are able to ignore the background, and successfully highlight the boundary of the moving objects, *e.g.* the wheels of the bicycle from the first column.

Effectiveness of memory. Comparing model-B (without memory) and model-C (with memory), the only difference lies on whether the frames are processed



(a) (b) (c)

Fig. 5: Registration results from (a) the validation set of DAVIS 2016; and (b)(c) MoCA. From top to bottom: Frame t, Frame $t+1$, Forward PWCNet optical flow, inlier weights (background pixels are shown as gray or white), image difference without alignment, aligned image difference.

individually or recurrently. As shown by the results, a significant boost is obtained with the help of the memory module (25.5 vs. 39.4 on All_Motion), showing its effectiveness.

Comparison to baselines and previous approaches.

From Table 2, we make the following observations: *First*, when comparing with MPNet [43], which also processes optical flow, our model demonstrates a superior performance on All_Motion (39.4 vs 22.2), and an even larger gap on

Locomotion, showing the usefulness of image registration and memory modules; *Second*, as expected, the state-of-the-art unsupervised video segmentation approaches relying on appearance (RGB image as input), *e.g.* AnchorDiff [22] and COSNet [23], tend to struggle on this camouflage breaking task, as the animals often blend with the background, and appearance then does not provide informative cues for segmentation, emphasising the importance of motion information (by design) in this dataset; *Third*, when we adopt a two-stream model, *i.e.* extend the architecture with a Deeplabv3-based appearance model, and naively average the prediction from appearance and flow models, the performance can be further boosted from 39.4 to 42.4, significantly outperforming all the other two-stream competitors, *e.g.* LVO [2], Seg-det and Seg-seg [10].

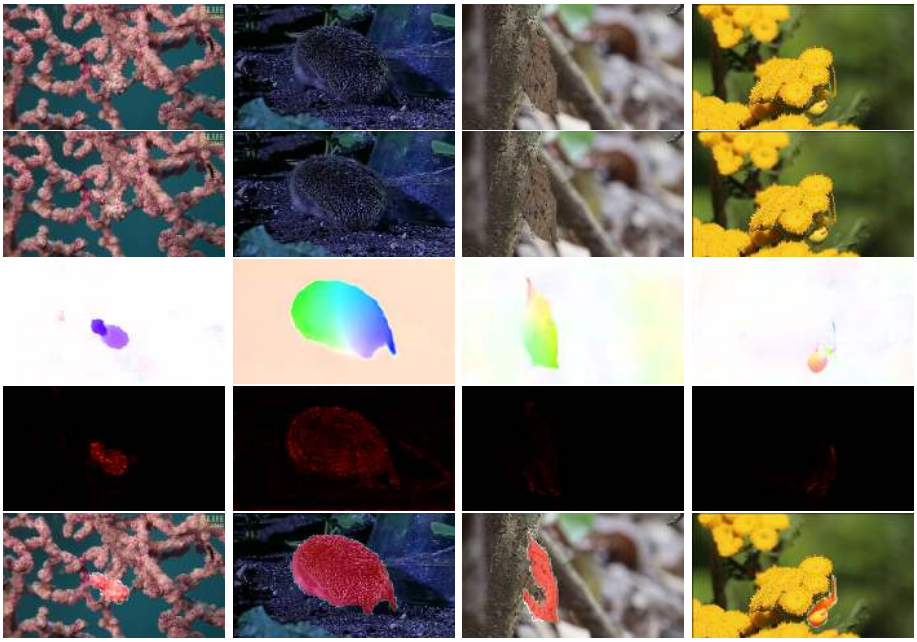


Fig. 6: Motion segmentation results on MoCA. From top to bottom: frame t, frame $t+1$, PWCNet optical flow, aligned image difference, moving object segmentation.

6.2 Results on DAVIS2016 Benchmark

We compare to previous approaches on the Unsupervised Video Object Segmentation protocol. In all experiments, we *do not* use any post-processing *e.g.* CRF. Both our re-trained MPNet and LVO model outperform the original reported results in the papers, which guarantees a fair comparison with their model. For

Seg-det, replacing the PWC-Flow leads to a small drop (around 2.3%) in the mean $\mathcal{J}$, but this will not affect our conclusion here.

As shown in Table 3, when compared with MPNet which also take flow-only input, our model (ours-C) outperforms it on all metrics by a large margin. Note that, the DAVIS benchmark is fundamentally different from MoCA, as the approaches only relying on appearance are very effective [22], indicating that the objects in the DAVIS sequences can indeed be well-identified by the appearance, and motion is not playing the dominant role as it is in MoCA. This can also be observed from the results for Seg-det, Seg-track, AnchorDiff and COSNet, which show significantly stronger performance on DAVIS than on MoCA. Given this difference, our flow-based architecture still shows very competitive performance. Moreover, when we extend our model with an RGB appearance stream, we do observe a performance boost, but since our appearance model has only been finetuned on the DAVIS training set (30 training sequences), the two stream (ours-E) is not comparable with other models trained with more segmentation data.

Table 3: Results on the validation set of DAVIS 2016.

		Flow-based		RGB-based		Two-stream			
	Measure	MPNet[43]	**ours-C**	AD[22]	COSNet[23]	LVO[2]	[10]-det	[10]-track	**ours-E**
$\mathcal{J}$	Mean	60.3	**65.3**	**80.3**	77.6	69.8	**76.8**	75.8	69.9
	Recall	69.9	**77.3**	90.0	**91.4**	83.8	84.8	81.8	**85.3**
$\mathcal{F}$	Mean	58.7	**65.1**	**79.3**	77.5	70.1	**77.8**	76.1	70.3
	Recall	64.3	**74.7**	84.7	**87.4**	84.3	**91.3**	88.7	82.9

7 Conclusions

To summarise, in this paper we consider the problem of breaking animal camouflage in videos. Specifically, we propose a novel and effective architecture with two components: a differentiable registration module to highlight object boundaries; and a motion segmentation module with memory that discovers moving regions. As future work, we propose to improve the architecture from two aspects: *First*, building more effective memory module for handling longer video sequences, for example, a Transformer [46]. *Second*, for objects that are only partially moving, RGB appearance is required to get the sense of objectness, therefore a future direction is to explore RGB inputs, for both visual-matching-based registration and appearance features.

Acknowledgements This research was supported by the UK EPSRC CDT in AIMS, Schlumberger Studentship, and the UK EPSRC Programme Grant Seebibyte EP/M013774/1.

References

1. Goodale, M.A., Milner, A.D.: Separate visual pathways for perception and action. Trends in Neurosciences **15** (1992) 20–25
2. Tokmakov, P., Schmid, C., Alahari, K.: Learning to segment moving objects. IJCV (2019)
3. Bideau, P., Learned-Miller, E.: A detailed rubric for motion segmentation. arXiv preprint arXiv:1610.10033 (2016)
4. Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., Gool, L.V.: The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:1704.00675 (2017)
5. Xu, N., Yang, L., Fan, Y., Yue, D., Liang, Y., Yang, J., Huang, T.: Youtube-vos: A large-scale video object segmentation benchmark. In: Proc. ECCV. (2018)
6. Brox, T., Malik, J.: Object segmentation by long term analysis of point trajectories. In: Proc. ECCV. (2010)
7. Ochs, P., Brox, T.: Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In: Proc. ICCV. (2011)
8. Papazoglou, A., Ferrari, V.: Fast object segmentation in unconstrained video. In: Proc. ICCV. (2013)
9. Jain, S.D., Xiong, B., Grauman, K.: Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In: Proc. CVPR. (2017)
10. Dave, A., Tokmakov, P., Ramanan, D.: Towards segmenting anything that moves. In: ICCV Workshop on Holistic Video Understanding. (2019)
11. Oh, S.W., Lee, J.Y., Xu, N., Kim, S.J.: Video object segmentation using space-time memory networks. In: Proc. ICCV. (2019)
12. Voigtlaender, P., Chai, Y., Schroff, F., Adam, H., Leibe, B., Chen, L.C.: Feelvos: Fast end-to-end embedding learning for video object segmentation. In: Proc. CVPR. (2019)
13. Vondrick, C., Shrivastava, A., Fathi, A., Guadarrama, S., Murphy, K.: Tracking emerges by colorizing videos. In: ECCV. (2018)
14. Wang, W., Lu, X., Shen, J., Crandall, D.J., Shao, L.: Zero-shot video object segmentation via attentive graph neural networks. In: Proc. ICCV. (2019)
15. Lai, Z., Xie, W.: Self-supervised learning for video correspondence flow. In: Proc. BMVC. (2019)
16. Lai, Z., Lu, E., Xie, W.: MAST: A memory-augmented self-supervised tracker. In: Proc. CVPR. (2020)
17. Maninis, K.K., Caelles, S., Chen, Y., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: Video object segmentation without temporal information. (2018)
18. Voigtlaender, P., Leibe, B.: Online adaptation of convolutional neural networks for video object segmentation. arXiv (2017)
19. Caelles, S., Maninis, K.K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: One-shot video object segmentation. In: Proc. CVPR. (2017)
20. Fragkiadaki, K., Zhang, G., Shi, J.: Video segmentation by tracing discontinuities in a trajectory embedding. In: Proc. CVPR. (2012)
21. Keuper, M., Andres, B., Brox, T.: Motion trajectory segmentation via minimum cost multicuts. In: Proc. ICCV. (2015)
22. Yang, Z., Wang, Q., Bertinetto, L., Bai, S., Hu, W., Torr, P.H.: Anchor diffusion for unsupervised video object segmentation. In: Proc. ICCV. (2019)

23. Xiankai, L., Wenguan, W., Chao, M., Jianbing, S., Ling, S., Fatih, P.: See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In: Proc. CVPR. (2019)
24. Jun Koh, Y., Kim, C.S.: Primary object segmentation in videos based on region augmentation and reduction. In: Proc. CVPR. (2017)
25. Fan, D.P., Wang, W., Cheng, M.M., Shen, J.: Shifting more attention to video salient object detection. In: Proc. CVPR. (2019)
26. Le, T.N., Nguyen, T.V., Nie, Z., Tran, M.T., Sugimoto, A.: Anabranch network for camouflaged object segmentation. CVIU (2016)
27. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press, ISBN: 0521540518 (2004)
28. Szeliski, R.: Image alignment and stitching: A tutorial. Technical Report MSR-TR-2004-92 (2004)
29. Lowe, D.: Object recognition from local scale-invariant features. In: Proc. ICCV. (1999) 1150–1157
30. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Comm. ACM **24** (1981) 381–395
31. Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., Rother, C.: Dsac-differentiable ransac for camera localization. In: Proc. CVPR. (2017)
32. Brachmann, E., Rother, C.: Learning less is more-6d camera localization via 3d surface regression. In: Proc. CVPR. (2018)
33. Ranftl, R., Koltun, V.: Deep fundamental matrix estimation. In: Proc. ECCV. (2018)
34. Rocco, I., Arandjelovic, R., Sivic, J.: End-to-end weakly-supervised semantic alignment. In: Proc. CVPR. (2018)
35. Brachmann, E., Rother, C.: Neural- Guided RANSAC: Learning where to sample model hypotheses. In: Proc. ICCV. (2019)
36. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In: Proc. CVPR. (2018)
37. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press, ISBN: 0521540518 (2004)
38. Moo Yi, K., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., Fua, P.: Learning to find good correspondences. In: Proc. CVPR. (2018)
39. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Proc. MICCAI. (2015)
40. Ballas, N., Yao, L., Pal, C., Courville, A.: Delving deeper into convolutional networks for learning video representations. In: Proc. ICLR. (2016)
41. Bideau, P., Learned-Miller, E.: It's moving! a probabilistic model for causal motion segmentation in moving camera videos. In: Proc. ECCV. (2016)
42. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: Proc. CVPR. (2016)
43. Tokmakov, P., Alahari, K., Schmid, C.: Learning motion patterns in videos. In: Proc. CVPR. (2017)
44. Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: Proc. CVPR. (2016)
45. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep network. In: Proc. CVPR. (2017)

46. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: NIPS. (2017)