SWIFT - an efficient and effective application of instance segmentation and tracking in wildlife monitoring

Frank Schindler[1,*] *Volker Steinhage*[1]

[1] Department of Computer Science IV, University of Bonn, Bonn, Germany
** E-mail: schindler@cs.uni-bonn.de*

Abstract: Instance segmentation and tracking are topics that have been little explored in the context of wildlife monitoring, but provide an essential basis for further tasks such as population estimation or behavioral analysis. In this paper, we highlight the importance of these topics and show how they can be efficiently and effectively addressed using our own multi-object tracking and segmentation (MOTS) approach, SWIFT. For this purpose, we provide an overview of our three past publications on these topics. Moreover, we evaluate SWIFT on two datasets, our self-created wildlife camera trap video dataset Wildpark Daylight containing videos of red deer and fallow deer and the Wildlife Crossings dataset containing four different animal classes. Our own dataset is one of the very few datasets in wildlife monitoring that is annotated with instance masks and tracking IDs. SWIFT significantly improves the quality of the instance masks and also multi-object tracking accuracy scores compared to using state-of-the-art instance segmentation and tracking approaches on both datasets.

1 Introduction

Stationary camera traps that record videos of wildlife are a widely used tool in wildlife monitoring to monitor ecosystems. Camera traps are present in a wide range of ecological studies [12], [10] and are used more and more [16]. However, the generated data material from even very few camera traps placed at a site for a few weeks is so large that it requires enormous manual work from researchers and it takes months to sift through the recorded videos. The use of artificial intelligence allows to automate this process [33].

An instance segmentation detects animals in one frame of a video (or in general objects in an image) by assigning to each detection (1) a bounding box to locate it, (2) a segmentation mask to show its exact contour, (3) a class label to identify what kind of animal is present and (4) a score value to show how reliable the detection is. Tracking, more precisely multi-object tracking (MOT), combines the found detections of each frame into tracks by adding (5) a unique track ID to each detection. The combined task of instance segmentation and tracking is called multi-object tracking and segmentation (MOTS). So the goal of a MOTS pipeline in the area of wildlife monitoring is to detect and track animals in camera trap videos.

Before [58], [59] and [60], researchers have not focused on instance segmentation and tracking when analysing wildlife videos. If only individual images and no video data from camera traps are available as data material, either a classification of the overall image [17], [51], [72], [50], [81], [20], for example for the occurring animal species, or a detection with the help of a bounding box [71], [3], [4], [27], [11], [70] is usually carried out. Instance segmentation for images is mainly done for cattle in an enclosed environment [67], [57], [5], [39].

However, when video data is available, instance segmentation and tracking provide great added value for further analysis by ecologists. Instance masks allow better separation of animals moving in a group. Accordingly, if an abundance estimation [38], [73] is performed, a classification of a video is not sufficient to estimate the population. In addition to instance segmentation, tracking of animals prevents individuals from being counted twice within a video. For example, an animal can be briefly occluded by other individuals or objects and would possibly be perceived as a new instance in a simple detection without tracking. Other application areas for camera traps are the quantification of species diversity [68] detection and study of rare species [46] or the analysis of species replacement processes [15]. Exact detections of animals and tracking form an essential basis for more complex tasks such as action detection [58] or re-identification [61] of individual animals in multiple videos. Especially an action detection enables an efficient behavioral analysis [16] and [69] of wild animals with camera traps.

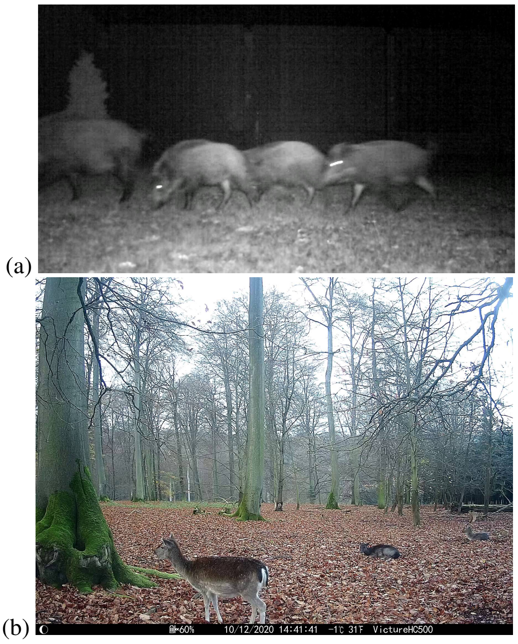

Fig. 1: Two exemplary frames from both datasets, the Wildlife Crossing Dataset (a) and our Rolandseck Daylight Dataset (b).

Detection in videos of great apes is performed by [90]. The apes are detected by bounding boxes, but there is no tracking and no instance masks. Tracking of animals is mainly considered for small animals like fish or insects in laboratory environments [30], [56] and [65]. Elephants are tracked in wildlife videos by [91], but they do not deliver MOT metric values for the tracking accuracy. [31] perform instance segmentation using a graph convolutional neural network and track the detections by analyzing the intersection over union of the detections. Finally, the authors determine the actions of the piglets. The videos viewed were made in a building in a fixed

© Copyright resides with the authors

environment and are not comparable to wildlife videos in changing terrain.

To our knowledge, the MOTS problem has not been previously considered in other work in the context of wildlife monitoring. [31] perform instance segmentation and tracking on piglets. But the videos were recorded in a building in a fixed environment, which is not comparable to changing environments of wildlife videos. [86] address the MOTS problem, but they only segment and track one animal at a time in a video. Moreover, for the segmentation a user interaction is required by manually setting an initial instance mask of the animal in a so-called guidance frame.

In our previous work [58] we analysed instance segmentation and a basic action recognition for animals in wildlife crossings. Building on these findings, we had extended our existing approach with tracking using the Tracktor [6] to use instance segmentation and tracking to automate the annotation of the data material [59]. In our latest publication we present our novel MOTS pipeline SWIFT [60]. In this paper we summarize our publications [58], [59] and [60] and explain the general importance of instance segmentation and tracking in the context of wildlife monitoring. In particular, we focus on SWIFT [60], which is our approach to solving these issues. Moreover, we newly evaluate SWIFT on the Wildlife Crossing dataset we used in the first two publications and show that SWIFT outperforms our previous results.

2 Related Work

The fields of instance segmentation, multi-object tracking (MOT) and multi-object tracking and segmentation (MOTS) are major research areas in computer vision. In the following, we give a short overview of the most relevant work that does not explicitly deal with the analysis of animal-related data.

2.1 Instance Segmentation

To perform instance segmentation, an object must first be detected. Based on the underlying detector, instance segmentation approaches are divided into two groups, the single-stage and two-stage approaches. Two-stage approaches result in more accurate instance masks, but are in general slower than single-stage approaches. Single-stage methods are faster than two-stage approaches, but in general are less accurate. Since real-time detection is not relevant in the application area of wildlife monitoring (the videos already recorded by camera traps are analyzed afterwards), the advantage of single-stage detectors is not significant for us. The surveys [36], [47] and [34] present a good overview over different instance segmentation and detection approaches.

One of the most famous two-stage instance segmentation approaches is Mask R-CNN [37]. It is still the superior network for deriving segmentation masks in images. Mask R-CNN outperforms all previous winners of the COCO segmentation challenge. Mask R-CNN often forms the basis for other instance segmentation approaches, such as the approaches of Huang et al. [40], Fang et al. [28], Chen et al. [18] and [48].

Famous representatives of single-stage instance segmentation are TensorMask [21] and SipMask [14]. TensorMask uses a sliding window approach to tackle very dense segmentation tasks with many different objects. SipMask preserves instance-specific information by dividing the predicted segmentation mask of an object into subregions in one bounding box. Other single-stage detectors are CenterMask [41], ESE-Seg [82], RDSNet [76],[13], BorderPointsMask [88] and MetricMask [77] .

Video instance segmentation approaches fall under the category of MOTS, since these approaches perform tracking simultaneously with instance segmentation and are therefore described in more detail in section 2.3.

2.2 Multi-object tracking MOT

The most common and successful way to perform MOT today is the tracking-by-detection paradigm. In a first step, a detection model localizes all objects in a video. The data association then combines the detected objects into tracks. This means that an improved detection model improves the tracking accuracy. For the identification of the object a bounding box is sufficient in MOT. Our SWIFT approach also falls into the tracking-by-detection category. In addition to tracking-by-detection approaches there are one-shot tracking approaches or joint-detection-and-tracking approaches. They perform detection and tracking in one network simultaneously. Although they generally produce poorer results, one advantage of these methods is that they are faster and therefore more often suitable for real-time applications.

The surveys of [24] and [84] provide a comprehensive compilation of different current approaches in the field of multi-object tracking.

A very successful and well known representative of the tracking-by-detection paradigm is the Tracktor approach by [6]. The Tracktor exploits the regression ability of a detection model to perform the data association. The major advantage is that no further training with tracking data is needed. ByteTrack [93] and the approach of [9] rely on a combination of a Kalman Filter and the Hungarain Matching algorithm. Further tracking-by-detection approaches are [83], [19] and [42].

The single-shot tracker FairMOT [94], JDE [79] and the approach by [92] simultaneously detect bounding boxes of the objects and extract Re-ID features to track the objects. [23] and [66] use Transformer networks to detect and track the objects. The tracking approaches SiamMOT [62] and SMOT [43] rely on modelling the motion to track the detected objects.

2.3 Multi-object tracking and segmentation MOTS

Multi-object tracking and segmentation MOTS combines the tasks of instance segmentation and tracking. Many MOTS approaches use Mask R-CNN for the instance segmentation and build the tracking framework around this basis. Mask R-CNN is a reliable basis and easily modifiable. Examples for this are Track R-CNN [74], SORTS [1], MOTSNet [54], MaskProp [8] [44], [55] and [89].

The VisTr, a transformer network with an encoder and decoder part, is used in the work of [78] to perform simultaneously instance segmentation and tracking. STEm-SEg [2] present an end-to-end trainable network that takes a video input in form of a 3D spatio-temporal volume, where the network learns an embedding for each pixel. Further MOTS approaches are ReMOTS [87], STCN [22], PointTrack [85] and PolyTrack [29].

3 Datasets

Already annotated datasets for the tasks of instance segmentation, MOT, and MOTS in the field of wildlife monitoring are almost nonexistent. In [58] and [59] we used data material from camera traps that were positioned at wildlife crossings at the federal motorway 7 near the city Oberthulba. The video data was provided by the Bavarian Highway Directorate, Germany, but we had to manually annotate all of the videos with the instance masks, tracking ids and animal classes. We call this dataset the Wildlife Crossings dataset. All videos are recorded at nighttime. Each video is about 10 seconds long with 8 fps (frames per second) and a resolution of 1280 x 720 pixels. The videos include red deer, wild boar, hares and foxes. This means that the videos have a rather low resolution, which makes it difficult to recognise the animals, and at the same time they have a low temporal resolution of 8 fps, which in turn leads to blurring when the animals move quickly.

For our study [60], we created our own dataset, the Rolandseck Daylight dataset. With permission of the Wildpark Rolandseck GmbH, we captured video footage of fallow deer and red deer in their natural environment in the Wildpark Rolandseck (Germany) from November 2020 to December 2021, resulting in over 6000 recorded videos. We used two *Vicutre HC500 Trail Cameras* as camera traps and placed them at different locations to show changing backgrounds and different settings. The manual annotation was performed with the VGG Image Annotator (VIA) Version 2.0.8 [25]

© Copyright resides with the authors

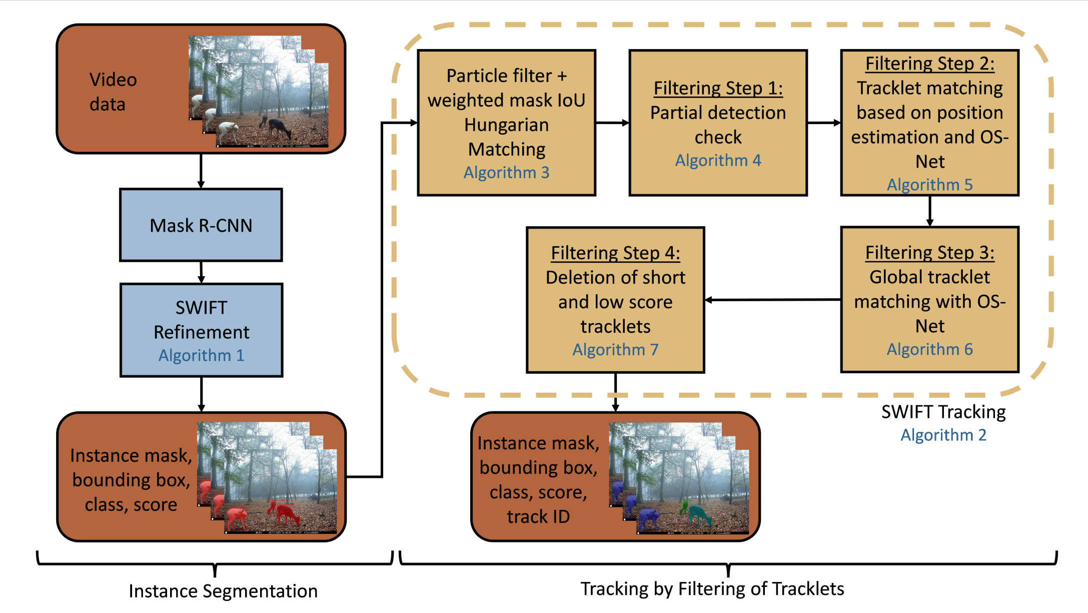
Video data
Mask R-CNN
SWIFT Refinement
Algorithm 1
Instance mask, bounding box, class, score
Particle filter + weighted mask IoU Hungarian Matching
Algorithm 3
Filtering Step 1: Partial detection check
Algorithm 4
Filtering Step 2: Tracklet matching based on position estimation and OS-Net
Algorithm 5
Filtering Step 3: Global tracklet matching with OS-Net
Algorithm 6
Filtering Step 4: Deletion of short and low score tracklets
Algorithm 7
SWIFT Tracking
Algorithm 2
Instance mask, bounding box, class, score, track ID
Instance Segmentation
Tracking by Filtering of Tracklets

Fig. 2: SWIFT detects the animals in a video in the instance segmentation part and forms tracks from those detections in the second part. [60]

and the annotation tool of [64]. For each animal in a frame a segmentation mask, a bounding box, a class label and a track ID is given. All videos are 30 seconds long with 30 fps (frames per second) and a high definition resolution of 1728 x 1296 pixels. Videos in the dataset were only shortened if there were no animals visible in the clips anymore. Our camera traps are equipped with PIR (Pyroelectric Infrared) sensors, which detect temperature changes in its field of view. Therefore the cameras are only triggered when an animal moves into the field of view (not for example by a moving branch). Our annotated dataset consists of 21 videos, which in turn correspond to 16566 individual frames. There are two animal classes present, red deer and fallow deer.

In this study, we evaluate SWIFT also on the Wildlife Crossings dataset to compare the results of our first publications with SWIFT and to show the generalizability of SWIFT.

Exemplary frames from both datasets are shown in Fig. 1.

4 SWIFT

In this section we present the functionality of SWIFT, Segmentation With Filtering of Tracklets. For a detailed explanation of the individual algorithms we refer to our publication [60].

The goal of SWIFT is to identify all animals in an input video by assigning (1) an instance mask, (2) a bounding box, (3) a class label, (4) a score value and (5) a unique track ID to each animal in each frame of the video. The workflow of SWIFT is shown in Fig. 2. First, SWIFT performs an instance segmentation with our refinement algorithm to detect all animals with their exact contours. This is followed by our tracking algorithm, which consists of several successive filtering steps to connect the found detections of all frames to tracks through the whole video.

4.1 SWIFT Instance Segmentation

The fundamental idea of SWIFT's instance segmentation is that a Mask R-CNN [37] trained on the wildlife data predicts reliable instance masks that are subsequently optimized by our refinement algorithm.

The choice for our base detection model is Mask R-CNN because it works reliably in different application scenarios. Therefore, Mask R-CNN often forms the basis for complex approaches in instance segmentation and MOTS (cf. Sec. 2.1). Furthermore, we have shown the effectiveness of Mask R-CNN for wildlife video data in our publications [58] and [59]. The backbone of Mask R-CNN extracts features from the input image (or frame). These features are the basis for the generation of regions of interest. Based on the regions of interest the bounding box and mask regression and classification is done in the head of the network. To improve the quality of the predicted masks we set the input resize of the Mask R-CNN to the exact resolution of our frames, which is higher than the default value, which would otherwise result in a reduction and corresponding loss of information of the input image. Moreover, we normalize the input frames by computing the image mean and standard deviation for our dataset. We replace the standard ResNet-50 by a ResNeXt-101 for a better feature extraction. The deeper ResNeXt-101 backbone is slower in training and inference, but extracts high quality features for better instance masks. We train the network for 60 epochs with an initial learning rate of 0.0005, a momentum of 0.9 and weight decay of 0.0005. The best values for these parameters were determined in different experiments. We reduce the learning rate after 25 and 45 epochs by a factor of 0.1. At these points, the loss no longer decreases and a smaller learning rate is necessary. However, a further reduction in the learning rate is not useful.

The inspiration for our refinement strategy comes from interactive segmentation of images. In this scenario a user sets positive and negative points that are either inside or outside the desired mask of the object. Usually an exact instance masks can be achieved with 3 to 10 clicks. We follow the approach of [64] for interactive segmentation. The authors have shown that their refinement network is superior to other interactive segmentation approaches. Especially the HR-Net backbone in the refinement network [75] enables the creation of high resolution instance masks, which is our goal. Our approach is fully automatic, therefore here is no user who can set clicks. The SWIFT refinement algorithm generates positive and negative clicks automatically by consideration of the dilated and eroded instance mask from the Mask R-CNN. The positive clicks can be sampled on the outline of the dilated (increased) instance mask. The negative clicks can be sampled in the same way by using the eroded (decreased) instance mask.

4.2 SWIFT Tracking

The SWIFT tracking algorithm tracks the refined detections from the SWIFT instance segmentation part. This means that each instance mask is assigned a unique track id to track all animals within a video. The tracking algorithm we designed works with different successive filtering steps. Each filtering step either deletes previously created tracklets or merges them into longer tracklets. As long as a track is not final, it is generally called a tracklet.

© Copyright resides with the authors

The initial tracklets are generated with a particle filter [32] and IoU matching with instance masks. The inspiration for this step comes from the simple, but very successful SORT tracking algorithm [9]. Instead of using the Intersection over Union values of the bounding boxes of the detections we use the IoU of the instance masks, which is useful for crowded situations like deer moving in a group. Furthermore, we change the motion model from a Kalman filter [80] to a particle filter [32]. The advantage of the particle filter compared to the Kalman filter in predicting the movement of the animals is that a particle filter can represent both linear and non-linear systems, while a Kalman filter is limited to linear systems.

The first filtering step deletes tracklets that contain another tracklet spatially and temporally. It is common that in the detection step not only the animal, but also parts like a leg or the head are detected as additional objects. These partial detections normally have lower confidence scores. It is also possible that an additional detection is larger than the animal, for example, because a tree or parts of another animal had a similar colour. In this case the larger detection will be deleted.

Filtering step 2 merges tracklets based on a position estimation and a Re-ID network, which is used to re-identify animals in the same video.

In the third filtering step, we match tracklets based only on the Re-ID features. Here we consider the cases that cannot be explained by the position estimation from filtering step 2, such as leaving and re-entering the scenery or a change of direction of the animal during its occlusion like by a tree. Therefore, we call this step global tracklet matching because there are no spatial constraints for the re-identification.

The final filtering step 4 deletes tracklets that are either very short or have a low overall score value.

4.3 Experiments

We use the standard COCO metrics to evaluate the instance segmentation. A detailed explanation is found in [58], [59], [52] and [26]. For the evaluation, we consider the average precision AP^{mask} and the average recall AR^{mask} of the instance masks. Both are calculated as an average over IoU threshold values ranging from $0.5, ..., 0.95$. Additionally, we report the average precision $AP_{0.5}^{mask}$ and $AP_{0.75}^{mask}$ that are computed for one specific IoU threshold. This means that $AP_{0.75}^{mask}$ represents a stricter metric than $AP_{0.5}^{mask}$.

The accuracy of the tracking is determined by the MOT metrics. Further explanations can be found in the works of [59], [7] and [49]. We measure the quality of our tracking results with the accuracy $MOTA$ and precision $MOTP$ of the MOT metrics. Moreover, we consider the false positives FP, false negatives FN and the id switches IDS. Each ground truth track is classified into three different categories: Mostly Tracked (MT), Partially Tracked (PT) and Mostly Lost (ML). The $IDF1$ metric describes the ratio of correctly identified detections over the average number of ground truth detections and observed detections.

Additionally, we use the MOTS metrics proposed by [74] for the combined task of instance segmentation and tracking. The MOTS metrics extend the MOT metrics by including instance masks instead of bounding boxes in the computation. The mask based metrics $MOTSA$ and $MOTSP$ describe the accuracy and the precision. Moreover, we use the $sMOTSA$ metric, which is a soft version of $MOTSA$ focusing on correct instance masks in the tracking process.

4.3.1 Implementation Details:

The training and testing was performed with a GeForce RTX 2080 Ti GPU with 11 GB graphic memory, 16 GB RAM and an Intel Core i7-6700K 4.00 GHz CPU.

All programs use Python 3 and PyTorch [53] for building and training the networks. The Mask R-CNN model is built around the PyTorch detection model of Mask R-CNN. The implementation of the refinement network is based on the official implementation from github from [64]. The HR-Net in this implementation is pretrained on the COCO dataset [45] and the LVIS dataset [35].

4.3.2 Evaluation Studies on SWIFT:

We examine the functionality of the individual parts of SWIFT by comparing them with other known competitive approaches. We start with the instance segmentation and refinement part and continue with the tracking algorithm. In this paper we newly evaluate SWIFT on the Wildlife Crossings dataset that we used in our previous work [58] and [59].

For the evaluation of SWIFT we split the Rolandseck Daylight dataset that we described in Sec. 3 in a train and test set. The train set contains 17 and the test set the remaining 4 videos. The test set consists of 2 red deer and 2 fallow deer videos. The test set represents the data set well, as it has a mix of challenging to moderately difficult videos in terms of detection and tracking. In the test videos are a total of 34 animal individuals. The Wildlife Crossings dataset is split with the same strategy, resulting in 33 videos in the train set and 8 video in the test set.

4.3.3 Instance Segmentation Evaluation:

We determine the quality of the instance segmentation of SWIFT by using the COCO metrics. We compare the results of our refinement algorithm with the results of the pure Mask R-CNN [37] without refinement. Moreover, we compare our choice of refinement network, the HR-Net [64], with the f-brs approach [63], which also uses positive and negative points for interactive segmentation. For the HR-Net, we examine two different execution modes in particular. As mentioned in Sec. 3, the approach of [64] was successfully used for the interactive annotation of the dataset. The Mask R-CNN network is trained as described in Sec.4.1.

With $setmask$ the refinement network, the HR-Net, offers the possibility to initialize the refinement process with a given instance mask. We analyse the results of the HR-Net with an initialized mask and without one.

The results of the instance segmentation are shown in Tab. 1. The best results in average precision AP^{mask} with 0.495 for the Rolandseck Daylight dataet are achieved by our approach SWIFT with the setmask function of HR-Net. SWIFT increases the quality of the instance masks significantly from the base value 0.432 of the Mask R-CNN. In particular, in the stricter $AP_{0.75}^{mask}$ metric, SWIFT shows an improvement over the baseline that means that the masks are more accurate due to the refinement. The average recall AR^{mask} is improved by using our approach as well as the average precision. In the same way, SWIFT shows a much better performance for the other dataset than the Mask R-CNN base model, improving the AP^{mask} from 0.503 to 0.532. The fact that SWIFT is improving the details of the masks in particular is also reflected here in the improvement of the stricter $AP_{0.75}^{mask}$ metric.

Fig. 3 shows exemplary results from the Rolandseck Daylight dataset for the instance segmentation and tracking with SWIFT in comparison to Mask R-CNN and Tracktor. The improvement in the accuracy of the instance masks with the refinement of SWIFT compared to Mask R-CNN is clearly visible here.

Our analysis of the refinement with the f-brs shows that not every refinement approach improves the instance masks successfully. To achieve an improvement, the high resolution of the frames must be exploited with the help of the HR-net. The accuracy and the stability of our approach profits significantly from the initialization of the HR-Net with the base instance mask. Even though more accurate instance masks can partially be generated by the HR-Net for animals standing alone without an initial mask, errors can occur more easily for groups of animals standing close together, since the similar coat structure of the animals makes them difficult to distinguish.

4.3.4 Tracking Evaluation:

We use the MOT metrics to determine the tracking capability of SWIFT and compare our approach with the well-known and successful Tracktor [6] approach. Furthermore, we compare the tracking results with our tracking algorithm using the original Mask R-CNN detections in comparison to the refined masks.

In Tab. 2 we show the tracking results of SWIFT using the MOT metrics. We analyse the Tracktor and its enhancement Tracktor++, which is also proposed by the authors [6]. Tracktor++ uses a simple motion model and a Re-ID network as extensions to the Tracktor. The SWIFT tracking improves both the MOTA value from 57.2% of

© Copyright resides with the authors

Table 1 Instance segmentation comparison: The best results are shown in bold.

Dataset	Method	AP^{mask}	$AP^{mask}_{0.50}$	$AP^{mask}_{0.75}$	AR^{mask}
Rolandseck Daylight	Mask R-CNN	0.432	0.752	0.451	0.509
	SWIFT (f-brs)	0.417	0.732	0.396	0.508
	SWIFT (HR-Net without setmask)	0.475	0.752	0.462	0.564
	SWIFT (HR-Net with setmask)	**0.495**	**0.762**	**0.540**	**0.575**
Wildlife Crossings	Mask R-CNN	0.503	0.897	0.517	0.650
	SWIFT (HR-Net with setmask)	**0.532**	**0.899**	**0.553**	**0.681**

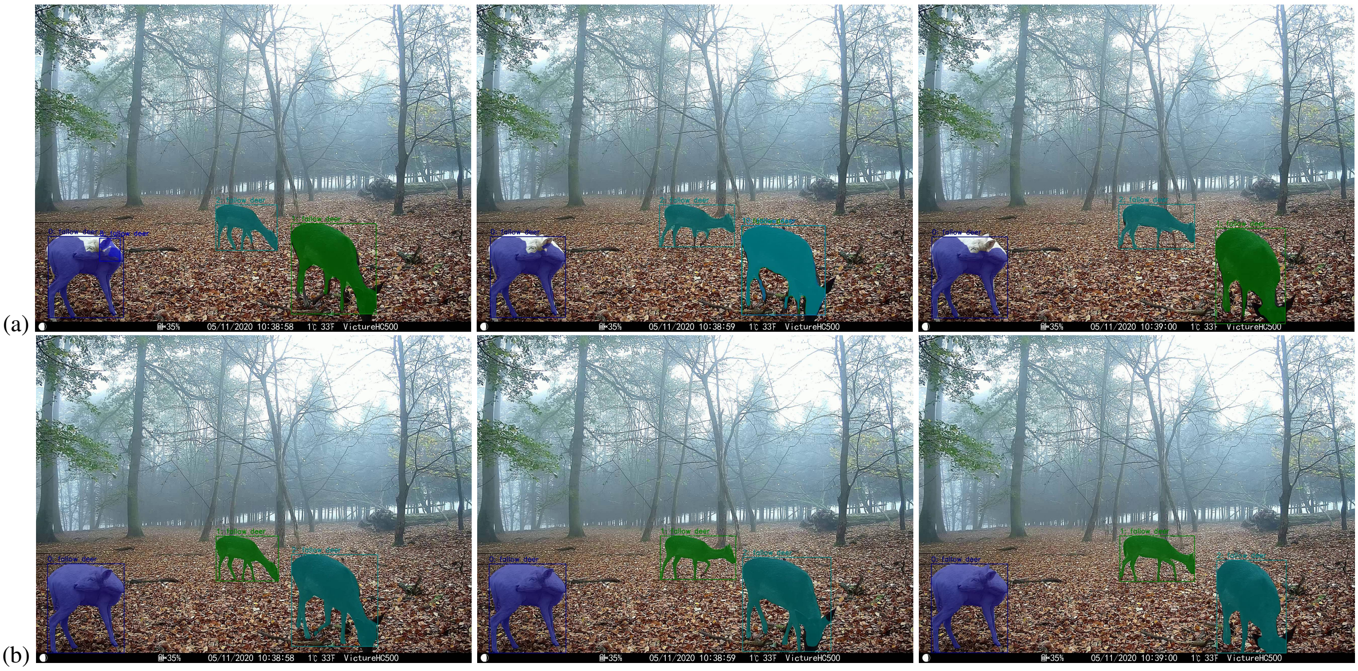

Fig. 3: Comparison of instance segmentation and tracking quality between Mask R-CNN + Tracktor++ (a) and SWIFT (b): Three frames of a video are displayed, each 1 second respectively 30 frames apart. Each found track is colored differently. [60]

the tracktor to 63.8% and the MOTP value from 84.5% to 86.0%. SWIFT's optimized tracking capability is also evident in the fact that SWIFT improves MOTA and MOTP values using the basic Mask R-CNN detections as input for tracking. The general advantage of a motion model and a Re-ID network is visible in the improvement of Tracktor to Tracktor++. SWIFT shows the highest number of mostly tracked ground truth tracks. Furthermore, the number of false positives and false negatives is significantly reduced by SWIFT due to its deletion and combining of tracklets. Tracktor++ shows less id switches and a higher IDF1 metric at the cost of worse tracking based on MOTA and MOTP. For the Wildlife Crossings dataset, the tracking performance of SWIFT with Mask R-CNN and with refined detections is almost the same. This is due to the significantly lower number of frames and instance masks in the (test) dataset, which is also visible in the number of FP and FN compared to our dataset.

In Fig. 3 we compare exemplary results for tracking with SWIFT in comparison to the Tracktor. Tracktor generates faulty tracks with partial detections of the animal, e.g., the head of the left animal in the first frame and the double detection of the right animal in the middle frame.

Additionally, we present the MOTS metric results for SWIFT using the Mask R-CNN detections in comparison to the refined detections in Tab. 3. The results show that using the refined instance masks for SWIFT significantly improves the MOTS metrics compared to the basic Mask R-CNN detections. In contrast to the small optimization of the MOTA score from 63.3% to 63.8%, the MOTSA value is significantly increased from 53.7% to 62.5%. This is also evident in the evaluation of SWIFT on the Wildlife Crossings dataset, where the MOTSA is improved from 73.2% to 76.3%. This improvement shows that the refinement of the instance masks in the SWIFT pipeline provides more reliable tracking results instead of using the Mask R-CNN detections. The sMOTSA value, which measures the joint segmentation and tracking quality of a system, is increased significantly as a result of SWIFT with the refined detections.

5 Discussion

Instance segmentation and tracking provide valuable information that is necessary for further or more complex analyses of the video data material. If an animal population is to be estimated on the basis of video data material, tracking is essential for an abundance estimation like [38], [73]. A detection of the animals alone would not enable a correct counting of the animals. During the recording an animal can be occluded by other animals or an object or can even leave and reenter the scene. Only by tracking the detections over the whole video it is possible to correctly detect all individuals. Instance masks are very useful for reliable tracking, as we have shown with SWIFT. Therefore, a MOTS pipeline is very useful for abundance estimations with video data.

To correctly describe actions of animals, instance segmentation and tracking are also very important. An action recognition or action classification like we analysed in our publication [58] is able to predict the actions of an animal in a video sequence without a detection of the animal. However, this only works until a single animal appears in the video. As soon as several individuals are in a scene and perform different actions at the same time, this is no longer possible.

© Copyright resides with the authors

Table 2 Tracking comparison: We compare the tracking results of the Tracktor with our SWIFT tracking algorithm. For SWIFT we differentiate between Mask R-CNN detections and the refined detections from our refinement algorithm. The test set of Rolandseck Daylight contains 3600 images with 23463 instances and the test set of Wildlife Crossings contains 5128 images with 632 instances. The depicted metrics are the MOT metrics. The best results are shown in bold.

Dataset	Method	MOTA ↑	MOTP ↑	FP ↓	FN ↓	IDS ↓	IDF1 ↑	MT	PT	ML
Rolandseck Daylight	Tracktor	57.2%	84.5%	5072	4479	492	56.0%	22	8	4
	Tracktor++	59.0%	84.5%	5061	4498	**56**	**66.1%**	22	8	4
	SWIFT (with Mask R-CNN detections)	63.3%	85.0%	4475	**4056**	69	61.7%	24	6	4
	SWIFT (with refined detections)	**63.8%**	**86.0%**	**4214**	4187	77	59.2%	24	6	4
Wildlife Crossings	**SWIFT** (with Mask R-CNN detections)	**64.1%**	**82.2%**	71	**149**	7	76.9%	9	4	3
	SWIFT (with refined detections)	63.0%	**82.2%**	**64**	166	**4**	76.9%	7	5	4

Action detection is needed for this. In order to correctly predict the actions of the animals, the animals must be detected and tracked. Then the actions for the individual detections can be determined. An exact outline of the animal through the instance mask can be very advantageous, especially for animals that are close to each other. A successful action detection would facilitate behavioral analysis like [16], [69].

The task of re-identification of individual animals like in [61] can benefit from an instance segmentation. Finding the exact contour of the individual helps to distinguish animals if they are close to each other. Bounding boxes as a result of a detection would overlap and include parts of other animals in the bounding box of the desired individual.

The experiments with SWIFT show that our approach is an efficient and effective solution for the MOTS task for wildlife camera trap videos better than other state-of-the-art approaches. The instance segmentation part of SWIFT is very flexible. That means that the Mask R-CNN can in principle be replaced by another instance segmentation method if another baseline model should become established in the community in the future. Moreover, the refinement network can also be replaced by other refinement approaches. This allows SWIFT to benefit from improvements in pre-trained foundation models that have not been specifically trained for the wildlife context. A drawback of the refinement process is that SWIFT is not suitable for real-time analysis of video data. But in the context of wildlife monitoring this does not play a major role. If camera trap data is analysed manually it can take up to months to sift through all recorded videos from one site. The flexibility of the instance segmentation part of SWIFT is also beneficial for the tracking part. We have shown in our experiments that improved instance masks boost the tracking accuracy in comparison to non-optimized detections. The SWIFT approach relies on the tracking-by-detection paradigma, which means that the tracking accuracy is dependent on the quality of the detections. Therefore, an improved instance segmentation will also result in better tracking accuracies.

In general the use of artificial intelligence approaches like SWIFT to automatically process video data in wildlife monitoring is very beneficial for ecologists. The enormous amount of video data that camera traps generate cannot be analysed manually. An automatic processing of video data enables standardised results and saves time for further tasks. However, to train SWIFT on, for example, videos for a new species, training data must first be created and annotated. All Deep Learning approaches require a large amount of diverse training data to make correct predictions on unseen data. We have shown in our work [59] that the annotation process can be partially automated. Especially if there are already annotated videos or similar dataset, for example, SWIFT can be used to generate instance masks and tracks as annotations for new videos like we proposed in general in [59].

6 Conclusion

In this work, we emphasize why instance segmentation and tracking are important in the context of wildlife monitoring. In general, instance segmentation and tracking are areas in wildlife monitoring that have hardly been explored so far. In our past three publications [58], [59] and [60] we explored these topics. With SWIFT [60] we presented our efficient and effective MOTS approach, which is the first approach, to the best of our knowledge, that tracks animals in wildlife monitoring videos with instance masks. We analyze the functionality of SWIFT on two camera trap datasets. In our experiments we have shown that SWIFT improves quality of the instance masks as well as the tracking accuracy compared to other state-of-the-art approaches.

Instance segmentation and tracking are beneficial for tasks like abundance estimation, quantification of species diversity, detection and study of rare species and the analysis of species replacement processes. Moreover, the results of the instance segmentation form an important basis for more complex tasks like action detection or re-identification, which are helpful for an behavioural studies.

We believe that the use of SWIFT in wildlife monitoring will be beneficial for ecologists by eliminating the need to analyze all video data material by camera traps themselves, while enabling new insights through application to large datasets. In the future, we plan to use SWIFT as our basis for a further action detection or re-identification of animals in videos.

7 Acknowledgements

This work was partially done within the project ”Automated Multisensor station for Monitoring Of species Diversity” (AMMOD) which is funded by the German Federal Ministry of Education and Research (Bundesministerium für Bildung und Forschung (BMBF), Bonn, Germany (FKZ 01LC1903B). We would like to thank the members of our working group, Timm Haucke and Florian Huber, for fruitful discussions on aspects of this study.

Table 3 MOTS comparison: The MOTS metrics for SWIFT with the Mask R-CNN detections and with the refined detections are shown. The best results are shown in bold.

Dataset	Method	sMOTSA ↑	MOTSA ↑	MOTSP ↑	FP ↓	FN↓	IDS↓
Rolandseck Daylight	**SWIFT** (with Mask R-CNN detections)	31.5%	53.7%	71.5%	5601	5182	**69**
	SWIFT (with refined detections)	**46.0%**	**62.5%**	**79.8%**	**4375**	**4348**	77
Wildlife Crossings	**SWIFT** (with Mask R-CNN detections)	54.9%	73.2%	77.0%	38	**116**	12
	SWIFT (with refined detections)	**61.4%**	**76.3%**	**81.6%**	**20**	122	**8**

8 References

1 Martin Ahrnbom, Mikael G Nilsson, and Håkan Ardö. Real-time and online segmentation multi-target tracking with track revival re-identification. In *VISIGRAPP (5: VISAPP)*, pages 777–784, 2021.

2 Ali Athar, Sabarinath Mahadevan, Aljosa Osep, Laura Leal-Taixé, and Bastian Leibe. Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In *European Conference on Computer Vision*, pages 158–177. Springer, 2020.

3 Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473, 2018.

4 Sara Beery, Dan Morris, and Siyu Yang. Efficient pipeline for camera trap image review. *arXiv preprint arXiv:1907.06772*, 2019.

5 Rotimi-Williams Bello, Ahmad Sufril Azlan Mohamed, and Abdullah Zawawi Talib. Contour extraction of individual cattle from an image using enhanced mask r-cnn instance segmentation method. *IEEE Access*, 9:56984–57000, 2021.

6 Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 941–951, 2019.

7 Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008.

8 Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9739–9748, 2020.

9 Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016.

10 Luigi Boitani. *Camera trapping for wildlife research*. Pelagic Publishing Ltd, 2016.

11 Mathieu Bonneau, Jehan-Antoine Vayssade, Willy Troupe, and Rémy Arquet. Outdoor animal tracking combining neural network and time-lapse cameras. *Computers and Electronics in Agriculture*, 168:105150, 2020.

12 A Cole Burton, Eric Neilson, Dario Moreira, Andrew Ladle, Robin Steenweg, Jason T Fisher, Erin Bayne, and Stan Boutin. Wildlife camera trapping: a review and recommendations for linking surveys to ecological processes. *Journal of Applied Ecology*, 52(3):675–685, 2015.

13 Jintong Cai and Yujie Li. Realtime single-stage instance segmentation network based on anchors. *Computers and Electrical Engineering*, 95:107464, 2021.

14 Jiale Cao, Rao Muhammad Anwer, Hisham Cholakkal, Fahad Shahbaz Khan, Yanwei Pang, and Ling Shao. Sipmask: Spatial information preservation for fast image and video instance segmentation. *arXiv preprint arXiv:2007.14772*, 2020.

15 Anthony Caravaggi, Marco Zaccaroni, Francesco Riga, Stéphanie C Schai-Braun, Jaimie TA Dick, W Ian Montgomery, and Neil Reid. An invasive-native mammalian species replacement process captured by camera trap survey random encounter models. *Remote Sensing in Ecology and Conservation*, 2(1):45–58, 2016.

16 Anthony Caravaggi, Peter B Banks, A Cole Burton, Caroline MV Finlay, Peter M Haswell, Matt W Hayward, Marcus J Rowcliffe, and Mike D Wood. A review of camera trapping for conservation behaviour research. *Remote Sensing in Ecology and Conservation*, 3(3):109–122, 2017.

17 Guobin Chen, Tony X Han, Zhihai He, Roland Kays, and Tavis Forrester. Deep convolutional neural network based species recognition for wild animal monitoring. In *2014 IEEE international conference on image processing (ICIP)*, pages 858–862. IEEE, 2014.

18 Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019.

19 Longtao Chen, Xiaojiang Peng, and Mingwu Ren. Recurrent metric networks and batch multiple hypothesis for multi-object tracking. *IEEE Access*, 7:3093–3105, 2018.

20 Ruilong Chen, Ruth Little, Lyudmila Mihaylova, Richard Delahay, and Ruth Cox. Wildlife surveillance using deep learning methods. *Ecology and Evolution*, 9(17): 9453–9466, 2019. doi: 10.1002/ece3.5410. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/ece3.5410`.

21 Xinlei Chen, Ross Girshick, Kaiming He, and Piotr Dollar. Tensormask: A foundation for dense object segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

22 Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. *Advances in Neural Information Processing Systems*, 34, 2021.

23 Peng Chu, Jiang Wang, Quanzeng You, Haibin Ling, and Zicheng Liu. Transmot: Spatial-temporal graph transformer for multiple object tracking. *arXiv preprint arXiv:2104.00194*, 2021.

24 Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, 2020.

25 Abhishek Dutta and Andrew Zisserman. The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6889-6/19/10. doi: 10.1145/3343031.3350535. URL `https://doi.org/10.1145/3343031.3350535`.

26 Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. In *Int J Comput Vis*, volume 88, page 303–338, 2010. doi: https://doi.org/10.1007/s11263-009-0275-4.

27 Greg Falzon, Christopher Lawson, Ka-Wai Cheung, Karl Vernes, Guy A Ballard, Peter JS Fleming, Alistair S Glen, Heath Milne, Atalya Mather-Zardain, and Paul D Meek. Classifyme: a field-scouting software for the identification of wildlife in camera trap images. *Animals*, 10(1):58, 2020.

28 Hao-Shu Fang, Jianhua Sun, Runzhong Wang, Minghao Gou, Yong-Lu Li, and Cewu Lu. Instaboost: Boosting instance segmentation via probability map guided copy-pasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 682–691, 2019.

29 Gaspar Faure, Hughes Perreault, Guillaume-Alexandre Bilodeau, and Nicolas Saunier. Polytrack: Tracking with bounding polygons. *arXiv preprint arXiv:2111.01606*, 2021.

30 Tsukasa Fukunaga, Shoko Kubota, Shoji Oda, and Wataru Iwasaki. Grouptracker: video tracking system for multiple animals under severe occlusion. *Computational biology and chemistry*, 57:39–45, 2015.

31 Haiming Gan, Mingqiang Ou, Chengpeng Li, Xiarui Wang, Jingfeng Guo, Axiu Mao, Maria Camila Ceballos, Thomas D Parsons, Kai Liu, and Yueju Xue. Automated detection and analysis of piglet suckling behaviour using high-accuracy amodal instance segmentation. *Computers and Electronics in Agriculture*, 199: 107162, 2022.

32 Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F-radar and signal processing*, volume 140, pages 107–113. IET, 1993.

33 Siân E Green, Jonathan P Rees, Philip A Stephens, Russell A Hill, and Anthony J Giordano. Innovations in camera trapping technology and approaches: The integration of citizen science and artificial intelligence. *Animals*, 10(1):132, 2020.

34 Wenchao Gu, Shuang Bai, and Lingxing Kong. A review on 2d instance segmentation based on deep neural networks. *Image and Vision Computing*, page 104401, 2022.

35 Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

36 Abdul Mueed Hafiz and Ghulam Mohiuddin Bhat. A survey on instance segmentation: state of the art. *International journal of multimedia information retrieval*, 9 (3):171–189, 2020.

37 Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

38 Shun Hongo, Yoshihiro Nakashima, Gota Yajima, and Shun Hongo. A practical guide for estimating animal density using camera traps: Focus on the rest model. 2021.

39 Zhiwei Hu, Hua Yang, and Tiantian Lou. Dual attention-guided feature pyramid network for instance segmentation of group pigs. *Computers and Electronics in Agriculture*, 186:106140, 2021.

40 Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6409–6418, 2019.

41 Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13906–13915, 2020.

42 Rui Li, Baopeng Zhang, Zhu Teng, and Jianping Fan. An end-to-end identity association network based on geometry refinement for multi-object tracking. *Pattern Recognition*, 129:108738, 2022.

43 Wei Li, Yuanjun Xiong, Shuo Yang, Siqi Deng, and Wei Xia. Smot: Single-shot multi object tracking. *arXiv preprint arXiv:2010.16031*, 2020.

44 Chung-Ching Lin, Ying Hung, Rogerio Feris, and Linglin He. Video instance segmentation tracking with a modified vae architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13147–13157, 2020.

45 Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing.

46 Matthew Linkie, Yoan Dinata, Agung Nugroho, and Iding Achmad Haidir. Estimating occupancy of a data deficient mammalian species living in tropical rainforests: sun bears in the kerinci seblat region, sumatra. *Biological Conservation*, 137(1):20–27, 2007.

47 Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *International journal of computer vision*, 128(2):261–318, 2020.

48 Kefeng Lv, Yongsheng Zhang, Ying Yu, Hanyun Wang, Lei Li, Huaigang Jiang, and Chenguang Dai. Contour deformation network for instance segmentation. *Pattern Recognition Letters*, 159:213–219, 2022.

49 Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

50 Mohammad Sadegh Norouzzadeh, Anh Nguyen, Margaret Kosmala, Alexandra Swanson, Meredith S. Palmer, Craig Packer, and Jeff Clune. Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proceedings of the National Academy of Sciences*, 115(25):E5716–E5725, 2018. ISSN 0027-8424. doi: 10.1073/pnas.1719367115. URL `https://www.pnas.org/content/115/25/E5716`.

51 E. Okafor, P. Pawara, F. Karaaba, O. Surinta, V. Codreanu, L. Schomaker, and M. Wiering. Comparative study between deep learning and bag of visual words for wild-animal recognition. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8, Dec 2016. doi: 10.1109/SSCI.2016.7850111.

52 Rafael Padilla, Wesley L Passos, Thadeu LB Dias, Sergio L Netto, and Eduardo AB da Silva. A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics*, 10(3):279, 2021.

53 Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito,

© Copyright resides with the authors

Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

54 Lorenzo Porzi, Markus Hofinger, Idoia Ruiz, Joan Serrat, Samuel Rota Bulo, and Peter Kontschieder. Learning multi-object tracking and segmentation from automatic annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6846–6855, 2020.

55 Jiyang Qi, Yan Gao, Xiaoyu Liu, Yao Hu, Xinggang Wang, Xiang Bai, Philip HS Torr, Serge Belongie, Alan Yuille, and Song Bai. Occluded video instance segmentation. *arXiv preprint arXiv:2102.01558*, 2021.

56 Alvaro Rodriguez, Hanqing Zhang, Jonatan Klaminder, Tomas Brodin, and Magnus Andersson. Toxid: an efficient algorithm to solve occlusions when tracking multiple animals. *Scientific reports*, 7(1):1–8, 2017.

57 Jennifer Salau and Joachim Krieter. Instance segmentation with mask r-cnn applied to loose-housed dairy cows in a multi-camera setting. *Animals*, 10(12):2402, 2020.

58 Frank Schindler and Volker Steinhage. Identification of animals and recognition of their actions in wildlife videos using deep learning techniques. *Ecological Informatics*, 61:101215, 2021.

59 Frank Schindler and Volker Steinhage. Saving costs for video data annotation in wildlife monitoring. *Ecological Informatics*, 65:101418, 2021.

60 Frank Schindler and Volker Steinhage. Instance segmentation and tracking of animals in wildlife videos: Swift-segmentation with filtering of tracklets. *Ecological Informatics*, 71:101794, 2022.

61 Stefan Schneider, Graham W Taylor, Stefan Linquist, and Stefan C Kremer. Past, present and future approaches using computer vision for animal re-identification from camera trap data. *Methods in Ecology and Evolution*, 10(4):461–470, 2019.

62 Bing Shuai, Andrew Berneshawi, Xinyu Li, Davide Modolo, and Joseph Tighe. Siammot: Siamese multi-object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12372–12382, 2021.

63 Konstantin Sofiiuk, Ilia Petrov, Olga Barinova, and Anton Konushin. f-brs: Rethinking backpropagating refinement for interactive segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8623–8632, 2020.

64 Konstantin Sofiiuk, Ilia A Petrov, and Anton Konushin. Reviving iterative training with mask guidance for interactive segmentation. *arXiv preprint arXiv:2102.06583*, 2021.

65 Vivek Hari Sridhar, Dominique G Roche, and Simon Gingins. Tracktor: image-based automated tracking of animal movement and behaviour. *Methods in Ecology and Evolution*, 10(6):815–820, 2019.

66 Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020.

67 Aram Ter-Sarkisov, Robert Ross, John Kelleher, Bernadette Earley, and Michael Keane. Beef cattle instance segmentation using fully convolutional neural network. *arXiv preprint arXiv:1807.01972*, 2018.

68 MW Tobler, SE Carrillo-Percastegui, R Leite Pitman, R Mares, and G Powell. Further notes on the analysis of mammal inventory data collected with camera traps. *Animal Conservation*, 11(3):187–189, 2008.

69 Suzanne TS van Beeck Calkoen, Rebekka Kreikenbohm, Dries PJ Kuijper, and Marco Heurich. Olfactory cues of large carnivores modify red deer behavior and browsing intensity. *Behavioral Ecology*, 32(5):982–992, 2021.

70 Lisette van der Zande, Oleksiy Guzhva, T Bas Rodenburg, et al. Individual detection and tracking of group housed pigs in their home pen using computer vision. *Frontiers in Animal Science*, 2:10, 2021.

71 Gyanendra K. Verma and Pragya Gupta. Wild animal detection using deep convolutional neural network. In Bidyut B. Chaudhuri, Mohan S. Kankanhalli, and Balasubramanian Raman, editors, *Proceedings of 2nd International Conference on Computer Vision & Image Processing*, pages 327–338, Singapore, 2018. Springer Singapore. ISBN 978-981-10-7898-9.

72 Alexander Gomez Villa, Augusto Salazar, and Francisco Vargas. Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks. *Ecological informatics*, 41:24–32, 2017.

73 Petra Villette, Charles J Krebs, and Thomas S Jung. Evaluating camera traps as an alternative to live trapping for estimating the density of snowshoe hares (lepus americanus) and red squirrels (tamiasciurus hudsonicus). *European Journal of Wildlife Research*, 63(1):1–9, 2017.

74 Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7942–7951, 2019.

75 Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020.

76 Shaoru Wang, Yongchao Gong, Junliang Xing, Lichao Huang, Chang Huang, and Weiming Hu. Rdsnet: A new deep architecture forreciprocal object detection and instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12208–12215, 2020.

77 Yang Wang, Wanlin Zhou, Qinwei Lv, and Guangle Yao. Metricmask: Single category instance segmentation by metric learning. *Neurocomputing*, 500:896–908, 2022.

78 Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. *arXiv preprint arXiv:2011.14503*, 2020.

79 Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking. In *European Conference on Computer Vision*, pages 107–122. Springer, 2020.

80 Greg Welch, Gary Bishop, et al. An introduction to the kalman filter. 1995.

81 Marco Willi, Ross T. Pitman, Anabelle W. Cardoso, Christina Locke, Alexandra Swanson, Amy Boyer, Marten Veldthuis, and Lucy Fortson. Identifying animal species in camera trap images using deep learning and citizen science. *Methods in Ecology and Evolution*, 10(1):80–91, 2019. doi: 10.1111/2041-210X.13099. URL `https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13099`.

82 Wenqiang Xu, Haiyang Wang, Fubo Qi, and Cewu Lu. Explicit shape encoding for real-time instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5168–5177, 2019.

83 Yihong Xu, Aljosa Osep, Yutong Ban, Radu Horaud, Laura Leal-Taixé, and Xavier Alameda-Pineda. How to train your deep multi-object tracker. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6787–6796, 2020.

84 Yingkun Xu, Xiaolong Zhou, Shengyong Chen, and Fenfen Li. Deep learning for multiple object tracking: a survey. *IET Computer Vision*, 13(4):355–368, 2019.

85 Zhenbo Xu, Wei Zhang, Xiao Tan, Wei Yang, Huan Huang, Shilei Wen, Errui Ding, and Liusheng Huang. Segment as points for efficient online multi-object tracking and segmentation. In *European Conference on Computer Vision*, pages 264–281. Springer, 2020.

86 Tengfei Xue, Yongliang Qiao, He Kong, Daobilige Su, Shirui Pan, Khalid Rafique, and Salah Sukkarieh. One-shot learning-based animal video segmentation. *IEEE Transactions on Industrial Informatics*, 18:3799–3807, 2021.

87 Fan Yang, Xin Chang, Chenyu Dang, Ziqiang Zheng, Sakriani Sakti, Satoshi Nakamura, and Yang Wu. Remots: Self-supervised refining multi-object tracking and segmentation. *arXiv preprint arXiv:2007.03200*, 2020.

88 Hanqing Yang, Liyang Zheng, Saba Ghorbani Barzegar, Yu Zhang, and Bin Xu. Borderpointsmask: One-stage instance segmentation with boundary points representation. *Neurocomputing*, 467:348–359, 2022.

89 Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5188–5197, 2019.

90 Xinyu Yang, Majid Mirmehdi, and Tilo Burghardt. Great ape detection in challenging jungle camera trap footage via attention-based spatial and temporal feature blending. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

91 Matthias Zeppelzauer. Automated detection of elephants in wildlife video. *EURASIP journal on image and video processing*, 2013(1):1–23, 2013.

92 Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. A simple baseline for multi-object tracking. *arXiv preprint arXiv:2004.01888*, 3(4):6, 2020.

93 Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. *arXiv preprint arXiv:2110.06864*, 2021.

94 Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129(11):3069–3087, 2021.

© Copyright resides with the authors