Visual Tracking of Small Animals in Cluttered Natural Environments Using a Freely Moving Camera

Benjamin Risse
School of Informatics
University of Edinburgh, UK
brisse@inf.ed.ac.uk

Michael Mangan
Lincoln Centre for Autonomous Systems
University of Lincoln, UK
mmangan@lincoln.ac.uk

Luca Del Pero
School of Informatics
University of Edinburgh, UK
prusso83@gmail.com

Barbara Webb
School of Informatics
University of Edinburgh, UK
bwebb@inf.ed.ac.uk

Abstract

Image-based tracking of animals in their natural habitats can provide rich behavioural data, but is very challenging due to complex and dynamic background and target appearances. We present an effective method to recover the positions of terrestrial animals in cluttered environments from video sequences filmed using a freely moving monocular camera. The method uses residual motion cues to detect the targets and is thus robust to different lighting conditions and requires no a-priori appearance model of the animal or environment. The detection is globally optimised based on an inference problem formulation using factor graphs. This handles ambiguities such as occlusions and intersections and provides automatic initialisation. Furthermore, this formulation allows a seamless integration of occasional user input for the most difficult situations, so that the effect of a few manual position estimates are smoothly distributed over long sequences. Testing our system against a benchmark dataset featuring small targets in natural scenes, we obtain 96% *accuracy for fully automated tracking. We also demonstrate reliable tracking in a new data set that includes different targets (insects, vertebrates or artificial objects) in a variety of environments (desert, jungle, meadows, urban) using different imaging devices (day / night vision cameras, smart phones) and modalities (stationary, hand-held, drone operated).*

1. Introduction

Knowledge of the precise movement patterns of animals in their natural habitats is important for many fields of study, from neuroscience to conservation. Tracking of animals in the wild is predominantly done through telemetry [14], but this has serious drawbacks: the need to tag animals with sensors limits the application to only 0.3% of all species [15]; and tags can affect the behaviour observed [18]. Also, telemetry does not provide information about an animal's actions, has a limited temporal resolution and provides no information about the surrounding environment [8].

Visual object tracking provides a solution [9] and has already been widely used to extract posture and motion features for different model organisms in laboratory conditions (*e.g.* flies [6, 32]; worms [33]; larvae [10, 25]; fish [21]; or mice [35, 24]). Yet it has proven difficult to extend the application of vision-based tracking to dynamic and complex natural environments [8] especially for tiny animals such as insects. As a result quantification of insect paths is still often done manually by human experts (*e.g.* [22, 5]). Challenges of tracking animals in natural settings include: (1) small targets do not provide sufficient visual features to extract a detectable model (caused by a very low per-animal resolution); (2) animals often provide a low foreground / background contrast ratio (caused by camouflage, etc.); (3) freely moving animals change their appearance over time (caused by shadows, locomotion method, etc.); (4) animals frequently navigate in very cluttered and ambiguous settings (caused by occlusions, background object motion, other animals in close proximity, etc.); and (5) creating manually labelled training data for different and potentially tiny animals is complicated and time consuming. In addition, if the route of the animal is not known beforehand, or not confined to a fixed region, the camera has to move freely to provide continuous recordings over long distances.

A comprehensive review of visual animal tracking approaches is given in [8] which also notes the complete absence of a system capable to track animals in natural habitats. Likewise many state-of-the-art techniques used for pedestrian or vehicle tracking might are not appropriate due to the potential absence of distinctive colour and texture features and the highly dynamic scenes [31]. Furthermore, challenge (5) makes deep learning techniques impractical for many biological studies. In contrast to the approach described in [12] our target application is not optimisation of online tracking, but rather to obtain, offline, the most accurate target position estimate over all video frames. Their method also requires a minimum target resolution so would not succeed for some instances of tiny animals (occupying a few pixels) in the data sets we explore here. These represent typical applications (most were obtained directly from biology research groups) where the video sequence may also be a unique (one-off) combination of animal target and background, which we would ideally be able to track without requiring initialisation or any pre-training. These constraints also clearly limit the applicability of appearance-based detection methods [13, 16, 36, 29].

In order to benchmark tracking systems for natural conditions Bagheri *et al*. recently introduced the Small Targets within Natural Scenes (STNS) dataset featuring 25 sequences of small objects moving in front of a heavily cluttered background [1]. The authors also provide a comparison between their own insect-inspired tracking system and 9 state-of-the-art algorithms, finding an overall low accuracy of current approaches, with the highest success rate reported as 52.1% [1].

In this paper we present an optimised visual tracking approach to track any animal in all kinds of wildlife videos. Given an overhead video of an unmarked animal recorded using a moving or static camera, our method outputs continuous and globally optimal 2D locations of the animal. Our main contribution is twofold. Firstly, we make principled use of motion cues to identify even tiny foreground objects in front of the complex background, following methods that have been previously used in different contexts [3].This approach means that we do not require any a-priori appearance model of the animal or the terrain, making our algorithm applicable for all kinds of moving animal and robust to different lighting conditions. Furthermore, our algorithm can cope with a freely moving camera by automatically compensating the camera motion, using methods similar to the feature based camera motion removal approach described in [34]. Secondly, we detect the position of the animal in all frames of a video jointly, using global optimisation over a factor graph. Factor graphs provide a general graphical framework to represent functions as a product of local functions. These probabilistic models are commonly used to provide consistent solutions in ambiguous multi-target tracking approaches [20, 30] or to identify physical parts of an articulated object class [7]. In our application of this method to animal tracking, the main advantage is to exploit the fact that the experimenter is naturally following the target animal with the camera. Global optimisation then automatically handles ambiguities such as occlusions and intersections, enables an implicit initialisation of the most prominent animal, reinitialises after very long occlusions, and allows for an easy integration of user input to deal with any remaining unresolved situations.

2. Methods

2.1. Algorithm overview

An overview of our proposed algorithm is given in Figure 1. A video of the animal is recorded using a moving or stationary camera. The algorithm first determines and removes any camera motion between consecutive frames (Section 2.2) then extracts remaining foreground motion for each frame to build 2D probability distributions of hypothetical animal locations called unary potentials (Section 2.3). These are combined with 2D motion models, called pairwise potentials, to define a factor graph which is used to induce smoothness in animal localizations over all video frames jointly (*i.e.* global optimisation). As illustrated in Figure 1 each, hypothetical animal location is associated with a 2D Gaussian centred at this location resulting in multiple pairwise potentials. The tracking results can be reviewed and manually corrected if necessary (Section 2.4). Corrections are transformed into unary potentials where the variance tends to zero, and incorporated into the global optimisation approach so that a single manual position influences the entire sequence. Software available at http://blog.inf.ed.ac.uk/insectrobotics/habitracks

2.2. Camera motion

In many cases, filming an animal in its natural habitat requires following it with a freely moving camera. Our algorithm starts by estimating the relative camera motion (Figure 1) by matching ORB features [28] in consecutive frames $\mathcal{I}_t, \mathcal{I}_{t+1}$. These matches are subsequently used in a RANSAC approach [26] to find the perspective transformation $\mathcal{H}_{t\leftarrow t+1}$ which warps all points on the image plane of $\mathcal{I}_{t+1}$ to the camera position of frame $\mathcal{I}_t$:

$$\mathcal{I}_t \approx \tilde{\mathcal{I}}_{t+1} = \mathcal{H}_{t\leftarrow t+1} \circ \mathcal{I}_{t+1} \tag{1}$$

where $\circ$ is the mathematical operator warping the pixels (images or points) in homogeneous coordinates using the transformation $\mathcal{H}_{t\leftarrow t+1}$. Note that $\mathcal{H}_{t\leftarrow t+1}$ is a non-singular and bijective matrix so that it can be inverted ($\mathcal{H}^{-1}_{t\rightarrow t+1}$) to warp frame $\mathcal{I}_t$ on frame $\mathcal{I}_{t+1}$ and that warping over a distance of k consecutive frames can be done by

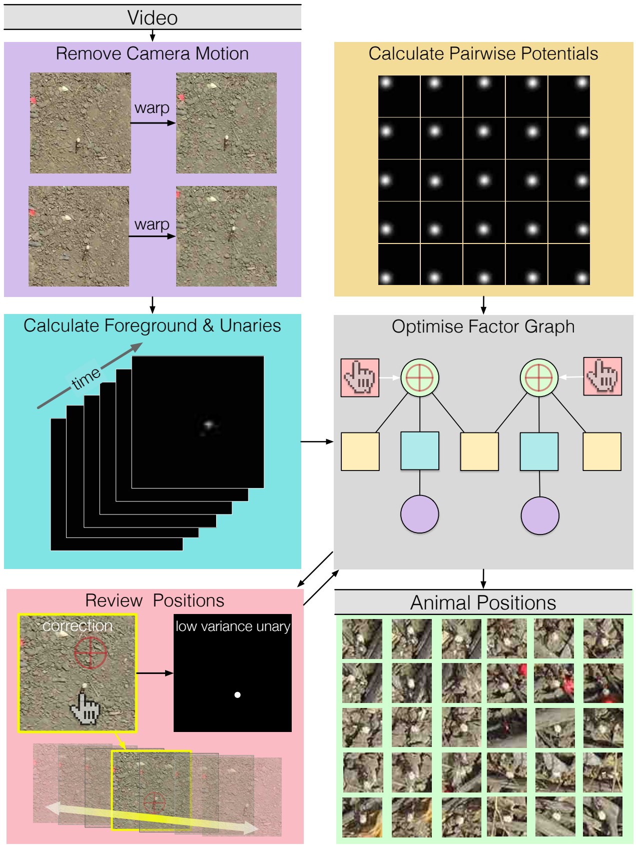

Figure 1. Overview of our tracking pipeline illustrating the processing steps from video input: camera motion removal (purple); unary (blue) and pairwise (yellow) potential calculation; optional sparse corrections (red); combined (same colour code) for factor graph optimisation to produce estimated animal locations.

using

$$\mathcal{H}_{t \leftarrow t+k} = \prod_{i=1}^{k-1} \mathcal{H}_{t \leftarrow t+i}. \quad (2)$$

Perspective transformations with these properties are also called homographies and can be used to virtually warp frames onto future and past relative camera positions.

2.3. Optimal tracking

We formulate animal detection as a probabilistic inference problem to estimate the animal positions $p_t = (x_t, y_t)$ for all frames $t \in \{1, ..., T\}$ with highest probability across the entire video. Each random variable p_t can take N states where N is the spatial resolution of the video frames ($N = \text{width} \times \text{height}$). If very high resolution recordings are used, this state space N can be reduced by a user-specified sub-sampling value (in $[0, 1]$), which speeds up computation by lowering the dimensionality. Given $\mathbf{p} = \{p_1, p_2, ..., p_T\}$ and $\mathbf{D} = \{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_T\}$ we model this problem as maximising an energy function associated with a factor graph

$$E(\mathbf{p} \mid \mathbf{D}) = \left(\sum_{i=1}^{T} \Phi(p_i | \mathcal{D}_i) \right) + \left(\sum_{i=1}^{T-1} \Psi(p_i, p_{i+1}) \right) \quad (3)$$

$\mathcal{D}_t$ specifies the observed variable extracted from the frames $\mathcal{I}_t$ and encodes the animal's motion between consecutive frames. Φ is the unary potential measuring the conditional probability of the animal's position p_t given the observation $\mathcal{D}_t$. Φ will encourage positioning the animal where there is observed motion. Ψ is a pairwise potential encouraging smoothness in animal motion between consecutive frames t and $t + 1$.

In order to compute the observed variables from the images we first warp the image at $t + 1$ onto the camera position of frame t (*c.f.* Equation 1). The remaining motion between the warped frame $\tilde{\mathcal{I}}_{t+1}$ and $\mathcal{I}_t$ should be the moving animal of interest, hence we define $\mathcal{D}_t$ as $|\mathcal{I}_t - \tilde{\mathcal{I}}_{t+1}|$. However note this will also include remaining background motion such as shadows, moving plants and other nearby animals. If the motion of the animal is much slower than the video frame rate this difference image can be generated for $t + k$ distant frames ($k \geq 1$) by warping using the transformation given in Equation 2. From a visual point of view $\mathcal{D}_t$ is a heat map with high values indicating motion at this position. This motion-based approach means that tracking does not rely on animal appearances, no marking is required, and all kinds of imaging sensors can be used (*e.g.* day / night vision, thermographic camera), but it does require that the animal is moving in the majority of frames (approximately $> 50\%$). Under this assumption, the heat maps $\mathcal{D}_t$ can be interpreted as two-dimensional probability distributions where high values correspond to a high probability of the animal's position.

The unary potential is then defined as

$$\Phi(p_t \mid \mathcal{D}_t) = \mathcal{D}_t \cdot \mathcal{N}(\mu_c, \sigma_U^2)|_{p_t} \quad (4)$$

The observed variable $\mathcal{D}_t$ is weighted by a two-dimensional Gaussian $\mathcal{N}$ centred at the image ($\mu_c = \left(\frac{\text{height}}{2}, \frac{\text{width}}{2}\right)$) using a user specified variance σ_U^2 and evaluated at p_t. Weighting biases the maps based on the assumption that the experimenter naturally tries to keep the animal in the centre of the image, in the case of a moving camera, and also avoids artefacts at the image boundaries caused by the warping. Since this assumption cannot be made for stationary recordings the observed variables are simplified to $\widehat{\mathcal{D}}_t = |\mathcal{I}_t - \mathcal{I}_{t+1}|$ and the unary potentials are defined as $\widehat{\Phi}(p_t \mid \widehat{\mathcal{D}}_t) = \widehat{\mathcal{D}}_t|_{p_t}$ for fixed camera videos.

Pairwise potentials ensure smooth animal motions p_t and p_{t+1} by

$$\Psi(p_t, p_{t+1}) = \mathcal{N}(p_{t+1} \mid p_t, \sigma_P^2) \quad (5)$$

σ_P^2 controls for the maximal velocity of the animal in consecutive frames given its resolution in the frame and the

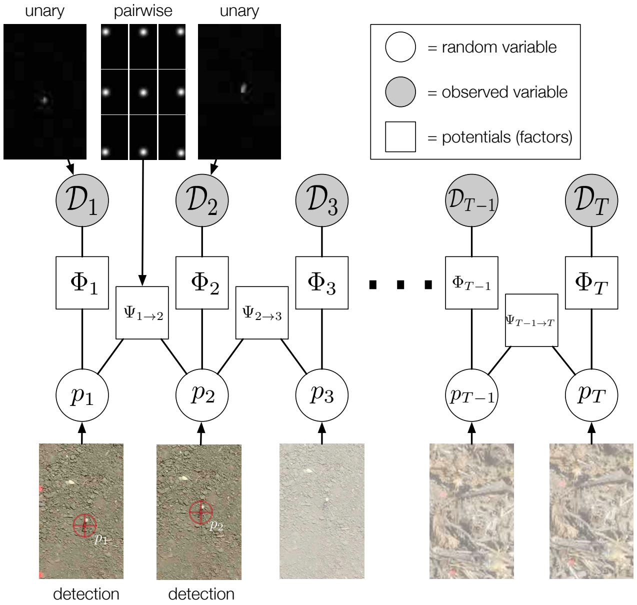

Figure 2. Global optimisation using a factor graph. The formal notation given in the text is linked to the individual variables and factors of the graph. The overall task is to identify the best possible sequence of random variables which maximise the energy of all potentials. Note, the 2D Gaussian which specifies Ψ could be centred anywhere in the image (nine examples are illustrated).

used frame-rate. Given the state space N for all p_t, p_{t+1} each pairwise potential has the dimensionality of N^2. Since σ_P^2 is usually small, Ψ is mostly zero, thus the computational expense can be greatly reduced by modelling Ψ with a truncated version [11]. Note that this motion model only assumes smooth motion in terms of proximity but does not restrict the motion to any direction so that it is general enough to apply for all kinds of behaviour (*e.g.* forward / backwards movement, erratic direction changes).

By combining the unary and pairwise potentials in factor graphs the goal is to identify the animal positions $p_t = (x_t, y_t)$ for all frames $t \in [1, T]$ (*c.f.* Figure 2) [7]. The overall energy of this optimisation task is then

$$\arg \max_{p_1,\ldots,p_T} E(\mathbf{p} \mid \mathbf{D}) \tag{6}$$

which can be solved globally by using the max-sum algorithm [2] that guarantees finding the global optimum of the energy.

This approach proves to be highly effective as the global optimisation automatically disambiguates many difficult tracking situations. For example, intersecting paths with other animals, or uncertainty about which animal is being followed in the initial frames, are usually resolved by the fact that the target animal is the only one present in all frames, as the experimenter follows it throughout the sequence. Similarly, most short pauses or temporary occlusions of the animal are handled correctly and the algorithm initialises automatically by identifying the most prominent moving object. Note that the output of our detection method is a single point, the estimated x,y position of the animal in the video frame.

2.4. Sparse corrections

Some situations such as long occlusions can result in invalid unary potentials Φ since the corresponding observed variables $\mathcal{D}_j$ will not represent the actual motion and location of the animal. To deal with this, we include a mechanism for sparse manual corrections to be incorporated in the global optimisation framework. For any single frame, the user can enter the animal's location p_m^*, and this is used to generate a unary potential Φ^* with highest probability at p_m^* and a zero probability everywhere else (Figure 1). This is automatically incorporated into the factor graph given by Equation 6, such that the influence of this single correction spreads over the entire sequence and thus affects the global result. In other words, a very small number of manually entered positions during ambiguous situations can fix hundreds of frames by seamlessly integrating the user input into the global detection pipeline.

2.5. Parameter settings

Our tracking approach works on videos recorded from uncalibrated cameras and only 3 straightforward parameters need to be set. The first parameter specifies the maximal motion of the camera relative to the used frame rate which is specified by the standard deviation of a 2D Gaussian σ_U^2 (used to define the unary potentials; Equation 4). In a similar fashion the maximum displacement of the animal has to be specified by σ_P^2 (with $\sigma_P^2 \ll \sigma_U^2$ (used to define the pairwise potentials, Equation 5). Both are given in pixels. Finally, in order to increase processing speed the observed variables $\mathcal{D}_t$ and thus dimensionality of the random variables p_t can be down-sampled by a constant factor. We used a down-sampling factor of 50% in all tests. In theory, these three parameters could also be estimated automatically from the video: The homographies comprise the camera translations between consecutive frames which could be used in combination with the resolution and frame rate to estimate σ_U^2 and since the experimenter is following the animal, this camera motion also indirectly reflects the speed of the animal and thus σ_P^2. Since the the detection accuracy is not very sensitive to these parameters we used the same σ_P^2 in all scenarios and only adjusted σ_U^2 according to the spatial resolution.

2.6. Evaluation datasets

Our tracking approach is evaluated based on the publicly available Small Targets within Natural Scenes (STNS) dataset and using an additional new dataset. The STNS dataset consists of 25 video sequences including heavy clutter and camera motion and is particularly dedicated to benchmark small target tracking systems in natural environments [1]. The smallest target length in the STNS dataset is ~ 33 pixels, and targets are recorded in front of natural environments. However the data set does not include realistic animal tracking scenarios *i.e.* overhead videos of terrestrial animals moving through their natural habitat. Therefore we have collected and manually annotated an additional dataset. The Wildlife Animal Tracking (WAT) dataset is a collection of 12 videos featuring various animals (ant, dung beetle, coyote, woodlouse, penguin, artificial object) and environments (desert, jungle, steppe, rocks, meadow, laboratory, urban). The set also covers a variety of imaging devices (camcorder, action camera, smart phone camera) and modalities (day / night vision, stationary / hendheld / drone operated). The scenarios include all kinds of challenges such as moving shadows, lens flare, clutter, direct occlusion of the animal, camouflage, changes in overall brightness and animal appearance, erratic animal motions and neighbouring animals. The video resolutions range from 640×480 to 3840×2160 pixels and the animal length ranges from less than 5 pixels up to 196 pixels (*c.f.* Figure 3). WAT database (videos and manual ground truth annotations) available at (http://blog.inf.ed.ac.uk/insectrobotics/WAT).

2.7. Evaluation metrics

The ground truth target position, based on hand annotation, is specified by its centre of mass in each frame p_t^G as well as the length of the object L. These measures are used to define a bounding circle centred on the target. The tracking accuracy is calculated using the Euclidean distance between the ground truth animal centres $\{p_1^G, ..., p_T^G\}$ and the calculated animal centres returned by our algorithm $\{p_1, ..., p_T\}$: $d(p_t, p_t^G) = ||p_t - p_t^G||$. Since both datasets include sequences with different video and target resolutions these distance measures are normalised by the bounding circle diameter (*i.e.* animal length) L to generate the so-called normalised centre errorn(NCE) which can be used to cross-compare all scenarios [4]:

$$NCE(p_t, p_t^G) = \frac{d(p_t, p_t^G)}{L} \tag{7}$$

3. Results

3.1. STNS dataset tracking accuracy

The authors of the STNS dataset benchmarked 10 state-of-the-art tracking approaches for their success rate, defined as the percentage of frames in which the calculated object position is within the ground truth bounding box [1]. The success rate for these algorithms, averaged over all 25 scenarios, varied from 14.2% to 52.1%. For comparison, we evaluated our tracking approach using the same data and error metric, obtaining a success rate of 96.5%. It should however be noted that the algorithms evaluated in [1] perform successive frame-to-frame tracking, which potentially enables real-time tracking. In contrast, our algorithm operates on all images in a global optimisation scheme so is inherently a post-processing method with processing time depending on the overall sequence. The time required was equivalent to 0.19 seconds per frame on the shortest and 0.37 seconds per frame on the longest sequence in the STNS dataset (measurements on a standard laptop with a 3.1GHz Intel Core i7 CPU and 16GB DDR3 memory without parallelisation).

We show the performance of our algorithm for each of the 25 individual scenarios quantified by the NCE for each frame (*c.f.* Equation 7) in Figure 4. The error is normalised by the length of the target so that deviations below 0.5 animal lengths from the centre of mass equate to successful matches (detections that are located on the object; *c.f.* Figure 4 top right). The average error is below 0.36 animal lengths and the average range specified by the first and third quantile is $[0.27, 0.52]$. As shown in Figure 4 the majority of detections are within the 0.5 animal length boundary. Only video 9 has a median NCE of 0.52 (slightly above 0.5), which is caused by a long occlusion of the object. For some scenarios there is a maximum NCE of 10 or more, and many outliers (+-symbols), but this is largely due to the target leaving the field of view of the camera.

3.2. WAT dataset tracking accuracy

In contrast to the STNS dataset the target is always in the field of view of the camera in the WAT dataset (yet sometimes occluded). Results of our tracking algorithm on the WAT dataset are given in Figure 5. The median error never exceeds 0.5 indicating the estimated position is within the animal's boundary. The worst median score (0.48) was obtained in the *Camouflage* scenario, where the background contrast ratio is very low and the animal tiny (36 pixels within a 2MP image; *c.f.* Figure 3). As a consequence the small appearance of the almost invisible object causes our algorithm to track the ants' shadow or nearby compression noise. The median distance between the centre of mass and detection is 17 pixels (animal length $\times$ median NCE $= 36 \times 0.48$) which is still in the range of manual annotation variability especially since the camouflaged ant is almost invisible.

The most outliers occurred in the *Occluded* and *Woodlouse* scenario (Figure 5). In the former video the animal was under foliage in more than 14% of all frames making determination of location already ambiguous in the manual annotation. Since our algorithm incorporates all frames in

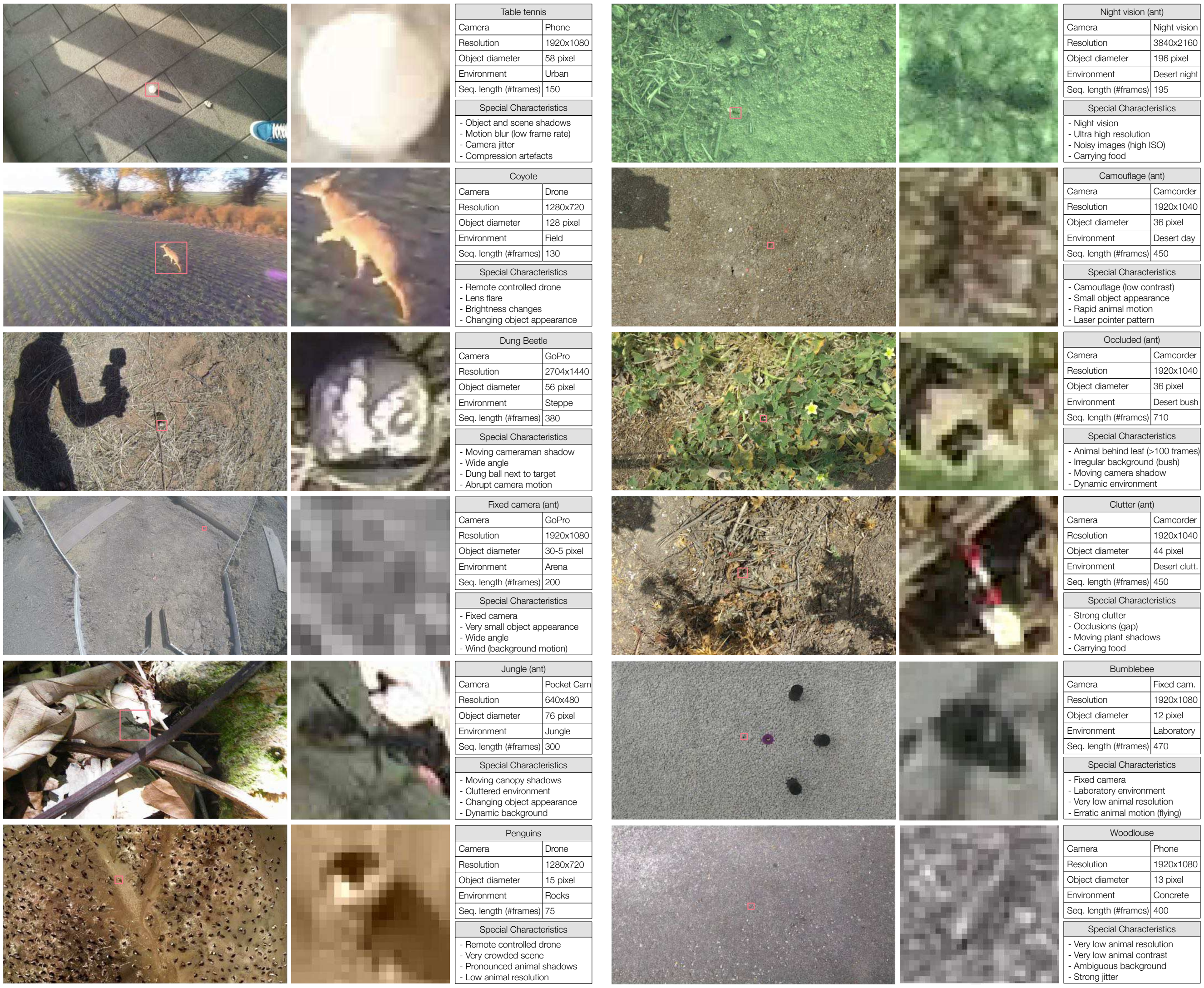

Table tennis	
Camera	Phone
Resolution	1920x1080
Object diameter	58 pixel
Environment	Urban
Seq. length (#frames)	150
Special Characteristics	
- Object and scene shadows - Motion blur (low frame rate) - Camera jitter - Compression artefacts	

Coyote	
Camera	Drone
Resolution	1280x720
Object diameter	128 pixel
Environment	Field
Seq. length (#frames)	130
Special Characteristics	
- Remote controlled drone - Lens flare - Brightness changes - Changing object appearance	

Dung Beetle	
Camera	GoPro
Resolution	2704x1440
Object diameter	56 pixel
Environment	Steppe
Seq. length (#frames)	380
Special Characteristics	
- Moving cameraman shadow - Wide angle - Dung ball next to target - Abrupt camera motion	

Fixed camera (ant)	
Camera	GoPro
Resolution	1920x1080
Object diameter	30-5 pixel
Environment	Arena
Seq. length (#frames)	200
Special Characteristics	
- Fixed camera - Very small object appearance - Wide angle - Wind (background motion)	

Jungle (ant)	
Camera	Pocket Cam
Resolution	640x480
Object diameter	76 pixel
Environment	Jungle
Seq. length (#frames)	300
Special Characteristics	
- Moving canopy shadows - Cluttered environment - Changing object appearance - Dynamic background	

Penguins	
Camera	Drone
Resolution	1280x720
Object diameter	15 pixel
Environment	Rocks
Seq. length (#frames)	75
Special Characteristics	
- Remote controlled drone - Very crowded scene - Pronounced animal shadows - Low animal resolution	

Night vision (ant)	
Camera	Night vision
Resolution	3840x2160
Object diameter	196 pixel
Environment	Desert night
Seq. length (#frames)	195
Special Characteristics	
- Night vision - Ultra high resolution - Noisy images (high ISO) - Carrying food	

Camouflage (ant)	
Camera	Camcorder
Resolution	1920x1040
Object diameter	36 pixel
Environment	Desert day
Seq. length (#frames)	450
Special Characteristics	
- Camouflage (low contrast) - Small object appearance - Rapid animal motion - Laser pointer pattern	

Occluded (ant)	
Camera	Camcorder
Resolution	1920x1040
Object diameter	36 pixel
Environment	Desert bush
Seq. length (#frames)	710
Special Characteristics	
- Animal behind leaf (>100 frames) - Irregular background (bush) - Moving camera shadow - Dynamic environment	

Clutter (ant)	
Camera	Camcorder
Resolution	1920x1040
Object diameter	44 pixel
Environment	Desert clutt.
Seq. length (#frames)	450
Special Characteristics	
- Strong clutter - Occlusions (gap) - Moving plant shadows - Carrying food	

Bumblebee	
Camera	Fixed cam.
Resolution	1920x1080
Object diameter	12 pixel
Environment	Laboratory
Seq. length (#frames)	470
Special Characteristics	
- Fixed camera - Laboratory environment - Very low animal resolution - Erratic animal motion (flying)	

Woodlouse	
Camera	Phone
Resolution	1920x1080
Object diameter	13 pixel
Environment	Concrete
Seq. length (#frames)	400
Special Characteristics	
- Very low animal resolution - Very low animal contrast - Ambiguous background - Strong jitter	

Figure 3. WAT dataset overview. Our dataset comprises 12 video sequences showing different targets (artificial, insects and vertebrates) in different environments (urban, desert, jungle, etc.). Length of each sequence is given in frames. For each video an image of the overall scene, a close-up of the object of interest and a table showing the key characteristics is given (from left to right).

which the ant is visible between the foliage and enforces smooth transition (*i.e.* linearly interpolates between valid detections) the resultant trajectory is a very plausible approximation to the hidden real path (Figure 3). The *Woodlouse* scenario suffers from the same contrast issues as the *Camouflage* scenario: the tiny animal (13 pixel) is almost indistinguishable from the background (*c.f*. Figure 3). Furthermore, abrupt camera motion and slow automatic focussing (recorded using a mobile phone) causing strong blur so that 57 frames are not tracked correctly (NCE > 0.5).

In all other scenarios the average NCE is below 0.26 animal lengths and the maximally observed deviation from the ground truth is below 2.8 animal lengths (again caused by occlusions). The global optimisation automatically disambiguates many difficult tracking situations. For example, intersecting paths with other animals, or uncertainty about which animal is being followed in the initial frames (*Penguin* scenario), are usually resolved by the fact that the target animal is the only one present in all frames, as the experimenter follows it throughout the sequence. Similarly, most short pauses or temporary occlusions of the animal are corrected for automatically.

3.3. Sparse correction evaluation

For particularly difficult video sequences, our automatic tracking procedure can be straightforwardly enhanced by allowing the user to indicate the animal's location manually in one or more frames (*c.f*. Figure 1) as high confidence unary

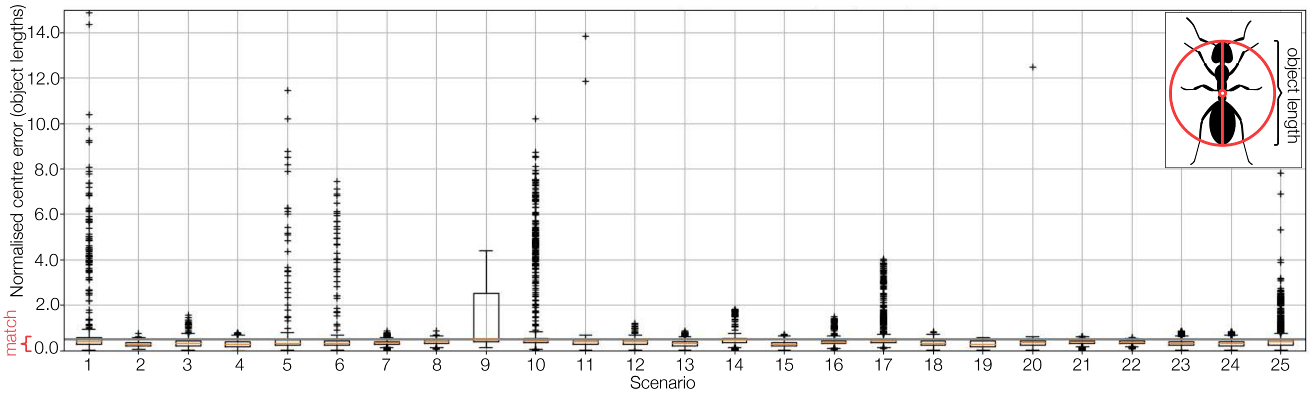

Figure 4. Distribution of normalised centre error (NCE) [4] for the 25 scenarios in the STNS dataset [1]: red bar is median and box shows quartiles. A value below 0.5 means the detection falls within the diameter of a bounding circle of the ground truth center position (top-left inlay).

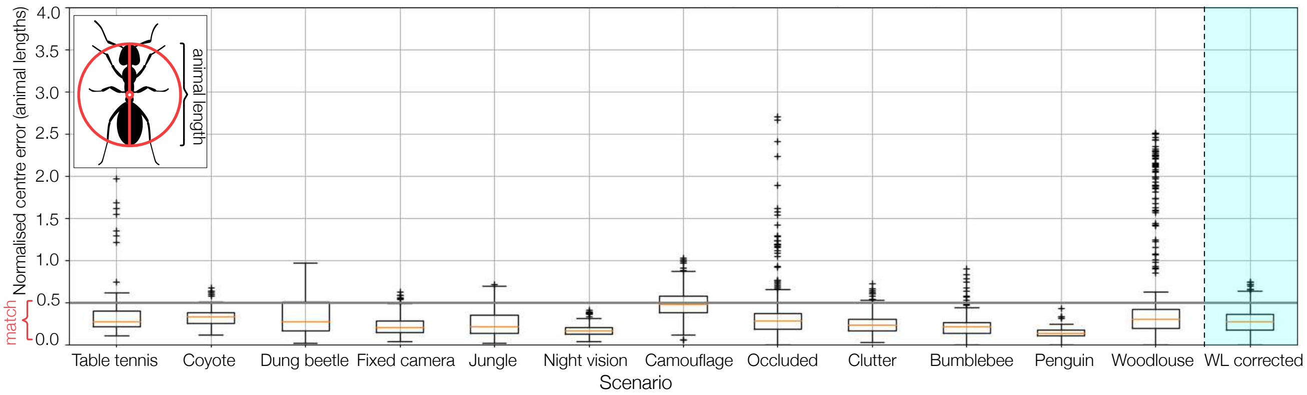

Figure 5. Distribution of normalised centre error (NCE) [4] using our new WAT dataset: as for Figure 4. Plot highlighted in blue indicates the NCE in the *Woodlouse* (WL) scenario after manual corrections.

potentials. The optimisation procedure automatically propagates these to neighbouring frames, such that a few manual inputs may be sufficient to correct hundreds of frames. To demonstrate the effect, we generated a version of the (already difficult) *Clutter* scenario in which we replaced 100 consecutive observed variables $\mathcal{D}_t$ by random salt and pepper noise images (with intensity values in $[0, 100]$). This increased the number of detection mismatches from ground truth (NCE > 0.5) from 3 to 98 for automatic tracking, and produces a median error distance of 2.15 animal lengths. We then incrementally added manual corrections, always dividing the distorted sequence into two equal halves *i.e.* starting with a single correction after 50 frames, etc. As visible in Figure 6 (red curves) the first correction has the strongest impact on the median NCE, already reducing the mismatch distances to ~ 1 animal length. After only 5 corrections there are few remaining mismatches with the NCE asymptotically approaching $y = 0.5$.

The impact of sparse corrections was additionally tested in a non-manipulated scenario. Figure 5 shows the NCE of the *Woodlouse* scenario before and after manual corrections (plot highlighted in blue) and the incremental changes caused by the corrections are given in Figure 6 (purple curves). After applying 5 manual corrections, there are few remaining outliers and the maximum NCE is 0.75 animal lengths. The median deviations of all outliers decreases from 1.83 to 0.83 animal lengths after the first correction and declines to 0.58 animal lengths after five corrections (with a standard deviation of 0.08). We conclude that our global optimisation strategy in combination with sparse corrections is a powerful tool to efficiently deal with even very ambiguous situations.

4. Discussion

We have demonstrated a system that performs fully automatic visual tracking of individual animals in complex wildlife environments. It works with a single uncalibrated

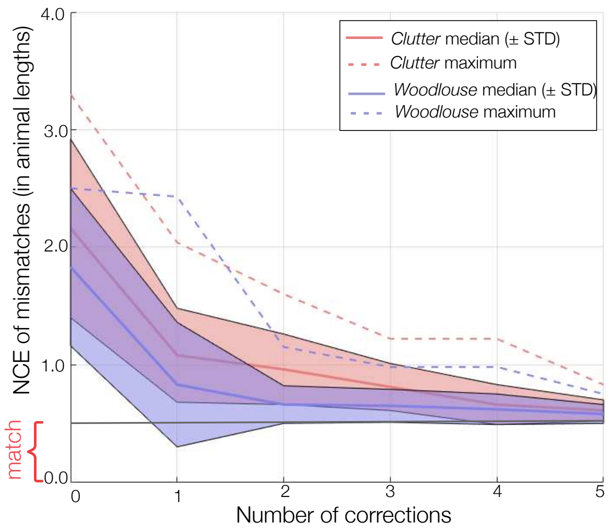

Figure 6. Sparse correction performance evaluation. Red : 100 consecutive images in the *Clutter* scenario were replaced by random noise images to induce mismatches. Blue: the woodlouse scenario contained 57 mismatches (caused by blurred images) which are corrected by 5 manual corrections. The median, standard deviation and maximum of the NCE for all mismatches is plotted against the number of corrections used. After only 5 corrections the error approaches 0.5.

camera, making no prior assumptions about the animal or environment appearance. It can be directly applied to existing video footage of highly variable quality and complexity, requiring only that the animal of interest remains mostly in shot and is moving in the majority of frames. Our algorithm is extremely robust, and sparse user input can be easily integrated to correct for the most difficult situations. Only three straight-forward parameters are necessary and the algorithm does not require a priori information about the animal or environment nor does it require any training. Since we make explicit use of the camera motion to extract the detections the inverse of these transformations (*i.e.* homographies $\mathcal{H}^{-1}_{t \leftarrow t+k}$; Equation 2) can be used to also extract camera motion compensated trajectories up to the limits of a cumulative error (Risse *et al.* in preparation). In the future, we will extend our algorithm towards multi-target tracking based on an iterative version of our probabilistic optimisation.

A limitation in the current implementation is that it represents the animal as a single point. This will be within the animal's boundary, but need not consistently represent the same location on the body, which may introduce trajectory artefacts for larger animals (relative to the video resolution). Additionally, it means no instantaneous orientation information or body shape information is provided. However, future enhancements could use our robust detection method as an initialisation for fitting a tight bounding box around the animal to obtain animal segmentation [27]. Template matching [17] or other techniques for shape description [23] and action classification [19] can then be applied straightforwardly for more detailed analyses tuned to the specific animal and question of interest. Another limitation is that the animal has to move in the majority of the frames in order to be tracked correctly. Even though moderate stops of the animal are compensated due to the global optimisation, very long motion pauses ($> 50\%$) in combination with strong background motion might cause incorrect detections. Sparse corrections are however an effective method to address these situations: the resultant low variance unary eliminates the background noise and highly scores the correct position which is smoothly distributed over the entire sequence as demonstrated using the WAT dataset.

Even though the global optimisation strategy of our algorithm does not allow real-time tracking its processing speed and batch processing capabilities allow to track animals in very long videos in reasonable time. Intermediate results like unary potentials are directly saved to disk and a truncated version of the pairwise potentials is used to prevent memory overflows. Furthermore, optional down-sampling of the observed variables is integrated into our algorithm to further increase the processing speed.

Our algorithm does not rely on any commercial software packages, is implemented in C++, and will be released as open source. It only requires OpenCV and Qt packages and provides a graphical user interface to include sparse corrections and review results. The focus in its development was to enable behavioural researchers to track animals in their natural habitats without specialised recording devices, complex calibration procedures, or expert knowledge. Our method thus has the potential to significantly advance behavioural, ecological and physiological research across many scenarios and model organisms.

Acknowledgments

We thank Antoine Wystrach for support and feedback, and Marie Dacke, Lana Khaldy (Department of Biology; Lund), Natalie Hempel de Ibarra, Theo Robert (CRAB, Exeter), Ajay Narendra (Ecological Neuroscience, Sydney) and Justin Watkins (Kansas, USA) for providing field recordings. This work was supported by EPSRC grant EP/M008479/1.

References

[1] Z. M. Bagheri, S. Wiederman, B. Cazzolato, S. Grainger, and D. O'Carroll. Performance of an insect-inspired target tracker in natural conditions. *Bioinspiration & Biomimetics*, 12(2):025006, Jan. 2017. 2, 5, 7

[2] C. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics), 1st edn. 2006. corr. 2nd printing edn*. Springer, 2007. 4

[3] C. Cedras and M. Shah. Motion-based recognition a survey. *Image and Vision Computing*, 1995. 2

[4] L. Cehovin, A. Leonardis, and M. Kristan. Visual Object Tracking Performance Measures Revisited. *IEEE Trans. Image Processing*, 25(3):1261–1274, 2016. 5, 7

[5] M. Collett, L. Chittka, and T. S. Collett. Spatial memory in insect navigation. *Current biology : CB*, 23(17):R789–800, Sept. 2013. 1

[6] H. Dankert, L. Wang, E. D. Hoopfer, D. J. Anderson, and P. Perona. Automated monitoring and analysis of social behavior in Drosophila. *Nature methods*, 6(4):297–303, Apr. 2009. 1

[7] L. Del Pero, S. Ricco, R. Sukthankar, and V. Ferrari. Discovering the Physical Parts of an Articulated Object Class from Multiple Videos. *CVPR*, pages 714–723, 2016. 2, 4

[8] A. I. Dell, J. A. Bender, K. Branson, I. D. Couzin, G. G. de Polavieja, L. P. J. J. Noldus, A. Pérez-Escudero, P. Perona, A. D. Straw, M. Wikelski, and U. Brose. Automated image-based tracking and its application in ecology. *Trends in Ecology & Evolution*, 29(7):417–428, July 2014. 1, 2

[9] S. E. R. Egnor and K. Branson. Computational Analysis of Behavior. *Annual Review of Neuroscience*, 39(1):217–236, July 2016. 1

[10] A. Gomez-Marin, N. Partoune, G. J. Stephens, and M. Louis. Automated Tracking of Animal Posture and Movement during Exploration and Sensory Orientation Behaviors. *PloS ONE*, 7(8):e41642, 2012. 1

[11] M. Guillaumin, L. Van Gool, and V. Ferrari. Fast Energy Minimization Using Learned State Filters. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1682–1689. IEEE, May 2013. 4

[12] D. Held, S. Thrun, and S. Savarese. Learning to Track at 100 FPS with Deep Regression Networks. *ECCV*, 2016. 2

[13] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *European Conference on Computer Vision*, pages 788–801. Springer, 2008. 2

[14] R. Kays, M. C. Crofoot, W. Jetz, and M. Wikelski. Terrestrial animal tracking as an eye on life and planet. *Science*, 348(6240), June 2015. 1

[15] W. D. Kissling. Animal telemetry: Follow the insects. *Science*, 349(6248):597–597, Aug. 2015. 1

[16] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE transactions on pattern analysis and machine intelligence*, 30(10):1683–1698, 2008. 2

[17] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. van den Hengel. A Survey of Appearance Models in Visual Object Tracking. *arXiv.org*, Mar. 2013. 8

[18] T. McIntyre. Animal telemetry: Tagging effects. *Science*, 349(6248):596–597, Aug. 2015. 1

[19] M. H. Nguyen, Z.-Z. Lan, and F. De la Torre. Joint segmentation and classification of human actions in video. *CVPR*, 2011. 8

[20] V. P. Panakkal and R. Velmurugan. Effective data association scheme for tracking closely moving targets using factor graphs. *Communications (NCC)*, pages 1–5, 2011. 2

[21] A. Pérez-Escudero, J. Vicente-Page, R. C. Hinz, S. Arganda, and G. G. de Polavieja. idTracker: tracking individuals in a group by automatic identification of unmarked animals. *Nature methods*, 11(7):743–748, July 2014. 1

[22] S. E. Pfeffer and M. Wittlinger. Optic flow odometry operates independently of stride integration in carried ants. *Science*, 353(6304):1155–1157, Sept. 2016. 1

[23] D. Ramanan and D. A. Forsyth. Using Temporal Coherence to Build Models of Animals. *ICCV*, 2003. 8

[24] S. L. Reeves, K. E. Fleming, L. Zhang, and A. Scimemi. M-Track: A New Software for Automated Detection of Grooming Trajectories in Mice. *PLoS computational biology*, 12(9):e1005115–19, Sept. 2016. 1

[25] B. Risse, D. Berh, N. Otto, C. Klämbt, and X. Jiang. FIMTrack: An open source tracking and locomotion analysis software for small animals. *PLoS computational biology*, 13(5):e1005530–15, May 2017. 1

[26] A. J. Robertson. A Set of Greedy Randomized Adaptive Local Search Procedure (GRASP) Implementations for the Multidimensional Assignment Problem. *Computational Optimization and Applications*, 19(2):145–164, 2001. 2

[27] C. Rother, V. Kolmogorov, and A. B. Blake. "GrabCut" - interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004. 8

[28] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *2011 International Conference on Computer Vision*, pages 2564–2571. IEEE, 2011. 2

[29] A. Sadeghian, A. Alahi, and S. Savarese. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. *arXiv preprint arXiv:1701.01909*, 2017. 2

[30] M. Schiegg, P. Hanslovsky, B. X. Kausler, L. Hufnagel, and F. A. Hamprecht. Conservation Tracking. In *2013 IEEE International Conference on Computer Vision (ICCV)*, pages 2928–2935. IEEE, Nov. 2013. 2

[31] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual Tracking - An Experimental Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014. 2

[32] A. D. Straw, K. Branson, T. R. Neumann, and M. H. Dickinson. Multi-camera real-time three-dimensional tracking of multiple flying animals. *Journal of the Royal Society, Interface / the Royal Society*, 8(56):395–409, Mar. 2011. 1

[33] N. A. Swierczek, A. C. Giles, C. H. Rankin, and R. A. Kerr. High-throughput behavioral analysis in C. elegans. *Nature methods*, 8(7):592–598, July 2011. 1

[34] H. Wang and C. Schmid. Action Recognition with Improved Trajectories. pages 3551–3558, 2013. 2

[35] A. Weissbrod, A. Shapiro, G. Vasserman, L. Edry, M. Dayan, A. Yitzhaky, L. Hertzberg, O. Feinerman, and T. Kimchi. Automated long-term tracking and social behavioural phenotyping of animal colonies within a semi-natural environment. *Nature communications*, 4:2018, 2013. 1

[36] Y. Xiang, A. Alahi, and S. Savarese. Learning to track: Online multi-object tracking by decision making. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4705–4713, 2015. 2