

# Movement Ecology

## Prying into the intimate secrets of animal lives; software beyond hardware for comprehensive annotation in 'Daily Diary' tags.

--Manuscript Draft--

<b>Manuscript Number:</b>	
<b>Full Title:</b>	Prying into the intimate secrets of animal lives; software beyond hardware for comprehensive annotation in 'Daily Diary' tags.
<b>Article Type:</b>	Software article
<b>Abstract:</b>	Smart tags attached to freely-roaming animals recording multiple parameters at infra-second rates are becoming commonplace, and are transforming our understanding of the way wild animals operate. However, interpretation of such data is complex and currently limits the ability of biologists to realise the value of their recorded information. This work presents a single program, FRAMEWORK 4, that uses a particular sensor constellation described in the 'Daily Diary' tag (recording tri-axial acceleration, tri-axial magnetic field intensity, pressure and e.g. temperature and light intensity) to determine the 4 key elements considered pivotal within the conception of the tag. These are; animal trajectory, behaviour, energy expenditure and quantification of the environment in which the animal operates. The program takes the original data recorded by the Daily Dairy and transforms it into dead-reckoned movements, template-matched behaviours, dynamic body acceleration-derived energetics and position-linked environmental data before outputting it all into a single file. Biologists are thus left with a single data set where animal actions and environmental conditions can be linked across time and space.
<b>Corresponding Author:</b>	James Stephen Walker, BSc, MRes Swansea University Swansea, UNITED KINGDOM
<b>Corresponding Author Secondary Information:</b>	
<b>Corresponding Author's Institution:</b>	Swansea University
<b>Corresponding Author's Secondary Institution:</b>	
<b>First Author:</b>	James Stephen Walker, BSc, MRes
<b>First Author Secondary Information:</b>	
<b>Order of Authors:</b>	James Stephen Walker, BSc, MRes Mark W Jones Robert S Laramee Mark D. Holton Emily L.C. Shepard Hannah J. Williams Michael D. Scantlebury Nikki J. Marks Elizabeth A. Magowan Iain E. Maguire Ed Grundy Agustina Di Virgilio Rory P. Wilson
<b>Order of Authors Secondary Information:</b>	

1 Prying into the intimate secrets of animal lives; software beyond hardware for  
2 comprehensive annotation in ‘Daily Diary’ tags.

3  
4 James S. Walker<sup>1</sup>, Mark W. Jones<sup>1</sup>, Robert S. Laramée<sup>1</sup>, Mark D. Holton<sup>2</sup>, Emily L. C.  
5 Shepard<sup>3</sup>, Hannah J. Williams<sup>3</sup>, D. Michael Scantlebury<sup>4</sup>, Nikki, J. Marks<sup>4</sup>, Elizabeth A.  
6 Magowan<sup>4</sup>, Iain E. Maguire<sup>4</sup>, Ed Grundy<sup>3</sup>, Agustina Di Virgilio<sup>5</sup>, Rory P. Wilson<sup>3</sup>.

7  
8  
9  
10 <sup>1</sup>Department of Computer Science, College of Science, Swansea University, Singleton Park,  
11 Swansea SA2 8PP, Wales, UK.

12 <sup>2</sup>College of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP, Wales,  
13 UK.

14 <sup>3</sup>Swansea Lab for Animal Movement, Biosciences, College of Science, Swansea University,  
15 Singleton Park, Swansea SA2 8PP, Wales, UK.

16 <sup>4</sup>School of Biological Sciences, Institute for Global Food Security, Queen’s University Belfast,  
17 Belfast BT9 7BL, Northern Ireland, UK

18 <sup>5</sup>Laboratorio Ecotono, INIBIOMA-CONICET, National University of Comahue, Quintral 1250,  
19 (8400) Bariloche, Argentina.  
20  
21  
22  
23

## 24 **Abstract**

25 Smart tags attached to freely-roaming animals recording multiple parameters at infra-second  
26 rates are becoming commonplace, and are transforming our understanding of the way wild  
27 animals operate. However, interpretation of such data is complex and currently limits the  
28 ability of biologists to realise the value of their recorded information. This work presents a  
29 single program, FRAMEWORK 4, that uses a particular sensor constellation described in the  
30 ‘Daily Diary’ tag (recording tri-axial acceleration, tri-axial magnetic field intensity, pressure  
31 and e.g. temperature and light intensity) to determine the 4 key elements considered pivotal  
32 within the conception of the tag. These are; animal trajectory, behaviour, energy expenditure  
33 and quantification of the environment in which the animal operates. The program takes the  
34 original data recorded by the Daily Dairy and transforms it into dead-reckoned movements,  
35 template-matched behaviours, dynamic body acceleration-derived energetics and position-  
36 linked environmental data before outputting it all into a single file. Biologists are thus left  
37 with a single data set where animal actions and environmental conditions can be linked across  
38 time and space.  
39  
40  
41  
42

## 43 **1. Introduction**

44 The development of hardware that can be attached to animals during their everyday life  
45 (Ropert-Coudert and Wilson 2005) has revolutionized our understanding of the biology of  
46 wild animals. Indeed, this approach has allowed researchers to look at everything from the  
47 behaviour of whales chasing prey at depths of over 1 km underwater (Aguilar Soto, Johnson  
48 et al. 2008) to the physiology of geese migrating over the Himalayas (Hawkes, Balachandran  
49 et al. 2011). A common feature facilitating these sorts of projects has been the increase in  
50 numbers and types of sensors used in smart animal tags, as well as increases in the frequency  
51 with which they can be sampled and concomitant increases in memory capacity. Thus, our  
52 ability to answer critical questions in biology appears to have been driven to a large extent by  
53 advances in technology (Ropert-Coudert and Wilson 2005). These advances in methodology  
54 come under two broad areas. One relates to methods that allow tags to be physically attached  
55 to animals for increasing lengths of time (e.g. Wilson, Putz et al. 1997; Hooker, Baird et al.  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65

2001; Rowley and Alford 2007) while minimizing animal detriment (Vandenabeele, Shepard et al. 2012; Vandenabeele, Wilson et al. 2013) while the other relates to the physical production of the complex solid-state units in smart tags (e.g. Miller, Johnson et al. 2004).

The excitement at the potential inherent in sophisticated animal tags has, however, been tempered by a new limiting factor. This is a methodology to deal with the problem of the analysis of the high resolution, multiple channel (and therefore multi-dimensional) data acquired by the tags – in short, software (cf. Jonsen, Myers et al. 2003). To be most useful to the community, software to help in the analysis of smart tag-acquired data needs to be able to deal with large quantities of multiple sensor data and, ideally, should be able to merge different derived analytical outputs together into one output file so that various elements derived from the primary data can be interrelated. Currently, the smart tag community has witnessed a number of software innovations, most of which are concerned with determination of behaviour, i.e. from dive profile data (Halsey, Bost et al. 2007) or, most notably, from tri-axial acceleration data (Watanabe, Izawa et al. 2005; Sakamoto, Sato et al. 2009). Analysis of acceleration data is particularly welcome because inspection of raw acceleration values over time to derive behaviours is not particularly intuitive (Shepard, Wilson et al. 2009). Thus, solutions for this have involved a suite of different approaches including cluster analysis (Sakamoto, Sato et al. 2009), support vector machine classification models (Martiskainen, Järvinen et al. 2009) and artificial neural networks (Nathan, Spiegel et al. 2012).

In 2008, Wilson et al. (2008) put forward a concept for a particular sensor configuration within a tag that they called the ‘Daily Diary’ (DD), where analysis yielded value well beyond the simple mathematical sum of its individual parts (Wilson 2008). Specifically, the suggestion advocated the combination of tri-axial accelerometers, tri-axial magnetometers, and pressure and speed transducers together with environmental sensors such as temperature, light and humidity. The theory was that this constellation of sensors, sampled at infra-second rates, would allow tag users to be able to derive four key elements of animal ecology seamlessly. These are: (1) animal trajectory, and therefore position (Wilson, Liebsch et al. 2007) (2) animal behaviour (Shepard, Wilson et al. 2008) (3) energy expenditure (Wilson, White et al. 2006) and (4) the environmental conditions to which the tag carrier is exposed (Wilson, Grémillet et al. 2002). Although this original work did indicate avenues by which these elements might be achieved, there was no specific suggestion of software that might actually do this. In short, currently, although some software is available to help determine some aspects of that advocated by the DD concept (e.g. Bidder, Campbell et al. 2014), there is nothing that binds the concepts together.

This paper describes the structure and functioning of a new software package (DD-FRAMEWORK 4) that allows DD (or similar) users to calculate all four key elements advocated by the DD and then to bind them together into one single output file so that workers can subsequently geo-reference behaviours, energy expenditures and environmental parameters to gain a more holistic picture of how animals react to and within their environment. Specifically, we introduce FRAMEWORK 4, a user-friendly turnkey solution for the analysis of DD data. Using our system, we can seamlessly obtain animal behaviour, animal trajectory, energy expenditure, and environmental conditions all within one application and export them in one merged data file. Our solution requires no knowledge of the underlying processes utilised in the software, or any advanced mathematical and computing skill sets. Our application has been produced with the end-user in mind, utilising wizards and graphical user interfaces wherever possible. We hope this software will assist in

1 the understanding of wild animal ecology, providing new insights as a result of advanced  
2 computing knowledge.

## 3 **2. FRAMEWORK 4 - software implementation**

4  
5  
6 The software implementation is outlined before specifying each of the software features and  
7 the underlying methods by which they operate.  
8

9  
10 Framework 4 is a desktop application for the Microsoft Windows operating system built  
11 using the programming language C++ with the Qt 4.8 framework library. This software has  
12 been implemented through a collaboration with the Bioscience and Computer Science  
13 departments at Swansea University and is maintained by the first author in this article. We  
14 selected to create a desktop application as it allows us to handle large data files directly.  
15 Alternatively, utilising a web application would allow access from anywhere on any platform  
16 with an active internet connection and a web browser, although this would entail long waiting  
17 times while data sets upload, resulting in a reduced ability to handle larger files. Utilising the  
18 desktop application, we directly communicate with the CPU for efficient data handling, and  
19 the GPU for visualisation purposes.  
20  
21

### 22 **2.1 Loading Data**

23 The software supports two file formats; comma separated values (CSV) and tabular  
24 delaminated formats. These are two of the standard file formats for storing tabular data in text  
25 format and are common outputs from commercial animal monitoring tags. The DD exports its  
26 data in a binary format which gets segmented into multiple files and converted to tabular  
27 delimited format post-deployment. FRAMEWORK 4 loads and operates on the individual  
28 tabulated files.  
29  
30  
31

32  
33 We incorporate an import wizard in the application to import CSV and tabular delimited  
34 files. Here, the user can specify the names and data types of each attribute, which are used as  
35 a reference to them throughout the application.  
36  
37

## 38 **3. Derivation of animal trajectory**

### 39 *3.1 Principles behind dead-reckoning*

40  
41  
42 The way in which animals use the environment is fundamental to understanding their  
43 behavioural ecology (Börger et al. 2008) and, as such, many different systems have been  
44 developed to examine animal movements (see e.g. Wikelski et al. 2007). A relatively recent  
45 addition to the field is dead-reckoning (Wilson et al. 2008), which has particular value in not  
46 being dependent on transmission technology and being able to resolve movement at infra-  
47 second scales. Dead-reckoning operates on the basis that the position of an animal at any time  
48 't' can be derived knowing the position at the animal at a previous time 't-1' and the distance  
49 and heading taken between the two time intervals. Dead-reckoning has received little interest  
50 until now, partially because early systems for dead-reckoning were crude (Ioale et al., 1994;  
51 Wilson and Wilson, 1988; Wilson et al., 1991) but with the development of sensors and  
52 techniques, headings can now be computed to within 1° utilising accelerometer and  
53 magnetometer sensors (Caruso, 2000; Wilson et al., 2008).  
54  
55  
56  
57

58 The earth's magnetic field is constructed of field lines approximating a magnetic dipole. Each  
59 field line originates at a point near the magnetic South Pole and terminates at a point near the  
60  
61  
62  
63  
64  
65

1 magnetic North Pole. The relationship between magnetic north and geographic north is  
2 defined by a declination angle which varies across the earth's surface and with time. This  
3 angle can be obtained from a reference table provided by the National Geophysical Data  
4 Center (<http://www.ngdc.noaa.gov/geomag-web/>). Measuring the strength and direction of  
5 the lines utilising the tri-axial magnetometer can be used to obtain a relative compass  
6 heading.

7  
8 In Framework 4, we introduce a user-friendly wizard for performing dead-reckoning on data  
9 with tri-axial magnetometer and tri-axial accelerometer components (Fig. 1). Dead-reckoning  
10 is subject to cumulative errors and, as the heading and speed are estimates, any systematic  
11 deviations from the actual heading and speed will lead to increasing errors. To assist in  
12 reducing such errors, we incorporate a hybrid GPS and dead-reckoned approach which  
13 utilises the GPS fixes as the actual position and forces the dead-reckoned path to go through  
14 these. This resets any cumulative error but does not factor in any GPS errors.. We now  
15 outline each of the components of the Dead-reckoning wizard which are modular so that the  
16 user can select which analysis steps are required. The steps are; (i) GPS import for  
17 synchronising GPS data with the DD data (ii) alignment correction for the accelerometer and  
18 magnetometer coordinate frames (iii) heading derivation from the magnetometer channels  
19 (iv) VeDBA for obtaining an estimate of the speed (v) constant speed options if necessary  
20 (vi) dead-reckoning to combine the heading and speed information to derive an animal  
21 trajectory and (vii) dead-reckoning correction using GPS points to eliminate drift in the final  
22 dead-reckoned path.  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65

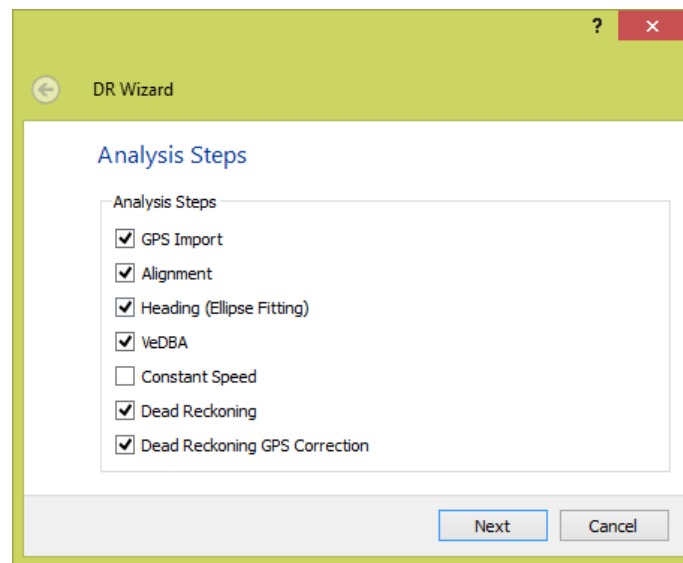


Figure 1 – The Dead-reckoning wizard features a number of modular classification steps. These are; GPS Import, Alignment, Heading, VeDBA, Constant Speed, Dead-reckoning, and Dead-reckoning with GPS Correction.

### 3.2.1 GPS Import

FRAMEWORK 4 is set up to load data from separate GPS and DD files, although both are assumed time-synchronised. In this step, the user can import the GPS data and merge it back in to the complete data set. The same import wizard is used as in the file importer to import the GPS data. The user imports the data, then selects the data attributes for the available time fields in both the DD and GPS data sets (i.e. day, hours, minutes, and seconds). The wizard

1 matches the time index columns and inserts the GPS data fields wherever a matching data  
2 item with the same time stamp is found.

### 3 3.2.2 Alignment

4 Within the DD, the accelerometer and  
5 magnetometer coordinate systems are not  
6 necessarily aligned such that the three axes of the  
7 magnetometer and accelerometer in the same  
8 direction. Appreciation of their relative positions  
9 is, however, vital for computing the heading and  
10 this is applied using rotational information from  
11 the accelerometer and applying it to the  
12 magnetometer channels. To deal with this, the  
13 program asks the user to specify the orientation of  
14 the device. As the user selects different  
15 orientations, the image of the device is updated  
16 appropriately (Fig. 2). After the correct settings  
17 have been chosen, the sensors attributes are  
18 transposed so they align correctly with each other.  
19  
20  
21  
22  
23  
24  
25  
26



27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  

Figure 2 - This image shows the alignment correction step. An image of the sensor board is updated to show the alignment selected.

### 3.2.3 Heading

Deriving the heading necessitates that the data attributes be sanitised. This is done via; magnetometer calibration, pitch and roll computation, coordinate frame adjustments, and finally the heading derivation. These processes are executed in the background of the wizard and the user is exposed to certain parameters and results which allow them to select the correct attributes and associated settings for each stage in the wizard. For clarity we now detail the methods for each process.

#### 3.2.3.1 Ellipse fitting

The measurements obtained by commercial magnetometer sensors are corrupted by several errors. Proper calibration of the magnetometer is required to obtain high accuracy heading measurements. Errors can be usually introduced by instrumental errors, such as scale factors, non-orthogonality between axes, offsets and magnetic deviations caused by perturbations and interference with the magnetic field lines.

Magnetic measurements are subject to sources of error primarily caused by hard iron and soft iron deposits acting on the magnetic field. Rotating a magnetometer around 360 degrees in all orientations under no sources of error should produce a perfect sphere centred on the origin. Hard iron effects are created by objects which produce a magnetic field with a constant bias in the output, resulting in a sphere displaced from the origin. Soft iron deposits are caused by ferrous materials which are more permissive to the magnetic field influencing the magnetic field as it passes through, via distortion or stretching. This distorts the sphere into an ellipse as hard iron errors are independent of the orientation of the device and can therefore occur across the sphere. Hard iron distortions are caused by metals such as nickel and iron and commonly have a much larger contribution towards the total error (Cai et al. 2001).

We utilise the state-of-the-art error model presented by Renaudin et al. (2010) consisting of an ellipsoid-fitting algorithm based on an adaptive least squares estimator which calibrates the magnetometer readings for both instrumental errors and magnetic deviations.

### 3.2.3.2 Pitch and Roll Computation

In order to determine the heading of a device affixed to an animal, the magnetometer should ideally have an  $x$   $y$  plane that is parallel to the earth's surface, something that is not always the case. Correction for this requires a correction for tilt and this can be achieved by using the static acceleration by deriving a moving mean from each of the acceleration axes (Shepard et al. 2008). For a sample  $S_j$ , with a smoothing window of  $w$  this is given by:

$$S_i = \frac{1}{w} \sum_{j=i-\frac{w}{2}}^{i+\frac{w}{2}} S_j$$

Device pitch and roll are given by the arctangent of the relevant smoothed  $x$ ,  $y$ , and  $z$  values.

$$\text{Roll } (\phi) = \left( \text{atan2} \left( S_x, \sqrt{S_y \cdot S_y + S_z \cdot S_z} \right) \right) \cdot \frac{180}{\pi}$$

$$\text{Pitch } (\theta) = \left( \text{atan2} \left( S_y, \sqrt{S_x \cdot S_x + S_z \cdot S_z} \right) \right) \cdot \frac{180}{\pi}$$

### 5.2.3.3 Coordinate frame adjustment

Once derived, device attitude via pitch and roll can be used to rotate magnetometer measurements. Here, magnetometer  $x$ ,  $y$ ,  $z$  values are rotated by the inverse of the pitch ( $\theta$ ) and roll ( $\phi$ ) to access the rotated column vector  $m^r$  using;

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{pmatrix}$$

$$R_y(\phi) = \begin{pmatrix} \cos \phi & 0 & -\sin \phi \\ 0 & 1 & 0 \\ \sin \phi & 0 & \cos \phi \end{pmatrix}$$

$$m^r = R_x(\theta)R_y(\phi) \begin{pmatrix} m_x \\ m_y \\ m_z \end{pmatrix}$$

### 3.2.3.4 Heading derivation

The compass heading ( $H$ ) with respect to magnetic north is determined using the  $x$  and  $y$  tilt- and error-corrected magnetometer components utilising;

$$H = \left( \text{atan2}(m_y, -m_x) \right) \cdot \frac{180}{\pi}$$

### 3.3.4 Using Vectorial Dynamic Body Acceleration (VeDBA) for Speed

VeDBA can be used for predicting speed in terrestrial animals (Bidder et al. 2012).  
 FRAMEWORK 4 calculates VeDBA using;

$$VeDBA = \sqrt{(DA_x^2 + DA_y^2 + DA_z^2)}$$

where  $DA_x$ ,  $DA_y$  and  $DA_z$  are the dynamic acceleration values derived by taking the absolute values of running means of the raw acceleration values of each of the 3 orthogonal measurement axes from the corresponding raw acceleration values. The program allows users to select the length of the running mean although a default of 2 s is used (Shepard et al. 2009). Animal speed ( $s$ ) can then be computed by applying a speed coefficient  $m$  and adding a constant  $c$  to the VeDBA value. The speed coefficient and offset is adjusted in the dead-reckoning wizard page. In addition, we utilise a threshold  $t$  whereby, if the VeDBA falls below this value, the VeDBA is set to zero.

$$s = \begin{cases} (VeDBA \cdot m) + c & \text{if } s > t \\ 0 & \text{else} \end{cases}$$

### 3.3.5 Constant Speed

For volant species, the dynamic acceleration is unlikely to correspond to the speed of the animal, invalidating VeDBA in this context. Until a satisfactory measure of speed is derived (such as a pitot tube), we suggest using a constant speed, with options to correct this using the GPS data later.

### 3.3.6 Dead-reckoning

Dead-reckoning combines the speed and heading to compute a trajectory for the given data with calculations computed in Latitude and Longitude to compensate for the spherical nature of the earth. Utilising Cartesian coordinates for dead-reckoning introduces error by computing trajectories based on linear distances.

There are a number of parameters which must be defined first by the user. These are; (i) an initial start position defining the point where the path starts (if GPS data is given, the start coordinates are taken from this), (ii) a number of speed parameters for the VeDBA threshold and speed coefficients (see Bidder et al. 2012, this volume) and (iii) a heading offset corresponding to the declination angle obtained from the NGDC website. The computed path is shown alongside the map so that the user can interactively adjust parameters and see the result on the generated path.

To compute the path, the speed must be converted to radial distance in terms of the radius of the earth  $R$  ( $6.371 \times 10^6$  m). This is calculated as.

$$q = \frac{d}{R}$$

The Latitude and Longitude at time  $T_i$  (where  $T_0$  is equal to the starting point of the track ) can be calculated as follows.

$$Lat_i = \text{asin}(\sin Lat_{i-1} \cdot \cos q + \cos Lat_{i-1} \cdot \sin q \cdot \cos H)$$



$$Lon_i = Lon_{i-1} + atan2((\sin H \cdot \sin q \cdot \cos Lat_{i-1}), (\cos q - \sin Lat_{i-1} \cdot \sin Lat_i))$$

The complete set of latitude and longitude points defines the trajectory of the animal movement.

### 3.3.7 Dead-reckoning with GPS Correction

The dead-reckoning solution can be subject to cumulative errors from the estimates of derived heading and speed. Even small, but systematic, errors in these channels will accumulate over time and can increase the resulting error correspondingly. To overcome these problems, the program utilises a dead-reckoning and GPS correction algorithm to correct the heading and speed of the obtained dead-reckoned trajectory using the aligned GPS trajectory information. This resets the accumulated error at each GPS fix.

We correct the heading and speed by adjusting the length and heading of the dead-reckoned path such that they align to the same positions along each fix along the GPS path. For this, we define two equations; the heading between two points and the distance.

The heading ( $\theta$ ) between two points ( $Lat_0, Long_0$ ) and ( $Lat_1, Long_1$ ) is calculated as so:

$$\begin{aligned} TrajectoryHeading(\theta) \\ &= atan2(\cos(Lat_0) * \sin(Lat_1) - \sin(Lat_0) * \cos(Lat_1) \\ &* \cos(Long_1 - Long_0), \sin(Long_1 - Long_0) * \cos(Lat_1)) \end{aligned}$$

The distance ( $d$ ) between two points ( $Lat_0, Long_0$ ) and ( $Lat_1, Long_1$ ) is calculated as so:

$$latitudeDistance(latD) = Lat_1 - Lat_0$$

$$longDistance(longD) = Long_1 - Long_0$$

$$arcDist = \sin\left(\frac{latD}{2}\right) * \sin\left(\frac{latD}{2}\right) + \sin\left(\frac{longD}{2}\right) * \sin\left(\frac{longD}{2}\right) * \cos(Lat_0) * \cos(Lat_1)$$

$$TrajectoryDistance(d) = R * 2 * atan2(\sqrt{arcDist}, \sqrt{1 - arcDist})$$

The headings and speed between fixes can be adjusted to those of the GPS fixes in an iterative manor. For two sequential GPS fixes there are usually many dead-reckoned fixes in-between. Firstly the heading is adjusted, this consists of adding a constant heading to all the dead-reckoned headings between the GPS fixes.

$$headingCoefficient(hC) = (gpsHeading - drHeading)$$

A speed coefficient to adjust the speed by is then computed. This consists of multiplying the speed values (derived from the VeDBA) between the GPS fixes.

$$speedCoefficient(sC) = \left(\frac{gpsDistance}{drDistance}\right)$$

The formulae for dead-reckoning are now adjusted to generate a dead-reckoned GPS-corrected path. The speed coefficient is multiplied by the original speed coefficient, along

with an addition of the heading offset, specific for that section of the track to counter the deviations from the GPS trajectory.

$$Lat_i = \text{asin}(\sin Lat_{i-1} \cdot \cos(q * sC) + \cos Lat_{i-1} \cdot \sin q \cdot \cos(H + hC))$$

$$Lon_i = Lon_{i-1} + \text{atan2}((\sin(H + hC) \cdot \sin(q * sC) \cdot \cos Lat_{i-1}), (\cos(q * sC) - \sin Lat_{i-1} \cdot \sin Lat_i))$$

The process is repeated by applying the GPS-corrected path in place of the dead-reckoned path utilising the previous formulae. Each iteration makes the path adhere more tightly to the GPS fixes as the heading and speeds are adjusted. We allow the user to repeat the adjustment process a set amount of times or continue until the speed and heading adjustments are under a specified threshold.

The particular advantage of a dead-reckoned track is that it can give very fine detail in the route of an animal and do so without reference to external ground-truthing sources, although confidence in the precision of the route will inevitably decrease with increasing time between ground-truthed points (Bidder et al. this volume). Nonetheless, the approach has particular value in being able allude to trajectories where conventional tracking methods are useless (Fig. 3).

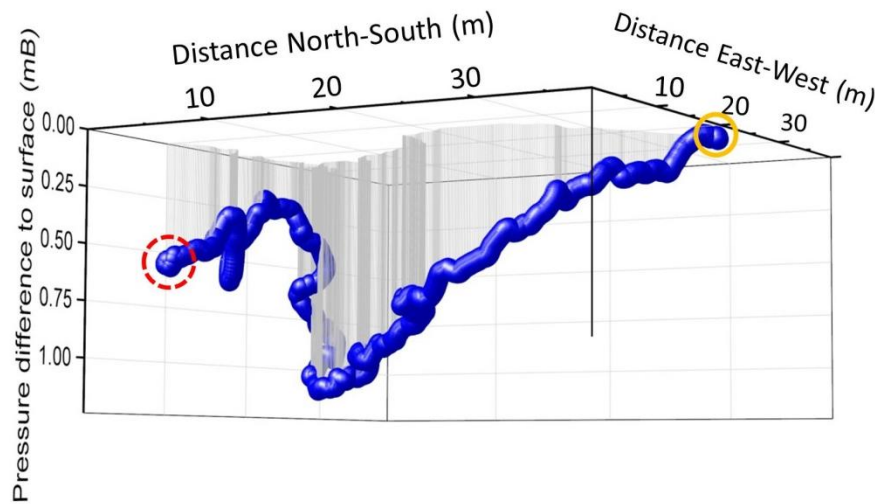


Fig. 3 – Dead-reckoned track of a European badger (*Meles meles*) in Northern Ireland leaving its sleeping quarters (red dashed circle) and moving through the underground sett to emerge at the entrance (yellow circle). The vertical axis representing depth is shown as the pressure difference between the surface and any time underground. The reconstruction assumes that animal speed is directly proportional to VeDBA (Bidder et al. 2012) underground in the same way it is on the surface. If this is not the case, the derived distances will be affected accordingly.

#### 4. Derivation of animal behaviour

##### 4.1 Problems with traditional Machine Learning Techniques

1 Extracting animal behaviour from the raw DD data is a time-consuming and cognitively  
2 demanding process for human analysis. Machine learning classification processes can be  
3 applied to identify and label behaviour by executing algorithms which learn from data to  
4 discover previously unknown properties (Esling et al. 2012) The learning aspect is typically  
5 split into two categories, supervised, and unsupervised learning. Supervised learning  
6 algorithms are trained on labelled data to generalize relationships between input and output  
7 samples. Conversely, unsupervised learning algorithms operate on unlabelled data to find  
8 previously unknown structure, and domain knowledge can then be applied to match found  
9 structure to a behaviour.  
10

11 Supervised techniques build a model from labelled data which generates predictions in  
12 response to new data. Traditionally, K-nearest neighbour (K-NN), support vector machines  
13 (SVM), and random forests have all been applied to accelerometer data. K-NN is a ‘lazy’  
14 learning method which predicts class membership based on the  $k$  closest training examples in  
15 the feature space (e.g. Bidder et al. 2014). The SVM algorithm finds a hyperplane which  
16 separates the feature space into the classes defined in the training set. Unseen data is assigned  
17 to a class based on the hyperplane region under which it falls (e.g. Gao et al. 2013). Random  
18 forests are the current state-of-the-art classification method in the data-mining community  
19 (e.g. Ellis et al 2014). Random forests construct many decision trees, each modelling the  
20 training set, with each tree voting for the resulting classification. A data item is assigned to  
21 the class with the most votes.  
22  
23  
24  
25

26 Cluster analysis is an unsupervised learning technique for exploratory analysis to find natural  
27 groupings of data. The most common clustering method used is K-Means. Data is partitioned  
28 into  $k$  clusters based on the distance to the nearest mean. The disadvantage of k-means is the  
29 need to specify how many partitions are in the data prior to clustering. Hierarchical clustering  
30 does not require any input parameters, instead grouping data over a variety of different scales  
31 to form a cluster tree, often visualized as a dendrogram. The user can choose the level most  
32 appropriate to their needs. However, this method is computationally expensive (order  $N^2$ ).  
33  
34  
35

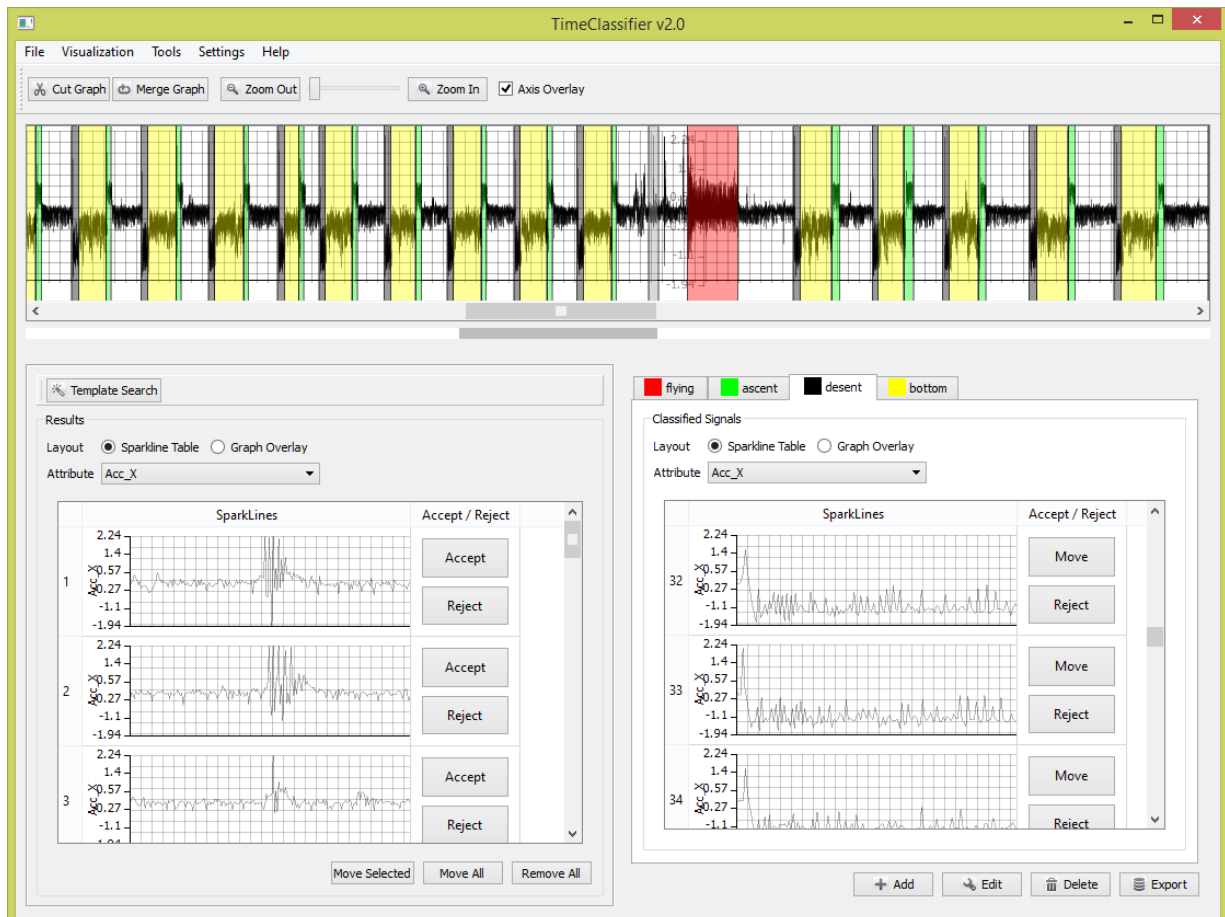
36 Supervised learning algorithms require extensively labelled instances of behaviours.  
37 Obtaining this data is time consuming, requires domain expertise, and the undertaking of  
38 field studies to gather video-synchronised data. It is obviously not possible to obtain all such  
39 data in all cases due to environmental constraints. Secondly, the choice of algorithm and  
40 parameters provides its own class of problems. Typically, in this process, the data  
41 dimensionality is reduced to a few parameters which contain the relevant information to  
42 perform classification; feature extraction. Good classification results rely heavily on the  
43 features chosen, however, extracting a desirable feature set is considered more of an art than  
44 a science and takes a great amount of skill along with trial and error (Smith 1997). Once the  
45 data is classified, if the accuracy is less than desired, decisions must be made as to whether it  
46 is useful to invest more time creating additional training input, modify the parameters, or use  
47 a different learning algorithm. It is not obvious what the next best step to take is without  
48 sufficient knowledge of the underlying algorithms.  
49  
50  
51  
52

#### 53 **4.2 FRAMEWORK 4 approach to behavioural identification.**

54 Unlike the previous methods, our method for identifying animal behaviour can use just one  
55 positive example and does not require large extensively labelled collections of behaviours  
56 (Walker et al. 2015 – Submitted to The Euro Graphics Conference on Visualization).  
57 Complicated parameters are avoided by utilising interactive visual interfaces which draw  
58 from the domain expertise when selecting matching behaviours in the input signal. A  
59  
60  
61  
62  
63  
64  
65

1 feedback loop is incorporated such that the precision and recall can be boosted by applying  
2 the user's subject knowledge. These features overcome the disadvantages of machine  
3 learning and provide a working solution that can cope with large complex data sets, a vital  
4 element given the rates at which DDs record (e.g. 10 channels, each at 40 Hz). The result is a  
5 system which supports the manual labelling of animal behaviour complimented with a user-  
6 driven approach for the semi-automatic classification of animal behaviour, requiring one  
7 instance of behaviour for the matching process to take place.  
8  
9

10 The user interface is split into three components (see Fig. 4 for overview). This consists of  
11 the data view at the top, being composed of the data in a stacked time-series graph format.  
12 Coloured segments overlaid on the graph indicate classified animal behaviour. A search panel  
13 is located in the bottom left, within which the user can perform searches on the data utilizing  
14 the template search wizard. Results are shown in this panel for the user to test, reject or  
15 accept results before moving to the appropriate classification in the bottom right panel, where  
16 the classification widget is situated. Classified behaviours are shown to the user in this  
17 tabulated panel. Each tab represents a separate behaviour and contains visualizations of the  
18 corresponding set of classified instances. The colours assigned to each tab correspond to  
19 those overlaid transparently on the time-series graph.  
20  
21  
22



23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  

Figure 4 – An image of an overview of the user interface provided in FRAMEWORK4. In the top there is a stacked time-series graph with labelled behaviours overlaid as transparent windows. The search widget (bottom left), allows the user to search for behaviours and filter through found instances by accepting and rejecting. Accepted results are moved into the

1 *classification widget (bottom right). This view encapsulates the behaviour groups and*  
2 *associated classified instances.*

3  
4 The components of our system and each of the processes towards classifying animal  
5 behaviour are split into five steps; (1) the user must select a behaviour to find in the data set,  
6 (2) matching is performed to find the similarity across the series, (3) A classification wizard  
7 allows the user to apply their knowledge to extract matches, (4) Extracted behaviours are  
8 presented to the user, (5) The user can improve the accuracy of their results by applying a  
9 feedback loop, (6) The classified results are displayed.

#### 11 *4.2.1 Behaviour Selection*

12 The first step in our system is for the user to select a behaviour to classify in the data. There  
13 are two methods for this in the application. Firstly, query-by-example, and secondly selecting  
14 previously saved behaviour instances from the template database.

15  
16  
17  
18 Query-by-example allows the direct selection of animal behaviour by drawing a window  
19 across the time-series encapsulating a subset of data exemplifying the behaviour the user  
20 wishes to search for. After selection, a dialog is then displayed where the user can select the  
21 data attributes to utilize for the template. Any data attribute can be used for searching  
22 throughout the system (not just the accelerometer component). For example, the  
23 magnetometer attribute is useful for finding thermalling cycles in condors, while the pressure  
24 component can indicate diving cycles in aquatic species.

25  
26  
27  
28 Behaviour templates used in the system are stored in a database back-end for future use. This  
29 is particularly useful because assigned behaviours can be used to search other files. The  
30 database can be set up to deal with behaviour templates assigned to classes of animals. The  
31 user can query for all patterns present for a specific animal or select an existing behaviour  
32 template previously saved in the database by navigating to the animal of interest and then  
33 selecting the appropriate behaviour template.

34  
35  
36  
37 The signal may be resampled to capture events at different frequencies as some behaviours  
38 occur at different speeds, for example running. To capture these events independently of the  
39 time duration, we can store and search for the signal at different time-intervals using  
40 resampling. Re-sampling is used by specifying an irrational factor consisting of an  
41 interpolation factor (amount to up-sample by) and a decimation factor (amount to down-  
42 sample by)e.g. resampling by 1 / 2 will half the sampling rate, while resampling by 2 / 1 will  
43 double the duration.

#### 44 *4.2.2 Template Matching*

45  
46 Supervised machine learning techniques rely on large bodies of labelled data. Such  
47 extensively labelled data sets do not exist in our application area. An alternative solution for  
48 classification is to use template matching, a process for determining the presence of a known  
49 waveform in a larger dataset. In essence, this works by sliding the specified template across  
50 the data set, computing the similarity of the template at each position in the data series  
51 corresponding to the concordance in fit between template and the sample at the position. The  
52 result is a similarity value at each position in the time-series. This allows the user to select a  
53 single positive example of a behaviour and search for all occurrences of it in the data.

54  
55  
56  
57  
58 A distance measure is used to determine a quantitative value corresponding to similarity or  
59 dissimilarity between time-series. Correlation is the optimal technique for detecting a known  
60  
61  
62  
63  
64  
65

1 waveform in random noise (Smith 1997). In signal processing, it is well known that  
2 Correlation has a linear complexity frequency space. We utilise correlation and a new fast  
3 normalized cross-correlation method for template matching in order to obtain results within  
4 real time (seconds) which maintains an interactive implementation. Standard cross-  
5 correlation performs matching, taking into account amplitude information, while normalized  
6 cross-correlation normalizes the template signal and current area under the template such that  
7 amplitude shifts are not taken into account. This is important when performing matching over  
8 regions where the represented waveform may be present at different orientations. Re-  
9 sampling the signal allows us to introduce time axis distortion to extract behaviours occurring  
10 at different durations.  
11

### 12 4.2.3 Classification Wizard

13 Once the template matching algorithm has been executed, the user is presented with the  
14 pattern-matching results in the classification wizard. The classification wizard is used to  
15 guide the user through refining a similarity threshold to verify matched signals. The user  
16 interactively modifies the threshold value which corresponds to the similarity of extracted  
17 matches. Matches are depicted to the user in an intuitive format with interaction to modify the  
18 result set according to the user's domain knowledge.  
19  
20  
21

22  
23 The classification wizard (Figure 5) is used to find all instances of the specified behaviour in  
24 the data series. The aim is to then obtain an appropriate threshold value through the  
25 interaction and inspection of visualization which maximizes the number of instances found,  
26 while minimizing the number of misclassifications. The similarity threshold is represented as  
27 a percentage of the match, with one hundred percent similarity representing an exact match,  
28 while zero represents no matching features. The user needs to find an appropriate estimate  
29 value using their expert knowledge of behavioural patterns and their occurrences in the data  
30 set.  
31  
32

33  
34 The classification wizard features two views. On the left (Figure 5 (b)) are visualisations to  
35 show where matches occur in the data series, while on the right (Figure 5 (c)), all of the  
36 extracted matches are overlaid on top of each other to show the variance between matches.  
37 The visualisations are updated as the threshold value is refined by adjusting a slider  
38 corresponding to the threshold percentage (Figure 5 (a)).  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65

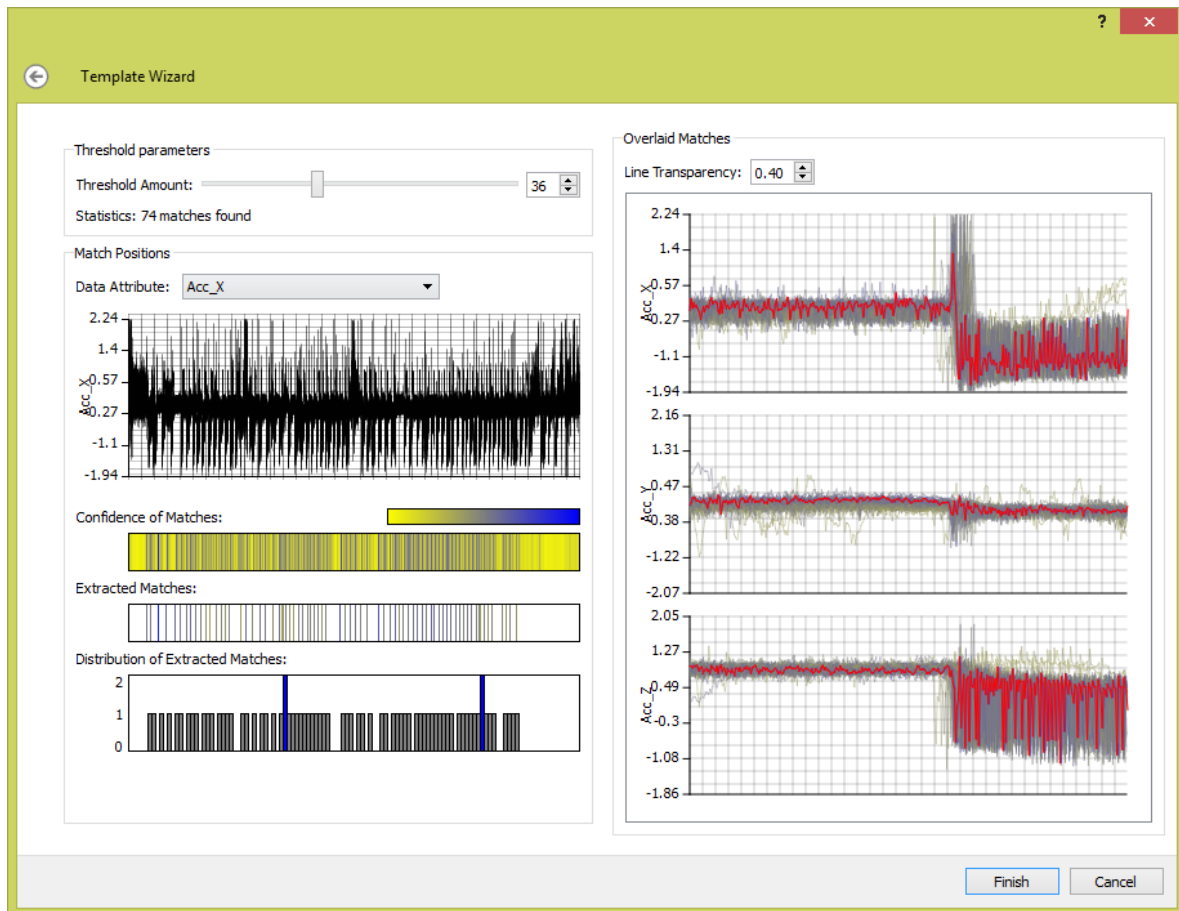


Figure 5 - This figure shows the classification wizard. (a) Illustrates our wizard parameters for dynamically adjusting the threshold. (b) Shows our density based visualizations to gain an understanding of where matches occur in the data series. (c) Shows our overlaid signals visualization of all the extracted matches in a stacked graph format, with one graph for each attribute of the template. The template signal is overlaid in red to show a direct comparison. A yellow to blue color scheme is used, yellow representing low similarity matches, while blue encodes high similarity matches

The positions where matches occur in the series are depicted using three graphical views (Figure 5 (b)), all of which are aligned with a time-series graph of the data series (Figure 5 (b1)). The confidence of a match visualization (Figure 5 (b2)) depicts a heatmap showing an overview of the pattern matching results to encode where high (blue) and low (yellow) similarity matches occur in the data series. The extracted matches view (Figure 5 (b3)) depicts where the extracted matches occur in the series and is updated as the similarity threshold is adjusted. Finally, a distribution of extracted matches (Figure 5 (b4)) which utilises a histogram, shows the number of matches at each position. The user may refine the result set by rejecting results by clipping rectangular regions of matching results from the data series to reject. This allows the user to reject results based on their knowledge of where they expect results to be present in the data series and the temporal trends expected.

All of the extracted matched signals are overlaid in a stacked time-series graph format, one graph for each data attribute of the pattern (Figure 5 (c)). The user can gain an overview of the general shape of the extracted signals from the graphs. This allows the verification of the shape of extracted matches as most outliers stand out immediately, not fitting into the general shape of the extracted results. The same colour-encoding scheme as the position of matches is

1 used to encode the strength of a match. As the user adjusts the threshold, results are added or  
2 removed. The user can directly see the cause and effect of modifying the threshold on the  
3 general shape of matched signals in comparison to the template signal overlaid in red. Results  
4 can be rejected in this view by manual selection of lines on the time-series graphs. All results  
5 falling within the selection are removed from the result set. This allows the user to filter  
6 results that should not be present manually. The user continues adjusting the similarity  
7 threshold and rejecting results in the synchronised graphical views until they are satisfied  
8 with the results being extracted. The user clicks 'finish' and the wizard closes.  
9

#### 10 *4.2.4 Results*

11 After the user has concluded with a good threshold value, the results are extracted and added  
12 to the results widget in the bottom left of Figure 5. The user can further inspect the results  
13 using our two views. Firstly the sparkline (embedded time-series) display, this puts the  
14 classifications in a table format, with each row corresponding to an identified instance of a  
15 behaviour visualized using a sparkline. The user can accept or reject results by selecting  
16 rows. Secondly, the overlaid plot view overlays the classified instances in a time-series graph.  
17 The user can accept or reject results by selection on the time-series. The overlaid plot view is  
18 useful where the signals shape is similar amongst results. Conversely, the sparkline display is  
19 useful where the behaviour signal varies. Matches displayed in the results view are also  
20 shown in the data view overlaid on top of the time-series graph in grey.  
21  
22  
23  
24

#### 25 *4.2.5 Improving Precision and Recall*

26 The variability and inconsistency of animal behaviour makes the automatic labelling of  
27 behaviour a challenging task. It is widely accepted in the machine learning community that  
28 achieving 100% precision and recall is a difficult, if not impossible, task. From a movement  
29 ecology viewpoint, we aspire to close to perfect labelling of behaviour. We incorporate a  
30 feedback loop which draws from domain expertise to enhance results. Firstly, the user can  
31 provide secondary examples of a behaviour to find more behaviour instances. Secondly, the  
32 user can directly manipulate the result set to accept and reject matches. Finally, the user can  
33 manually classify behaviour.  
34  
35  
36

37 Where the user believes the number of found instances to be low, boosting can be used to  
38 retrieve more instances. More examples of a behaviour are utilised in the template for  
39 searching. This, in effect, widens the search span to find patterns related to the secondary  
40 retrieved patterns but may not be directly related to the initial search pattern.  
41  
42

43 The results panel provides an effective means to inspect the newly found behaviour  
44 classifications. Results are accepted by moving them to an appropriate classification tab in  
45 the classification widget, or rejected by clicking the reject button. The reject button removes  
46 the result from the panel. The user should aim to keep accepting / rejecting results until this  
47 panel is empty.  
48  
49

50 We appreciate that some instances will never be identified by machine learning and may only  
51 be able to be extracted by the domain expert, be that because of a low number of instances of  
52 the behaviour, or because of the variability of the animal behaviour. We enable manual  
53 labelling in our system so that the user can manually select and classify behaviour regions. To  
54 classify a behaviour region in the data manually, the user cuts the time-series graph up into  
55 segments. Each cut contains a start and end boundary defined by that of a behaviour instance.  
56 Once a behaviour region is cut in the time-series graph, the user drags and drops the time-  
57  
58  
59  
60  
61  
62  
63  
64  
65

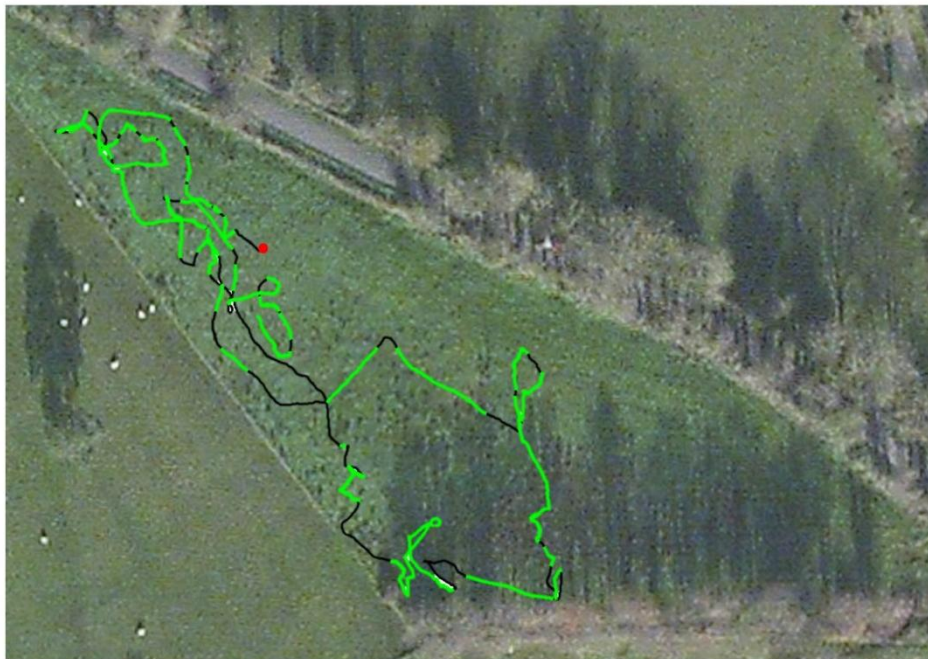


1 series segment into the appropriate classification tab. The cutting and dragging of data  
2 samples is similar to that used in video-editing software.

#### 3 4 4.2.6 *Classified Results*

5 Classified behaviours are shown in two views. Firstly, the classification widget which  
6 displays classified behaviour in a corresponding tabulated view. Secondly, classified  
7 instances are aligned and overlaid on top of time-series graphs as coloured rectangular  
8 regions identifying where in the data a match for the behaviour has occurred. Each behaviour  
9 is identified by a unique colour assigned to each classification tab in the classification widget.  
10

11 A typical output of this process is that, not only can animal behaviour be classified with  
12 respect to time, but that the occurrence of different behaviours can be represented on GPS-  
13 enabled dead-reckoned animal tracks in an obvious colour scheme (Fig. 6).  
14  
15



40  
41 *Fig 6. The dead-reckoned trajectory of a cow (Bos taurus) in a field in Northern Ireland over*  
42 *2 h, colour-coded according to different activities – green = grazing, black = walking, red =*  
43 *lying down.*  
44

## 45 46 **5. Derivation of animal energy expenditure**

### 47 48 5.1 *Proxies for energy expenditure.*

49 Since the suggestion by Wilson et al. (2006) that dynamic body acceleration could be used as  
50 a proxy for  $VO_2$ , there have been a number of studies that have confirmed its utility in species  
51 ranging from shellfish, through fish, amphibia and reptiles to birds and mammals (see Halsey  
52 et al. 2011 for review). Two measures have been used, Overall Dynamic Body Acceleration  
53 (ODBA) and Vectorial Dynamic Body Acceleration (VeDBA), which are essentially  
54 equivalent in terms of their power to predict  $VO_2$  (Qasem et al. 2012) although VeDBA has  
55 more utility for predicting speed (Bidder et al. 2012). FRAMEWORK 4 uses VeDBA (see  
56 section 3.3.4 for calculation) as a proxy for  $VO_2$  (Halsey et al. 2009) and so that plots of  
57 animal trajectory can be colour coded accordingly (Fig. 7).  
58  
59  
60  
61  
62  
63  
64  
65

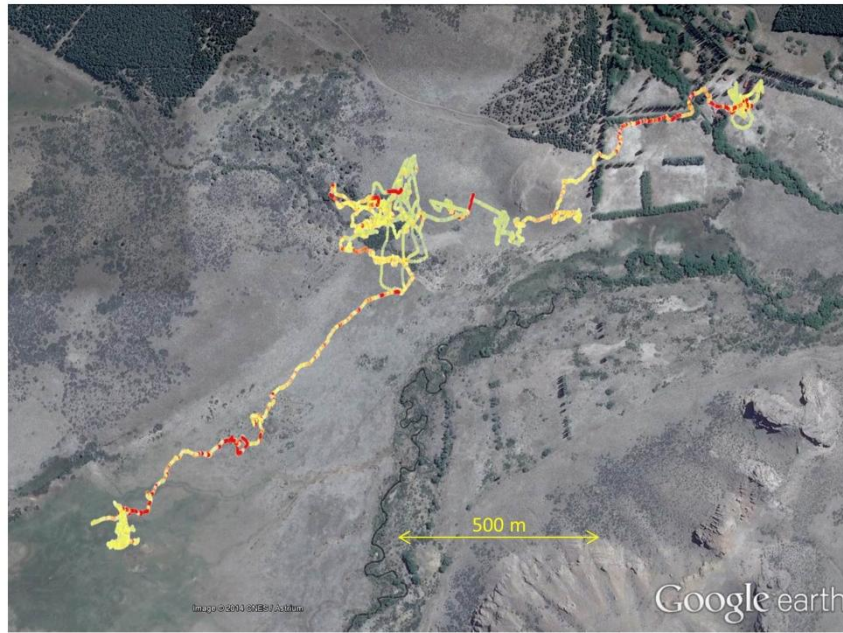


Fig. 7 – The dead-reckoned trajectory of a sheep (*Ovis aries*) in Argentina over 14 h showing how VeDBA, a metric that correlates linearly with metabolic rate, (ranging from pale green [low values] through yellow to red [high values] varies with location and track tortuosity. Note how higher track tortuosity is generally linked to lower VeDBA.

## 6. Derivation of the physical characteristics of the environment

Many animals modulate their behaviour in the environment according to its physical characteristics. For instance, reptiles may associate with areas of high temperature or insolation to warm up (Chelazzi and Calzolari 1986) while many bird species are limited in their foraging capacities by light (e.g. Wanless et al. 1999). Thus, the ability to resolve the geographic position of animals in tandem with environmental variables can help explain the incidence or emergence of particular behaviours (e.g. Wilson et al. 1993). Thus, the particular value of FRAMEWORK4 is that it allows the physical attributes of the environment to be allocated to space (and time) and to be associated with identified behaviours (Fig. 8).

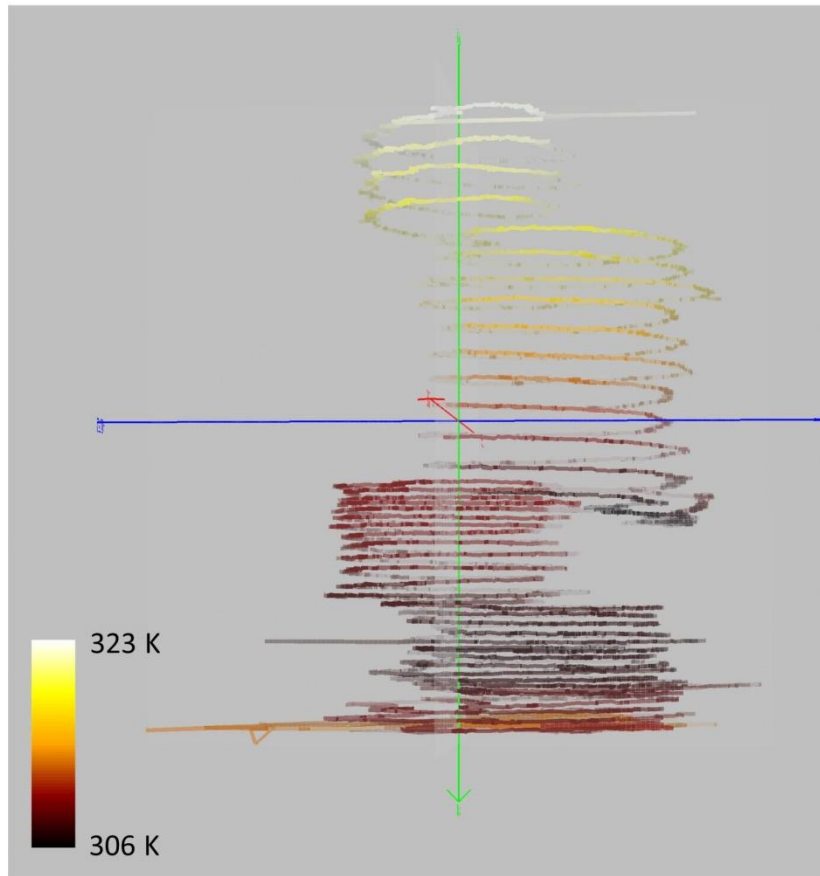


Fig. 8 – Track made by an Andean condor (*Vultur gryphus*) in Argentina during ascent in a thermal. The track is colour coded according to the potential temperature (a useful meteorological metric).

### 7. Merged file output

The derived analytical attributes from the software can be outputted together into one data file. Exporting data is supported via navigation to the ‘Export’ option in the ‘File’ menu on the main tool bar. Export is undertaken in CSV format where derived attributes are appended as an additional column in the data file alongside the existing data channels. Each behaviour is assigned a unique numerical value where, if a data item falls within a labelled region, it is assigned this value.

### 8. Obtaining Software

The software is freely available for download from the following web address. The website features instructional videos and documentation on using the software. As the website evolves there will become more documentation and features available. We hope this will be the foundation of a variety of software techniques for animal movement analysis.

<http://www.framework4.co.uk>

We ask if you utilise the software in a publication that you reference this paper appropriately.

### 9. Conclusion

1 This is a first attempt to create a single program that explicitly links space use, movement,  
2 behaviour, and energy expenditure in free-living animals together with environmental  
3 conditions, doing so using an accessible column-separated format for ASCII-type data.  
4 Although there is appreciable room for improvement in many facets of the program at the  
5 moment, the aspiration is to progress and refine it to make it as powerful as possible and  
6 thereby provide a methodology which will enhance our understanding of the processes that  
7 affect the way that animals move within their environment.  
8

### 9 **Acknowledgements**

10 JSW was funded by an EPSRC doctoral training grant. IEM and EAM were funded by PhD  
11 studentships from the Department of Employment and Learning (DEL) and the Department  
12 of Agriculture and Rural Development (DARD), Northern Ireland, respectively. HW  
13 benefitted from a Swansea University PhD studentship. The Andean condor work was  
14 conducted in collaboration with Sergio Lambertucci, it was funded by National Geographic  
15 (grant GEFNE89-13 awarded to E Shepard), CONICET (grants PIP-0095, and ANPCyT–  
16 PICT-1156 awarded to S Lambertucci) and Swansea University, and data were collected with  
17 the help of a large field team.  
18  
19  
20

### 21 **Declarations – Competing Interests**

22 The author(s) declare(s) that they have no competing interests.  
23  
24

### 25 **References**

- 26  
27 Aguilar Soto, N., M. P. Johnson, et al. (2008). "Cheetahs of the deep sea: deep foraging  
28 sprints in short-finned pilot whales off Tenerife (Canary Islands)." Journal of Animal  
29 Ecology **77**(5): 936-947.  
30  
31 Bidder, O. R., Qasem, L. A., Wilson, R. P. (2012). On higher ground: How well can dynamic  
32 body acceleration determine speed in variable terrain? PLoS One **7**: e50556  
33  
34 Bidder, O. R., H. A. Campbell, et al. (2014). "Love Thy Neighbour: Automatic Animal  
35 Behavioural Classification of Acceleration Data Using the K-Nearest Neighbour  
36 Algorithm." PloS one **9**(2): e88609.  
37  
38 Börger L, Dalziel BD, Fryxell JM (2008) "Are there general mechanisms of animal home  
39 range behaviour? A review and prospects for future research." Ecology Letters **11**;  
40 637-650.  
41  
42 Cai, J., Andersen, N. L., & Malureanu, C. (2001). "In-Field Practical Calibration of Three-  
43 Axis Magnetometers." Proceedings of the 2010 International Technical Meeting of  
44 The Institute of Navigation (pp. 67-73).  
45  
46 Caruso, M. J. (2000). "Applications of magnetic sensors for low cost compass systems." Position Location and Navigation Symposium, IEEE, San Diego, CA, USA,  
47 Honeywell, SSEC, pp. 177-184.  
48  
49 Chelazzi G, Calzolari R (1986) "Thermal benefits from familiarity with the environment in a  
50 reptile." Oecologia **68**: 557-558  
51  
52 Ellis, K, Kerr, J et al. (2014). "A random forest classifier for the prediction of energy  
53 expenditure and type of physical activity from wrist and hip accelerometers." Physiological measurement. **35**(11) 2191.2  
54  
55 Esling, P and Agon, C. (2012). "Time-series data mining" ACM Computing Surveys. **45**, 1,  
56 Article 12.  
57  
58 Gao, L, H. A. Campbell et al. (2013). "Web-based semantic tagging and activity recognition  
59 system for species' accelerometry data", Ecological Informatics, Volume 13, January  
60 2013, Pages 47-56  
61  
62  
63  
64  
65

- 1 Halsey, L., C.-A. Bost, et al. (2007). "A thorough and quantified method for classifying  
2 seabird diving behaviour." Polar Biology **30**(8): 991-1004.
- 3 Halsey, L. G., Shepard, E. L. C., Quintana, F., Gomez Laich, A., Green, J. A., & Wilson, R.  
4 P. (2009). "The relationship between oxygen consumption and body acceleration in a  
5 range of species." Comparative Biochemistry and Physiology Part A: Molecular &  
6 Integrative Physiology, 152(2), 197-202.
- 7 Halsey LG, Shepard ELC, Wilson RP (2011). "Assessing the development and application of  
8 the accelerometry technique for estimating energy expenditure." Comparative  
9 Biochemistry and Physiology A 158: 305-314
- 10 Hawkes, L. A., S. Balachandran, et al. (2011). "The trans-Himalayan flights of bar-headed  
11 geese (*Anser indicus*)." Proceedings of the National Academy of Sciences **108**(23):  
12 9516-9519.
- 13 Hooker, S., R. Baird, et al. (2001). "Behavioural reactions of northern bottlenose whales  
14 (*Hyperoodon ampullatus*) to biopsy darting and tag attachment procedures." Fishery  
15 Bulletin.
- 16 Ioalè, P., Dall'Antonia, P., Dall'Antonia, L., & Benvenuti, S. (1994). "Flight paths of homing  
17 pigeons studied by means of a direction recorder". Ethology ecology & evolution,  
18 6(4), 519-527.
- 19 Jonsen, I. D., R. A. Myers, et al. (2003). "Meta-analysis of animal movement using state-  
20 space models." Ecology **84**(11): 3055-3063.
- 21 Martiskainen, P., M. Järvinen, et al. (2009). "Cow behaviour pattern recognition using a  
22 three-dimensional accelerometer and support vector machines." Applied Animal  
23 Behaviour Science **119**(1): 32-38.
- 24 Miller, P. J., M. P. Johnson, et al. (2004). "Sperm whale behaviour indicates the use of  
25 echolocation click buzzes 'creaks' in prey capture." Proceedings of the Royal Society  
26 of London. Series B: Biological Sciences **271**(1554): 2239-2247.
- 27 Nathan, R., O. Spiegel, et al. (2012). "Using tri-axial acceleration data to identify behavioural  
28 modes of free-ranging animals: general concepts and tools illustrated for griffon  
29 vultures." The Journal of experimental biology **215**(6): 986-996.
- 30 Qasem L, Cardew A, Wilson A, Griffiths I, Halsey LG, Shepard ELC, Gleiss AC, Wilson R.  
31 (2012). "Tri-axial dynamic acceleration as a proxy for animal energy expenditure :  
32 Should we be summing values of calculating the vector?" PLoS One 7: e31187
- 33 Renaudin , V, Afzal, M. H. et al (2010). "Complete Triaxis Magnetometer Calibration in the  
34 Magnetic Domain" Journal of Sensors. Volume 2010, Article ID 967245
- 35 Ropert-Coudert, Y. and R. P. Wilson (2005). "Trends and perspectives in animal-attached  
36 remote sensing." Frontiers in Ecology and the Environment **3**(8): 437-444.
- 37 Rowley, J. J. and R. A. Alford (2007). "Techniques for tracking amphibians: the effects of tag  
38 attachment, and harmonic direction finding versus radio telemetry." Amphibia-  
39 Reptilia **28**(3): 367-376.
- 40 Sakamoto, K. Q., K. Sato, et al. (2009). "Can ethograms be automatically generated using  
41 body acceleration data from free-ranging birds?" PloS one **4**(4): e5379.
- 42 Shepard, E., R. Wilson, et al. (2008). "Identification of animal movement patterns using tri-  
43 axial accelerometry." Endang Species Res, doi **10**.
- 44 Shepard, E. L. C., R. P. Wilson, et al. (2009). "Derivation of body motion via appropriate  
45 smoothing of acceleration data." Aquatic Biology **4**(3): 235-241.
- 46 Smith, S. W (1997). "The scientist and engineer's guide to digital signal processing".  
47 California Technical Publishing, San Diego, CA, USA.
- 48 Vandeenabeele, S. P., E. L. Shepard, et al. (2012). "When three per cent may not be three per  
49 cent; device-equipped seabirds experience variable flight constraints." Marine  
50 Biology **159**(1): 1-14.
- 51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65

- 1 Vandeenabeele, S. P., R. P. Wilson, et al. (2013). "New tracking philosophy for birds."  
2 Frontiers in Ecology and the Environment **11**(1): 10-12.
- 3 Walker, J. S., Jones, M. W., Laramée, R. S., Bidder, O. R., Williams, H. J., Scott, R.,  
4 Shepard, E. L. C., Wilson, R. P. (2015). "TimeClassifier - A Visual Analytic System  
5 for the Classification of Multi-Dimensional Time-Series Data". Submitted to EuroVis  
6 conference (Awaiting review).
- 7 Wanless S, Finney SK, Harris MP, McCafferty DJ (1999). "Effect of the diel light cycle on  
8 the diving behaviour of two bottom-feeding marine birds: the blue-eyed shag  
9 Phalacrocorax atriceps and the European shag P. aristotelis." Marine Ecology  
10 Progress Series 188: 219-224.
- 11 Watanabe, S., M. Izawa, et al. (2005). "A new technique for monitoring the detailed  
12 behaviour of terrestrial animals: a case study with the domestic cat." Applied Animal  
13 Behaviour Science **94**(1): 117-131.
- 14 Wikelski M, Kays RW, Kasdin NJ, Thorup K, Smith JA, Swenson GW (2007). "Going wild:  
15 what a global small-animal tracking system could do for experimental biologists."  
16 The Journal of Experimental Biology 210; 181-186.
- 17 Wilson, R. P., D. Grémillet, et al. (2002). "Remote-sensing systems and seabirds: their use,  
18 abuse and potential for measuring marine environmental variables." Marine Ecology  
19 Progress Series **228**: 241-261.
- 20 Wilson RP, Culik BM, Bannasch R, Lage J (1993). "Monitoring Antarctic variables using  
21 penguins." Marine Ecology Progress Series 106: 199-202.
- 22 Wilson, R. P., N. Liebsch, et al. (2007). "All at sea with animal tracks; methodological and  
23 analytical solutions for the resolution of movement." Deep-Sea Research Part II-  
24 Topical Studies in Oceanography **54**(3-4): 193-210.
- 25 Wilson, R. P., K. Putz, et al. (1997). "Long-term attachment of transmitting and recording  
26 devices to penguins and other seabirds." Wildlife Society Bulletin **25**(1): 101-106.
- 27 Wilson, R. P., & Wilson, M. P. T. (1988). "Foraging behaviour in four sympatric  
28 cormorants." The Journal of Animal Ecology, 943-955.
- 29 Wilson, R. P., Shepard, E. L. C., Liebsch, N. (2008). "Prying into the intimate details of  
30 animal lives; use of a daily diary on animals." Endangered Species Research **4**: 123-  
31 137.
- 32 Wilson, R. P., C. R. White, et al. (2006). "Moving towards acceleration for estimates of  
33 activity-specific metabolic rate in free-living animals: the case of the cormorant."  
34 Journal of Animal Ecology **75**(5): 1081-1090.
- 35 Wilson, Rory P., B. Culik, R. Danfeld, and D. Adelung. "People in Antarctica—how much  
36 do Adélie Penguins *Pygoscelis adeliae* care?." Polar Biology 11, no. 6 (1991): 363-  
37 370.
- 38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65